

Oracle® Warehouse Builder

API and Scripting Reference

10g Release 2 (10.2)

B28225-01

June 2006

Oracle Warehouse Builder API and Scripting Reference, 10g Release 2 (10.2)

B28225-01

Copyright © 2000, 2006, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

This program contains Batik version 1.6.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

For additional information about the terms and conditions, search for "Apache License" in Oracle Warehouse Builder online help.

Contents

Preface	xv
Purpose	xv
Audience	xv
Documentation Accessibility	xvi
Related Documents	xvi
Conventions	xvii
Part I Application Programming Interfaces	
1 Public Views for the Runtime Environment	
Warehouse Builder Runtime Repository Public Views	1-1
Deployment Auditing Views	1-2
Execution Auditing Views	1-8
2 Public Views for the Design Environment	
Warehouse Builder Design Repository Public Views	2-1
General Model Views	2-7
Data Model Views	2-22
Flat Files Views	2-40
Collections Views	2-42
Function Model Views	2-43
Configuration Model Views	2-46
Deployment Model Views	2-47
Mapping Model Views	2-48
Process Flow Model Views	2-53
Data Profiling Views	2-56
Data Rules Views	2-63
User Defined Object Views	2-67
Expert Views	2-69
Business Intelligence Views	2-72
Real Time Views	2-84
Scheduling Views	2-86
Others	2-87

3 Using SQL*Plus to Schedule and Execute Jobs

Managing Jobs Using SQL Scripts	3-1
Starting ETL Jobs in SQL*Plus	3-2
Scheduling ETL Jobs in Oracle Enterprise Manager.....	3-3
The SQLPLUS_EXEC_TEMPLATE SQL Script	3-3
The WB_RT_API_EXEC.RUN_TASK Function.....	3-4
Managing a Control Center	3-6

Part II OMB Plus Scripting Language

4 Introduction to OMB Plus

About the OMB Scripting Language	4-1
Using OMB Plus	4-1
Writing OMB Plus Commands	4-2
Running Scripts in OMB Plus.....	4-4
OMB Plus Commands	4-5
Metadata Manipulation Language (MML) Commands	4-5
Examples	4-6
Shell Commands	4-7
Administrative Commands	4-8
Navigation Commands	4-8
Service Commands	4-9
How to Read Syntax Diagrams	4-10
Required Keywords and Parameters	4-11
Optional Keywords and Parameters	4-11
Syntax Loops.....	4-11
Sample OMB Plus Scripts	4-11
Updating a Design Repository	4-11
Reporting on Repository Objects	4-12
Finding Invalid Objects	4-13
Using OMB Plus to Navigate Repositories.....	4-13
New to OMB Plus in This Release	4-14
Commands Introduced in This Release	4-14
Changes to the OMB Plus Syntax	4-14

5 OMB Commands

OMBCAC	5-2
OMBCC	5-3
OMBCOMMIT	5-4
OMBCOMPARE SNAPSHOT	5-5
OMBCOMPILE.....	5-7
OMBCONN	5-9
OMBCONNECT	5-11
OMBCONNECT CONTROL_CENTER	5-13
OMBCONN CONTROL_CENTER	5-15
OMBCOPY.....	5-17

OMBDAC.....	5-19
OMBDCC.....	5-20
OMBDEBUG MAPPING.....	5-21
OMBDEFINE ASSOCIATION_DEFINITION.....	5-25
OMBDEFINE CLASS_DEFINITION.....	5-29
OMBDEFINE COMPONENT_DEFINITION.....	5-32
OMBDEFINE DOMAIN_DEFINITION.....	5-33
OMBDEFINE FOLDER_DEFINITION.....	5-36
OMBDEINSTALL OWB_REPOSITORY.....	5-37
OMBDEINSTALL OWB_TARGET_USER.....	5-38
OMBDEPLOY.....	5-39
OMBDERIVE.....	5-41
OMBDESCRIBE ASSOCIATION_DEFINITION.....	5-44
OMBDESCRIBE CLASS_DEFINITION.....	5-48
OMBDESCRIBE MODEL.....	5-50
OMBDISC.....	5-51
OMBDISCONNECT.....	5-52
OMBDISCONNECT CONTROL_CENTER.....	5-53
OMBDISC CONTROL_CENTER.....	5-54
OMBDISPLAYCURRENTMODE.....	5-55
OMBENV.....	5-56
OMBEXPORT.....	5-57
OMBEXPORT MDL_FILE.....	5-61
OMBGRANT DEFAULT_OBJ_PRIV.....	5-65
OMBGRANT OBJ_PRIV.....	5-67
OMBGRANT ROLE.....	5-69
OMBGRANT SYS_PRIV.....	5-70
OMBHELP.....	5-71
OMBIMPACT.....	5-72
OMBIMPORT.....	5-74
OMBIMPORT MDL_FILE.....	5-78
OMBIMPORT METADATA_LOCATION.....	5-81
OMBINSTALL OWB_RAC.....	5-82
OMBINSTALL OWB_REPOSITORY.....	5-83
OMBINSTALL OWB_TARGET_USER.....	5-85
OMBLINEAGE.....	5-86
OMBLIST.....	5-88
OMBLIST DEFAULT_OBJ_PRIVS.....	5-91
OMBLIST ICONSETS.....	5-92
OMBLIST OBJ_PRIVS.....	5-93
OMBLIST ROLES.....	5-95
OMBLIST SNAPSHOT.....	5-96
OMBLIST SYS_PRIVS.....	5-97
OMBLIST USERS.....	5-98
OMBLOCK.....	5-99
OMBMLSUPDATE OWB_REPOSITORY.....	5-101
OMBMOVE.....	5-102

OMBPROFILE.....	5-104
OMBRECONCILE.....	5-105
OMBREDEFINE ASSOCIATION_DEFINITION	5-108
OMBREDEFINE CLASS_DEFINITION.....	5-112
OMBREGISTER LOCATION	5-115
OMBREGISTER USER	5-116
OMBRESTORE SNAPSHOT.....	5-124
OMBREVERT.....	5-126
OMBREVOKE DEFAULT_OBJ_PRIV	5-127
OMBREVOKE OBJ_PRIV	5-129
OMBREVOKE ROLE	5-131
OMBREVOKE SYS_PRIV	5-132
OMBROLLBACK.....	5-133
OMBSAVE	5-134
OMBSHOW.....	5-135
OMBSTART.....	5-136
OMBSTATS	5-139
OMBSTOP	5-140
OMBSWITCHBACKMODE	5-141
OMBSWITCHMODE.....	5-142
OMBSYNCHRONIZE.....	5-143
OMBTRANSLATE EXTRACT.....	5-145
OMBTRANSLATE MERGE.....	5-148
OMBUNDEFINE ASSOCIATION_DEFINITION	5-151
OMBUNDEFINE CLASS_DEFINITION	5-152
OMBUNLOCK.....	5-153
OMBUNREGISTER CONTROL_CENTER	5-155
OMBUNREGISTER LOCATION	5-156
OMBUNREGISTER USER.....	5-157
OMBVALIDATE.....	5-158

6 OMBALTER to OMBALTER_EXTERNAL_TABLE

OMBALTER.....	6-2
OMBALTER ACTIVITY_TEMPLATE.....	6-5
OMBALTER ACTIVITY_TEMPLATE_FOLDER.....	6-7
OMBALTER ADVANCED_QUEUE.....	6-9
OMBALTER ALTERNATIVE_SORT_ORDER.....	6-13
OMBALTER ANALYZE_ACTION_PLAN	6-18
OMBALTER BUSINESS_AREA.....	6-21
OMBALTER BUSINESS_DEFINITION_MODULE	6-24
OMBALTER BUSINESS_PRESENTATION_MODULE.....	6-29
OMBALTER CALENDAR.....	6-33
OMBALTER CALENDAR_MODULE	6-39
OMBALTER CHANGE_DATA_CAPTURE	6-41
OMBALTER CMI_DEFINITION.....	6-46
OMBALTER CMI_MODULE.....	6-49
OMBALTER COLLECTION	6-53

OMBALTER CONFIGURATION	6-56
OMBALTER CONNECTOR.....	6-58
OMBALTER CONTROL_CENTER.....	6-61
OMBALTER CORRECTION_MAPS_ACTION_PLAN	6-67
OMBALTER CORRECTION_SCHEMA_ACTION_PLAN	6-70
OMBALTER CUBE.....	6-73
OMBALTER DATA_AUDITOR	6-81
OMBALTER DATA_PROFILE.....	6-90
OMBALTER DATA_RULE	6-102
OMBALTER DATA_RULE_MODULE.....	6-112
OMBALTER DEPLOYMENT.....	6-115
OMBALTER DEPLOYMENT_ACTION_PLAN	6-117
OMBALTER DIMENSION	6-121
OMBALTER DRILL_PATH.....	6-136
OMBALTER DRILL_TO_DETAIL.....	6-140
OMBALTER EXPERT	6-143
OMBALTER EXPERT_MODULE.....	6-175
OMBALTER EXTERNAL_TABLE.....	6-177

7 OMBALTER FLAT_FILE to OMBALTER STREAMS_QUEUE

OMBALTER FLAT_FILE.....	7-2
OMBALTER FLAT_FILE_MODULE	7-12
OMBALTER FUNCTION	7-16
OMBALTER GATEWAY_MODULE.....	7-22
OMBALTER ICONSET	7-25
OMBALTER IMPORT_ACTION_PLAN.....	7-28
OMBALTER ITEM_FOLDER	7-31
OMBALTER LIST_OF_VALUES.....	7-45
OMBALTER LOCATION	7-49
OMBALTER MAPPING	7-58
OMBALTER MATERIALIZED_VIEW.....	7-134
OMBALTER MDL_ACTION_PLAN.....	7-176
OMBALTER MINING_MODEL	7-179
OMBALTER NESTED_TABLE.....	7-189
OMBALTER OBJECT_TYPE.....	7-192
OMBALTER ORACLE_MODULE.....	7-196
OMBALTER PACKAGE	7-200
OMBALTER PLSQL_RECORD_TYPE.....	7-204
OMBALTER PLSQL_REF_CURSOR_TYPE.....	7-219
OMBALTER PLSQL_TABLE_TYPE	7-233
OMBALTER PLUGGABLE_MAPPING	7-247
OMBALTER PLUGGABLE_MAPPING_FOLDER	7-322
OMBALTER PRESENTATION_TEMPLATE.....	7-324
OMBALTER PROCEDURE.....	7-329
OMBALTER PROCESS_FLOW.....	7-335
OMBALTER PROCESS_FLOW_MODULE.....	7-344
OMBALTER PROCESS_FLOW_PACKAGE.....	7-346

OMBALTER PROFILE_REFERENCE	7-349
OMBALTER PROJECT	7-359
OMBALTER QUEUE_PROPAGATION	7-361
OMBALTER QUEUE_TABLE	7-366
OMBALTER REAL_TIME_MAPPING	7-369
OMBALTER REGISTERED_FUNCTION	7-445
OMBALTER ROLE.....	7-451
OMBALTER SAP_MODULE.....	7-453
OMBALTER SEQUENCE	7-457
OMBALTER SNAPSHOT.....	7-460
OMBALTER STREAMS_CAPTURE_PROCESS	7-463
OMBALTER STREAMS_QUEUE	7-468

8 OMBALTER TABLE to OMBALTER VIEW

OMBALTER TABLE.....	8-2
OMBALTER TABLE_FUNCTION	8-42
OMBALTER TIME_DIMENSION.....	8-48
OMBALTER TRANSPORTABLE_MODULE	8-57
OMBALTER USER.....	8-65
OMBALTER VARYING_ARRAY	8-73
OMBALTER VIEW.....	8-76

9 OMBCREATE to OMBCREATE PLSQL_TABLE_TYPE

OMBCREATE.....	9-2
OMBCREATE ACTIVITY_TEMPLATE.....	9-5
OMBCREATE ACTIVITY_TEMPLATE_FOLDER	9-7
OMBCREATE ADVANCED_QUEUE	9-9
OMBCREATE ALTERNATIVE_SORT_ORDER.....	9-13
OMBCREATE ANALYZE_ACTION_PLAN	9-17
OMBCREATE BUSINESS_AREA.....	9-19
OMBCREATE BUSINESS_DEFINITION_MODULE	9-21
OMBCREATE BUSINESS_PRESENTATION_MODULE.....	9-25
OMBCREATE CALENDAR	9-28
OMBCREATE CALENDAR_MODULE.....	9-34
OMBCREATE CHANGE_DATA_CAPTURE.....	9-36
OMBCREATE CMI_DEFINITION	9-41
OMBCREATE CMI_MODULE.....	9-43
OMBCREATE COLLECTION	9-46
OMBCREATE CONFIGURATION.....	9-49
OMBCREATE CONNECTOR.....	9-51
OMBCREATE CONTROL_CENTER	9-54
OMBCREATE CORRECTION_MAPS_ACTION_PLAN	9-59
OMBCREATE CORRECTION_SCHEMA_ACTION_PLAN	9-63
OMBCREATE CUBE.....	9-66
OMBCREATE DATA_AUDITOR	9-74
OMBCREATE DATA_PROFILE	9-82
OMBCREATE DATA_RULE	9-89

OMBCREATE DATA_RULE_MODULE.....	9-93
OMBCREATE DEPLOYMENT.....	9-95
OMBCREATE DEPLOYMENT_ACTION_PLAN.....	9-97
OMBCREATE DIMENSION.....	9-100
OMBCREATE DRILL_PATH.....	9-114
OMBCREATE DRILL_TO_DETAIL.....	9-117
OMBCREATE EXPERT.....	9-120
OMBCREATE EXPERT_MODULE.....	9-151
OMBCREATE EXTERNAL_TABLE.....	9-153
OMBCREATE FLAT_FILE.....	9-162
OMBCREATE FLAT_FILE_MODULE.....	9-172
OMBCREATE FUNCTION.....	9-175
OMBCREATE GATEWAY_MODULE.....	9-180
OMBCREATE ICONSET.....	9-183
OMBCREATE IMPORT_ACTION_PLAN.....	9-186
OMBCREATE ITEM_FOLDER.....	9-189
OMBCREATE LIST_OF_VALUES.....	9-201
OMBCREATE LOCATION.....	9-205
OMBCREATE MAPPING.....	9-214
OMBCREATE MATERIALIZED_VIEW.....	9-289
OMBCREATE MDL_ACTION_PLAN.....	9-328
OMBCREATE MINING_MODEL.....	9-331
OMBCREATE NESTED_TABLE.....	9-340
OMBCREATE OBJECT_TYPE.....	9-343
OMBCREATE ORACLE_MODULE.....	9-346
OMBCREATE PACKAGE.....	9-355
OMBCREATE PLSQL_RECORD_TYPE.....	9-358
OMBCREATE PLSQL_REF_CURSOR_TYPE.....	9-373
OMBCREATE PLSQL_TABLE_TYPE.....	9-376

10 OMBCREATE PLUGGABLE_MAPPING to OMBCREATE VIEW

OMBCREATE PLUGGABLE_MAPPING.....	10-2
OMBCREATE PLUGGABLE_MAPPING_FOLDER.....	10-76
OMBCREATE PRESENTATION_TEMPLATE.....	10-78
OMBCREATE PROCEDURE.....	10-82
OMBCREATE PROCESS_FLOW.....	10-87
OMBCREATE PROCESS_FLOW_MODULE.....	10-95
OMBCREATE PROCESS_FLOW_PACKAGE.....	10-97
OMBCREATE PROJECT.....	10-100
OMBCREATE QUEUE_PROPAGATION.....	10-102
OMBCREATE QUEUE_TABLE.....	10-107
OMBCREATE REAL_TIME_MAPPING.....	10-110
OMBCREATE REGISTERED_FUNCTION.....	10-185
OMBCREATE ROLE.....	10-190
OMBCREATE SAP_MODULE.....	10-192
OMBCREATE SEQUENCE.....	10-195
OMBCREATE SNAPSHOT.....	10-198

OMBCREATE STREAMS_CAPTURE_PROCESS.....	10-201
OMBCREATE STREAMS_QUEUE	10-206
OMBCREATE TABLE.....	10-210
OMBCREATE TABLE_FUNCTION	10-247
OMBCREATE TIME_DIMENSION.....	10-252
OMBCREATE TRANSPORTABLE_MODULE	10-259
OMBCREATE VARYING_ARRAY	10-263
OMBCREATE VIEW.....	10-266

11 OMBRETRIEVE to OMBRETRIEVE LOCATION

OMBRETRIEVE	11-2
OMBRETRIEVE ACTIVITY_TEMPLATE	11-4
OMBRETRIEVE ACTIVITY_TEMPLATE_FOLDER.....	11-5
OMBRETRIEVE ADVANCED_QUEUE.....	11-6
OMBRETRIEVE ALTERNATIVE_SORT_ORDER	11-10
OMBRETRIEVE ANALYZE_ACTION_PLAN.....	11-14
OMBRETRIEVE BUSINESS_AREA	11-16
OMBRETRIEVE BUSINESS_DEFINITION_MODULE	11-18
OMBRETRIEVE BUSINESS_PRESENTATION_MODULE	11-21
OMBRETRIEVE CALENDAR.....	11-24
OMBRETRIEVE CALENDAR_MODULE	11-25
OMBRETRIEVE CHANGE_DATA_CAPTURE	11-26
OMBRETRIEVE CMI_DEFINITION.....	11-27
OMBRETRIEVE CMI_MODULE	11-29
OMBRETRIEVE COLLECTION	11-32
OMBRETRIEVE CONFIGURATION	11-34
OMBRETRIEVE CONNECTOR	11-36
OMBRETRIEVE CONTROL_CENTER.....	11-39
OMBRETRIEVE CORRECTION_MAPS_ACTION_PLAN	11-44
OMBRETRIEVE CORRECTION_SCHEMA_ACTION_PLAN.....	11-46
OMBRETRIEVE CUBE	11-48
OMBRETRIEVE DATA_AUDITOR.....	11-49
OMBRETRIEVE DATA_PROFILE.....	11-56
OMBRETRIEVE DATA_RULE.....	11-63
OMBRETRIEVE DATA_RULE_MODULE	11-69
OMBRETRIEVE DEPLOYMENT	11-70
OMBRETRIEVE DEPLOYMENT_ACTION_PLAN	11-71
OMBRETRIEVE DIMENSION	11-73
OMBRETRIEVE DRILL_PATH.....	11-74
OMBRETRIEVE DRILL_TO_DETAIL.....	11-77
OMBRETRIEVE EXPERT	11-79
OMBRETRIEVE EXPERT_MODULE	11-88
OMBRETRIEVE EXTERNAL_TABLE	11-90
OMBRETRIEVE FLAT_FILE.....	11-98
OMBRETRIEVE FLAT_FILE_MODULE.....	11-108
OMBRETRIEVE FUNCTION	11-110
OMBRETRIEVE GATEWAY_MODULE.....	11-116

OMBRETRIEVE ICONSET	11-118
OMBRETRIEVE IMPORT_ACTION_PLAN.....	11-120
OMBRETRIEVE ITEM_FOLDER	11-122
OMBRETRIEVE LIST_OF_VALUES.....	11-133
OMBRETRIEVE LOCATION	11-137

12 OMBRETRIEVE MAPPING to OMBRETRIEVE VIEW

OMBRETRIEVE MAPPING	12-2
OMBRETRIEVE MATERIALIZED_VIEW	12-75
OMBRETRIEVE MDL_ACTION_PLAN	12-110
OMBRETRIEVE MINING_MODEL.....	12-111
OMBRETRIEVE NESTED_TABLE.....	12-118
OMBRETRIEVE OBJECT_TYPE.....	12-121
OMBRETRIEVE ORACLE_MODULE.....	12-124
OMBRETRIEVE PACKAGE	12-133
OMBRETRIEVE PLSQL_RECORD_TYPE	12-136
OMBRETRIEVE PLSQL_REF_CURSOR_TYPE	12-150
OMBRETRIEVE PLSQL_TABLE_TYPE.....	12-164
OMBRETRIEVE PLUGGABLE_MAPPING.....	12-178
OMBRETRIEVE PLUGGABLE_MAPPING_FOLDER.....	12-251
OMBRETRIEVE PRESENTATION_TEMPLATE.....	12-253
OMBRETRIEVE PROCEDURE.....	12-257
OMBRETRIEVE PROCESS_FLOW.....	12-262
OMBRETRIEVE PROCESS_FLOW_MODULE.....	12-266
OMBRETRIEVE PROCESS_FLOW_PACKAGE	12-268
OMBRETRIEVE PROFILE_REFERENCE	12-270
OMBRETRIEVE PROJECT	12-273
OMBRETRIEVE QUEUE_PROPAGATION	12-275
OMBRETRIEVE QUEUE_TABLE.....	12-280
OMBRETRIEVE REAL_TIME_MAPPING.....	12-283
OMBRETRIEVE REGISTERED_FUNCTION.....	12-356
OMBRETRIEVE ROLE	12-361
OMBRETRIEVE SAP_MODULE.....	12-362
OMBRETRIEVE SEQUENCE.....	12-364
OMBRETRIEVE SNAPSHOT	12-367
OMBRETRIEVE STREAMS_CAPTURE_PROCESS	12-369
OMBRETRIEVE STREAMS_QUEUE.....	12-373
OMBRETRIEVE TABLE	12-377
OMBRETRIEVE TABLE_FUNCTION	12-412
OMBRETRIEVE TIME_DIMENSION	12-417
OMBRETRIEVE TRANSPORTABLE_MODULE.....	12-423
OMBRETRIEVE USER.....	12-427
OMBRETRIEVE VARYING_ARRAY.....	12-435
OMBRETRIEVE VIEW	12-438

13 OMBDROP

OMBDROP	13-2
OMBDROP ACTIVITY_TEMPLATE	13-3
OMBDROP ACTIVITY_TEMPLATE_FOLDER	13-4
OMBDROP ADVANCED_QUEUE	13-5
OMBDROP ALTERNATIVE_SORT_ORDER	13-6
OMBDROP ANALYZE_ACTION_PLAN	13-7
OMBDROP BUSINESS_AREA	13-8
OMBDROP BUSINESS_DEFINITION_MODULE	13-9
OMBDROP BUSINESS_PRESENTATION_MODULE	13-10
OMBDROP CALENDAR	13-11
OMBDROP CALENDAR_MODULE	13-12
OMBDROP CHANGE_DATA_CAPTURE	13-13
OMBDROP CMI_DEFINITION	13-14
OMBDROP CMI_MODULE	13-15
OMBDROP COLLECTION	13-16
OMBDROP CONFIGURATION	13-17
OMBDROP CONNECTOR	13-18
OMBDROP CONTROL_CENTER	13-19
OMBDROP CORRECTION_MAPS_ACTION_PLAN	13-20
OMBDROP CORRECTION_SCHEMA_ACTION_PLAN	13-21
OMBDROP CUBE	13-22
OMBDROP DATA_AUDITOR	13-23
OMBDROP DATA_PROFILE	13-24
OMBDROP DATA_RULE	13-25
OMBDROP DEPLOYMENT	13-26
OMBDROP DEPLOYMENT_ACTION_PLAN	13-27
OMBDROP DIMENSION	13-28
OMBDROP DRILL_PATH	13-29
OMBDROP DRILL_TO_DETAIL	13-30
OMBDROP EXPERT	13-31
OMBDROP EXPERT_MODULE	13-32
OMBDROP EXTERNAL_TABLE	13-33
OMBDROP FLAT_FILE	13-34
OMBDROP FLAT_FILE_MODULE	13-35
OMBDROP FUNCTION	13-36
OMBDROP GATEWAY_MODULE	13-37
OMBDROP ICONSET	13-38
OMBDROP IMPORT_ACTION_PLAN	13-39
OMBDROP ITEM_FOLDER	13-40
OMBDROP LIST_OF_VALUES	13-41
OMBDROP LOCATION	13-42
OMBDROP MAPPING	13-43
OMBDROP MATERIALIZED_VIEW	13-44
OMBDROP MDL_ACTION_PLAN	13-45
OMBDROP MINING_MODEL	13-46
OMBDROP NESTED_TABLE	13-47

OMBDROP OBJECT_TYPE	13-48
OMBDROP ORACLE_MODULE	13-49
OMBDROP PACKAGE	13-50
OMBDROP PLSQL_RECORD_TYPE	13-51
OMBDROP PLSQL_REF_CURSOR_TYPE	13-52
OMBDROP PLSQL_TABLE_TYPE	13-53
OMBDROP PLUGGABLE_MAPPING	13-54
OMBDROP PLUGGABLE_MAPPING_FOLDER	13-55
OMBDROP PRESENTATION_TEMPLATE	13-56
OMBDROP PROCEDURE	13-57
OMBDROP PROCESS_FLOW	13-58
OMBDROP PROCESS_FLOW_MODULE	13-59
OMBDROP PROCESS_FLOW_PACKAGE	13-60
OMBDROP PROJECT	13-61
OMBDROP QUEUE_PROPAGATION	13-62
OMBDROP QUEUE_TABLE	13-63
OMBDROP REAL_TIME_MAPPING	13-64
OMBDROP REGISTERED_FUNCTION	13-65
OMBDROP ROLE	13-66
OMBDROP SAP_MODULE	13-67
OMBDROP SEQUENCE	13-68
OMBDROP SNAPSHOT	13-69
OMBDROP STREAMS_CAPTURE_PROCESS	13-70
OMBDROP STREAMS_QUEUE	13-71
OMBDROP TABLE	13-72
OMBDROP TABLE_FUNCTION	13-73
OMBDROP TIME_DIMENSION	13-74
OMBDROP TRANSPORTABLE_MODULE	13-75
OMBDROP VARYING_ARRAY	13-76
OMBDROP VIEW	13-77

14 OMU Commands

OMUALTER	14-2
OMUANALYZEIMPACT	14-3
OMUANALYZELINEAGE	14-4
OMUCOMPILE	14-5
OMUCONFIGURE	14-6
OMUCONNECT	14-7
OMUCONTROLCENTER	14-8
OMUCONTROLCENTERJOBS	14-9
OMUCREATE	14-10
OMUDATAVIEWER	14-12
OMUDEPLOY	14-13
OMUDERIVE	14-14
OMUEXPORT MDL_FILE	14-15
OMUIMPACT	14-16
OMUIMPORT	14-17

OMUIMPORT MDL_FILE	14-18
OMULINEAGE.....	14-19
OMULIST	14-20
OMUPROMPT	14-21
OMUPROPAGATECHANGE.....	14-31
OMUPROPERTIES	14-32
OMUSELECTSOURCE.....	14-33
OMUSELECTTARGET	14-34
OMUSHOWCHANGECONSOLE.....	14-35
OMUSHOWLIA	14-36
OMUSTART	14-37
OMUSTARTJOB.....	14-39
OMUSTART EXPERT	14-40
OMUVALIDATE.....	14-42

A Additional and Optional Usages

Using Control Files to Import and Export Metadata	A-1
Creating MDL Control Files	A-1
Exporting Metadata Using OMB Plus.....	A-8
Importing Metadata Using OMB Plus	A-8
Accessing Transformation Modules Using OMBPlus	A-9
Working with Mappings and Operators.....	A-11
Defining Expressions in Mappings	A-12
Default Group Names and Attribute Names.....	A-12
Accessing Transformation Modules	A-14
Predefined Transformations.....	A-14
Custom Transformations	A-15
Running OMB Plus in Oracle JDeveloper	A-16
Installing OMB Plus in Oracle JDeveloper	A-17
Opening the Syntax Highlighting Editor in JDeveloper	A-17
Invoking Keyword Auto Completion	A-17
Invoking the OMBPlus Interpreter	A-17
Viewing the OMBPlus Console.....	A-18
Viewing Help Documentation	A-18

Preface

This preface includes the following topics:

- [Audience](#) on page xv
- [Conventions](#) on page xvii
- [Related Documents](#) on page xvi
- [Documentation Accessibility](#) on page xvi

Purpose

Oracle Warehouse Builder is a comprehensive toolset for practitioners who move and transform data, develop and implement business intelligence systems, perform metadata management, or create and manage Oracle databases and metadata. This guide describes how to use Warehouse Builder Oracle MetaBase (OMB) Scripting Language to:

- Create a definition of a data warehouse.
- Configure the definitions for a physical instance of the data warehouse.
- Validate the set of definitions and their configurations.
- Generate a set of scripts to create and populate the data warehouse instance.
- Generate data transformation scripts.
- Deploy and initially load the data warehouse instance.
- Maintain the physical instance by conditionally refreshing it with generated scripts.
- Integrate Warehouse Builder metadata with other Business Intelligence products.
- Populate Oracle Discoverer EULs and OLAP catalogs for analyzing the data warehouse.

Audience

This guide is intended for data warehouse practitioners who want to access Warehouse Builder functionality programmatically:

- Business Intelligence application developers
- Warehouse architects, designers, and developers—especially SQL and PL/SQL developers
- Developers of large-scale products based on data warehouses

- Warehouse administrators
- System administrators
- Other MIS professionals

In order to use the information in this guide, you need to be comfortable with the concepts of Relational Database Management Systems and Data Warehouse design. For information on data warehousing, refer to the Oracle Database *Data Warehousing Guide*. Also, you need to be familiar with Oracle's relational database software products such as Oracle Database, SQL*Plus, SQL*Loader, Oracle Enterprise Manager, and Oracle Workflow.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documents

The Warehouse Builder documentation set includes these manuals:

- Oracle Warehouse Builder User's Guide
- Oracle Warehouse Builder Installation and Configuration Guide
- Oracle Warehouse Builder Transformation Guide
- Oracle Warehouse Builder Release Notes

In addition to the Warehouse Builder documentation, you can refer to other documents listed below:

- Oracle Database *Data Warehousing Guide*

Oracle provides additional information sources, including other documentation, training, and support services that can enhance your understanding and knowledge of Oracle Warehouse Builder.

- For more information on Oracle Warehouse Builder technical support, contact Oracle World Wide Support services at:

<http://www.oracle.com/support>

- For the latest information on, and downloads of, software and documentation updates to Oracle Warehouse Builder, visit MetaLink at:

<http://metalink.oracle.com>

- You can order other Oracle documentation at:

<http://oraclestore.oracle.com>

Conventions

In this manual, Windows refers to the Windows NT, Windows 2000, and Windows XP operating systems. The SQL*Plus interface to Oracle Database may be referred to as SQL.

In the examples, an implied carriage return occurs at the end of each line, unless otherwise noted. You must press the Return key at the end of a line of input.

The following conventions are also used in this manual:

Convention	Meaning
.	Vertical ellipsis points in an example mean that information not directly related to the example has been omitted.
...	Horizontal ellipsis points in statements or commands mean that parts of the statement or command not directly related to the example have been omitted.
boldface text	Boldface type in text refers to interface buttons and links. Boldface type also serves as emphasis to set apart main ideas.
<i>italicized text</i>	Italicized text applies to new terms introduced for the first time. Italicized text also serves as an emphasis on key concepts.
unicode text	Unicode text denotes exact code, file directories and names, and literal commands.
<i>italicized unicode text</i>	Italicized unicode text refers to parameters whose value is specified by the user.
[]	Brackets enclose optional clauses from which you can choose one or none.

Part I

Application Programming Interfaces

This part contains the following chapters:

- [Chapter 1, "Public Views for the Runtime Environment"](#)
- [Chapter 2, "Public Views for the Design Environment"](#)
- [Chapter 3, "Using SQL*Plus to Schedule and Execute Jobs"](#)

Public Views for the Runtime Environment

The Warehouse Builder provides a set of pre-built views for both the design and runtime environments. These views are called the Warehouse Builder Public Views and are the API alternative to using the Repository Browser described in the *Oracle Warehouse Builder User's Guide*.

Use these views to access to metadata and data stored in Warehouse Builder repositories. This chapter contains a catalog of the Public Views for the runtime environment.

- [Deployment Auditing Views](#) on page 1-2
- [Execution Auditing Views](#) on page 1-8

Warehouse Builder Runtime Repository Public Views

The Runtime Repository contains all of the deployment and execution audit data. Use these Public Views to access this data. These views are used by Runtime Audit Browser to provide audit reporting.

Deployment Auditing Views

- [ALL_RT_AUDIT_LOCATIONS](#) on page 1-2
- [ALL_RT_AUDIT_LOCATION_MESSAGES](#) on page 1-3
- [ALL_RT_AUDIT_LOCATION_FILES](#) on page 1-3
- [ALL_RT_AUDIT_OBJECTS](#) on page 1-3
- [ALL_RT_AUDIT_SCRIPT_MESSAGES](#) on page 1-4
- [ALL_RT_AUDIT_SCRIPT_RUNS](#) on page 1-4
- [ALL_RT_AUDIT_SCRIPT_FILES](#) on page 1-5
- [ALL_RT_AUDIT_DEPLOYMENTS](#) on page 1-5
- [ALL_RT_INSTALLATIONS](#) on page 1-6
- [ALL_RT_LOCATIONS](#) on page 1-6
- [ALL_RT_LOCATION_PARAMETERS](#) on page 1-6
- [ALL_RT_OBJECTS](#) on page 1-6
- [ALL_RT_TASKS](#) on page 1-7
- [ALL_RT_TASK_PARAMETERS](#) on page 1-7

Execution Auditing Views

- [ALL_RT_AUDIT_EXECUTIONS](#) on page 1-8
- [ALL_RT_AUDIT_EXECUTION_PARAMS](#) on page 1-9
- [ALL_RT_AUDIT_EXEC_MESSAGES](#) on page 1-9
- [ALL_RT_AUDIT_EXEC_FILES](#) on page 1-10
- [ALL_RT_AUDIT_MAP_RUNS](#) on page 1-10
- [ALL_RT_AUDIT_MAP_RUN_SOURCES](#) on page 1-11
- [ALL_RT_AUDIT_MAP_RUN_TARGETS](#) on page 1-11
- [ALL_RT_AUDIT_STEP_RUNS](#) on page 1-11
- [ALL_RT_AUDIT_STEP_RUN_SOURCES](#) on page 1-12
- [ALL_RT_AUDIT_STEP_RUN_TARGETS](#) on page 1-12
- [ALL_RT_AUDIT_MAP_RUN_ERRORS](#) on page 1-12
- [ALL_RT_AUDIT_MAP_RUN_TRACE](#) on page 1-13
- [ALL_RT_AUDIT_PROC_RUN_ERRORS](#) on page 1-13
- [ALL_RT_AUDIT_STEP_RUN_STRUCTS](#) on page 1-13

Deployment Auditing Views

Table 1–1 *ALL_RT_AUDIT_LOCATIONS*

Column Name	Data Type	Description
LOCATION_AUDIT_ID	NUMBER (22)	Internal primary key to audit_location
RUNTIME_VERSION	VARCHAR2 (64)	Runtime version number
CLIENT_VERSION	VARCHAR2 (64)	Design client version number
CLIENT_REPOSITORY	VARCHAR2 (30)	Name of the client repository
CLIENT_REPOSITORY_VERSION	VARCHAR2 (64)	Client repository version number
REPOSITORY_USER	VARCHAR2 (30)	Username of the design repository
GENERATION_TIME	DATE	When the deployment was generated
DEPLOYMENT_AUDIT_ID	NUMBER (22)	Internal audit ID of the deployment
DEPLOYMENT_SEQUENCE_NUMBER	NUMBER (10)	Sequence number of this location in the deployment
DEPLOYMENT_AUDIT_NAME	VARCHAR2 (64)	Audit name of the location
DEPLOYMENT_AUDIT_STATUS	VARCHAR2 (4000)	INACTIVE, READY, or COMPLETE
LOCATION_AUDIT_STATUS	VARCHAR2 (4000)	INACTIVE, READY, BUSY_PREPARE, BUSY_UNPREPARE, BUSY_DEPLOY, BUSY_UNDO, BUSY_FINALIZE, or COMPLETE
LOCATION_UOID	VARCHAR2 (32)	Client UOID of the location
LOCATION_NAME	VARCHAR2 (64)	Name of the location
LOCATION_TYPE	VARCHAR2 (64)	Type of the location. (ODB, OWF, OEM)
LOCATION_TYPE_VERSION	VARCHAR2 (64)	Version of the target
NUMBER_SCRIPT_RUN_ERRORS	NUMBER (10)	Number of errors detected
NUMBER_SCRIPT_RUN_WARNINGS	NUMBER (10)	Number of warnings detected

Table 1-1 (Cont.) ALL_RT_AUDIT_LOCATIONS

Column Name	Data Type	Description
CREATED_ON	DATE	The time audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1-2 ALL_RT_AUDIT_LOCATION_MESSAGES

Column Name	Data Type	Description
MESSAGE_AUDIT_ID	NUMBER (22)	Internal key to audit_location_message. Primary when used with message_line_number
LOCATION_AUDIT_ID	NUMBER (22)	Internal key to audit_location
MESSAGE_SEVERITY	VARCHAR2 (4000)	INFORMATIONAL, WARNING, ERROR, or RECOVERY
MESSAGE_LINE_NUMBER	NUMBER (10)	1 for single line messages >0 for multiple line messages (Forms primary key when used with message_audit_id)
MESSAGE_TEXT	VARCHAR2 (4000)	plain_text or nls_key
CREATED_ON	DATE	The time audit data was created
CREATED_BY	VARCHAR2 (30)	Database username

Table 1-3 ALL_RT_AUDIT_LOCATION_FILES

Column Name	Data Type	Description
FILE_AUDIT_ID	NUMBER (22)	Internal primary key to audit_location_file
LOCATION_AUDIT_ID	NUMBER (22)	Internal key to audit_location
FILE_TYPE	VARCHAR2 (64)	SQLLoaderLogFile, ShellOutputStream, ShellErrorStream, FTPOutputStream, or FTPErrrorStream
FILE_TEXT	CLOB	Contents of the file
FORMAT	VARCHAR2(4000)	TEXT or HTML
CREATED_ON	DATE	The time audit data was created
CREATED_BY	VARCHAR2 (30)	Database username

Table 1-4 ALL_RT_AUDIT_OBJECTS

Column Name	Data Type	Description
OBJECT_AUDIT_ID	NUMBER (22)	Internal primary key to audit_object
PARENT_OBJECT_AUDIT_ID	NUMBER (22)	Internal key to parent audit_script_run
LOCATION_AUDIT_ID	NUMBER (22)	Internal key to audit_location
LOCATION_SEQUENCE_NUMBER	NUMBER (10)	Sequence number of this object in the location
OBJECT_UOID	VARCHAR2 (32)	UOID of the deployed object
OBJECT_NAME	VARCHAR2 (64)	Name of the deployed object

Table 1-4 (Cont.) ALL_RT_AUDIT_OBJECTS

Column Name	Data Type	Description
OBJECT_TYPE	VARCHAR2 (64)	Type of deployed object. (PLSQLMap, Table, Dimension, SQLLoaderControlFile)
CLIENT_VERSION_TAG	VARCHAR2 (80)	Client version identifier of this object
NUMBER_SCRIPT_RUN_ERRORS	NUMBER (10)	Number of errors detected
NUMBER_SCRIPT_RUN_WARNINGS	NUMBER (10)	Number of warnings detected
STATUS_WHEN_DEPLOYED	VARCHAR2 (4000)	VALID, INVALID, REMOVED, or UNCERTAIN
CREATED_ON	DATE	The time audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATE_ON	DATE	The time audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1-5 ALL_RT_AUDIT_SCRIPT_MESSAGES

Column Name	Data Type	Description
MESSAGE_AUDIT_ID	NUMBER (22)	Internal primary key to audit_script_file
SCRIPT_RUN_AUDIT_ID	NUMBER (22)	Internal key to audit_script_run
MESSAGE_SEVERITY	VARCHAR2 (4000)	INFORMATIONAL, WARNING, ERROR, or RECOVERY
MESSAGE_LINE_NUMBER	NUMBER (10)	1 for single line messages >0 for multiple line messages (Forms primary key when used with message_audit_id)
MESSAGE_TEXT	VARCHAR2 (4000)	plain_text or nls_key
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username

Table 1-6 ALL_RT_AUDIT_SCRIPT_RUNS

Column Name	Data Type	Description
SCRIPT_RUN_AUDIT_ID	NUMBER (22)	Internal primary key to audit_script_run
LOCATION_AUDIT_ID	NUMBER (22)	Internal key to audit_location
OBJECT_AUDIT_ID	NUMBER (22)	Internal key to audit_object
SCRIPT_RUN_AUDIT_STATUS	VARCHAR2 (4000)	BUSY, COMPLETE, UNCERTAIN, FAILED or INACTIVE
OPERATION	VARCHAR2 (4000)	DEPLOY, or UNDO
SCRIPT_ACTION	VARCHAR2 (4000)	CREATE, DROP, UPGRADE or REPORT
SCRIPT	CLOB	Script used to perform the action
SCRIPT_FORMAT	VARCHAR2 (4000)	TEXT or HTML
SCRIPT_GENERATION_TIME	DATE	The time the script was created
NUMBER_SCRIPT_RUN_ERRORS	NUMBER	The number of errors detected
NUMBER_SCRIPT_RUN_WARNINGS	NUMBER	The number of warnings detected
ELAPSE_TIME	NUMBER (10)	The number of seconds that elapsed

Table 1–6 (Cont.) ALL_RT_AUDIT_SCRIPT_RUNS

Column Name	Data Type	Description
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time the audit data updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1–7 ALL_RT_AUDIT_SCRIPT_FILES

Column Name	Data Type	Description
FILE_AUDIT_ID	NUMBER (22)	Internal primary key to audit_script_file
SCRIPT_RUN_AUDIT_ID	NUMBER (22)	Internal key to audit_script_run
FILE_TYPE	VARCHAR2 (64)	SQLLoaderLogFile, ShellOutputStream, ShellErrorStream, FTPOutputStream, or FTPErrorStream
FILE_TEXT	CLOB	Contents of the file
FORMAT	VARCHAR2 (4000)	TEXT or HTML
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username

Table 1–8 ALL_RT_AUDIT_DEPLOYMENTS

Column Name	Data Type	Description
DEPLOYMENT_AUDIT_ID	NUMBER(22)	ID of the deployment audit
DEPLOYMENT_AUDIT_NAME	VARCHAR2(64)	Name of the deployment audit
NUMBER_OF_UNITS	NUMBER(10)	Number of units
RUNTIME_VERSION	VARCHAR2(64)	Version of the control center
CLIENT_VERSION	VARCHAR2(64)	Version of the Warehouse Builder client
CLIENT_REPOSITORY	VARCHAR2(30)	Name of Client-Repository
CLIENT_REPOSITORY_VERSION	VARCHAR2(64)	Version of the Warehouse Builder client repository
REPOSITORY_USER	VARCHAR2(30)	Name of the repository user
GENERATION_TIME	DATE	Timestamp of the object generation
DEPLOYMENT_AUDIT_STATUS	VARCHAR2(4000)	Status of the deployment
NUMBER_SCRIPT_RUN_ERRORS	NUMBER(10)	Number of errors running the scripts within the deployment
NUMBER_SCRIPT_RUN_WARNINGS	NUMBER(10)	Number of warnings whilst running the scripts within the deployment
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 1–9 ALL_RT_INSTALLATIONS

Column Name	Data Type	Description
INSTALLATION_ID	VARCHAR2	ID of the installation
INSTALLATION_NAME	CHAR	Name of the installation
DESCRIPTION	CHAR	Description of the installation
INSTALLED_VERSION	VARCHAR2(4000)	Version of the repository that is installed
RELEASE	VARCHAR2	Release number of Warehouse Builder
PUBLIC_VIEW_VERSION	CHAR(4)	Version of the PublicViews implemented by this installation
SCHEMA_TYPE	VARCHAR2(24)	Type of schema
UPDATED_ON	VARCHAR2	Update timestamp
CREATED_ON	VARCHAR2	Creation timestamp

Table 1–10 ALL_RT_LOCATIONS

Column Name	Data Type	Description
LOCATION_NAME	VARCHAR2(64)	Name of the location
LOCATION_UUID	VARCHAR2(32)	UUID of the location
IS_DEPLOYMENT_TARGET	VARCHAR2(5)	Indicates whether this location is a deployment target
LOCATION_TYPE	VARCHAR2(64)	Type of the source or target with which the location is associated
LOCATION_TYPE_VERSION	VARCHAR2(64)	Version of the source or target

Table 1–11 ALL_RT_LOCATION_PARAMETERS

Column Name	Data Type	Description
LOCATION_NAME	VARCHAR2(64)	Name of the location
PARAMETER_NAME	VARCHAR2(64)	Name of the parameter
IS_ACCESS_RESTRICTED	VARCHAR2(5)	Indicates whether the parameter-value has a restricted value, such as for a password
PARAMETER_VALUE	VARCHAR2(4000)	The value of a non-access restricted parameter

Table 1–12 ALL_RT_OBJECTS

Column Name	Data Type	Description
OBJECT_NAME	VARCHAR2(64)	Name of the object
OBJECT_TYPE	VARCHAR2(64)	Type of the object
PARENT_OBJECT_NAME	VARCHAR2(64)	Name of the parent object
PARENT_OBJECT_TYPE	VARCHAR2(64)	Type of the parent object
LOCATION_NAME	VARCHAR2(64)	Name of the location to which this object is deployed
OBJECT_UUID	VARCHAR2(32)	UUID of the object
VERSION_TAG	VARCHAR2(80)	Version string used by Control Center Manager

Table 1–12 (Cont.) ALL_RT_OBJECTS

Column Name	Data Type	Description
LAST_DEPLOYMENT_SCRIPT	CLOB	Script that was used during this objects last deployment
STATUS_WHEN_LAST_DEPLOYED	VARCHAR2(4000)	Object status at the end of its last deployment
SUB_OBJECTS_VALID	VARCHAR2(5)	Indicates whether the subobjects of an object were valid at last deployment
RELATED_OBJECTS_VALID	VARCHAR2(5)	Indicates whether the related objects of an object were valid at last deployment
DEPLOYMENT_DATE	DATE	Date the object was deployed
DEPLOYED_BY	VARCHAR2(30)	User who deployed the object

Table 1–13 ALL_RT_TASKS

Column Name	Data Type	Description
CONTEXT_OBJECT_NAME	VARCHAR2(64)	Name of the tasks context
CONTEXT_OBJECT_TYPE	VARCHAR2(64)	Type of the tasks context
CONTEXT_LOCATION_NAME	VARCHAR2(64)	Name of the location where the context was deployed to
EXEC_LOCATION_NAME	VARCHAR2(64)	Name of the location where the task will be executed
TASK_NAME	VARCHAR2(64)	Name of the task
TASK_TYPE	VARCHAR2(64)	Type of the task
TASK_INPUT	CLOB	Script that implements the task
OBJECT_NAME	VARCHAR2(64)	Name of the object
OBJECT_TYPE	VARCHAR2(64)	Type of the object
OBJECT_LOCATION_NAME	VARCHAR2(64)	Location to which the object is deployed

Table 1–14 ALL_RT_TASK_PARAMETERS

Column Name	Data Type	Description
CONTEXT_OBJECT_NAME	VARCHAR2(64)	Name of the tasks context
CONTEXT_OBJECT_TYPE	VARCHAR2(64)	Type of the tasks context
CONTEXT_LOCATION_NAME	VARCHAR2(64)	Name of the location where the context was deployed to
EXEC_LOCATION_NAME	VARCHAR2(64)	Name of the location where the task will be executed
TASK_NAME	VARCHAR2(64)	Name of the task
PARAMETER_KIND	VARCHAR2(4000)	Kind of Parameter
CUSTOM_PARAMETER_UOID	VARCHAR2(32)	UOID of the parameter if it is a custom parameter
PARAMETER_NAME	VARCHAR2(64)	Name of the parameter
PARAMETER_TYPE	VARCHAR2(4000)	Type of the parameter
PARAMETER_MODE	VARCHAR2(4000)	Mode of the parameter (IN/OUT/INOUT/VARIABLE)
PARAMETER_SCOPE	VARCHAR2(4000)	Scope of the parameter (GLOBAL, SHARED, PARAMETER, LOCAL or INNER)
BOUND_TO_NAME	VARCHAR2(64)	Name of object that the parameter might be bound to

Table 1–14 (Cont.) ALL_RT_TASK_PARAMETERS

Column Name	Data Type	Description
BOUND_TO_KIND	VARCHAR2(4000)	Kind of object that the parameter might be bound to
BOUND_TO_SCOPE	VARCHAR2(4000)	Scope of the object that the parameter might be bound to
IS_FIXED	VARCHAR2(5)	Is parameter value fixed
IS_REQUIRED	VARCHAR2(5)	Is parameter value required
TYPE_LENGTH	NUMBER(10)	Data type length of parameter
TYPE_SCALE	NUMBER(10)	Datatype scale of parameter
TYPE_PRECISION	NUMBER(10)	Datatype precision of parameter
DEFAULT_VALUE_KIND	NUMBER(8)	Integer indicating kind of value (Literal, Expression, Evaluated Expression, Not Set)
DEFAULT_VALUE	VARCHAR2(4000)	Default value for parameter

Execution Auditing Views

Table 1–15 ALL_RT_AUDIT_EXECUTIONS

Column Name	Data Type	Description
EXECUTION_AUDIT_ID	NUMBER (22)	Internal primary key to audit_execution
PARENT_EXECUTION_AUDIT_ID	NUMBER (22)	Internal key to parent audit_execution
TOP_LEVEL_EXECUTION_AUDIT_ID	NUMBER (22)	Internal key to top-level audit_execution
EXECUTION_NAME	VARCHAR2 (64)	Name of the execution run
TASK_NAME	VARCHAR2 (64)	Name of the task executed
TASK_TYPE	VARCHAR2 (64)	Type of task executed. (PL/SQL, ProcessFlow)
TASK_INPUT	CLOB	Input stream for the task
EXEC_LOCATION_UOID	VARCHAR2 (32)	UOID of the location where execution is performed
EXEC_LOCATION_NAME	VARCHAR2 (64)	Name of the location where execution is performed
EXEC_LOCATION_TYPE	VARCHAR2 (64)	Type of the location where execution is performed. (Runtime Platform, OEM)
EXEC_LOCATION_TYPE_VERSION	VARCHAR2 (64)	Version of the location where execution is performed
OBJECT_UOID	VARCHAR2 (32)	Client UOID of mapping executed
OBJECT_NAME	VARCHAR2 (64)	Name of mapping executed
OBJECT_TYPE	VARCHAR2 (64)	Type of mapping executed
OBJECT_LOCATION_UOID	VARCHAR2 (32)	Location UOID where mapping deployed
OBJECT_LOCATION_NAME	VARCHAR2 (64)	Location name where mapping deployed
OBJECT_LOCATION_TYPE	VARCHAR2 (64)	Location type where mapping deployed
OBJECT_LOCATION_TYPE_VERSION	VARCHAR2 (64)	Location version where mapping deployed
RETURN_RESULT	VARCHAR2 (64)	FAILURE, OK, OK_WITH_WARNINGS, or OK_WITH_ERRORS

Table 1–15 (Cont.) ALL_RT_AUDIT_EXECUTIONS

Column Name	Data Type	Description
RETURN_CODE	NUMBER (10)	<0: Failure >= 0: Success
EXECUTION_AUDIT_STATUS	VARCHAR2	INACTIVE, BUSY, READY or COMPLETE
ELAPSE_TIME	NUMBER (10)	Number of seconds elapsed
NUMBER_TASK_ERRORS	NUMBER (10)	Number of errors detected
NUMBER_TASK_WARNINGS	NUMBER (10)	Number of warnings detected
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time the audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1–16 ALL_RT_AUDIT_EXECUTION_PARAMS

Column Name	Data Type	Description
PARAMETER_AUDIT_ID	NUMBER (22)	Internal primary key to audit_execution_param
EXECUTION_AUDIT_ID	NUMBER (22)	Internal key to audit_execution
CUSTOM_PARAMETER_UOID	VARCHAR2 (32)	UOID of custom parameter
PARAMETER_NAME	VARCHAR2 (64)	Name of parameter
PARAMETER_TYPE	VARCHAR2(4000)	BOOLEAN, CHAR, DATE, FLOAT, NUMBER, VARCHAR, VARCHAR2, OPERATING_MODE or AUDIT_LEVEL
PARAMETER_KIND	VARCHAR2(4000)	SYSTEM or CUSTOM
PARAMETER_MODE	VARCHAR2(4000)	IN, OUT, or INOUT
VALUE_KIND	VARCHAR2 (12)	INPUT VALUE or OUTPUT VALUE
VALUE	VARCHAR2 (4000)	Character representation of parameter value

Table 1–17 ALL_RT_AUDIT_EXEC_MESSAGES

Column Name	Data Type	Description
MESSAGE_AUDIT_ID	NUMBER (22)	Internal key to audit_exec_message. Primary key when used with message_line_number
EXECUTION_AUDIT_ID	NUMBER (22)	Internal key to audit_execution
MESSAGE_SEVERITY	VARCHAR2	INFORMATIONAL, WARNING, ERROR, or RECOVERY
MESSAGE_LINE_NUMBER	NUMBER (10)	1 for single line messages >0 for multiple line messages (Forms primary key when used with message_audit_id)
MESSAGE_TEXT	VARCHAR2 (4000)	Plain_text or nls_key
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username

Table 1–18 ALL_RT_AUDIT_EXEC_FILES

Column Name	Data Type	Description
FILE_AUDIT_ID	NUMBER (22)	Internal primary key to audit_exec_file
EXECUTION_AUDIT_ID	NUMBER (22)	Internal key to audit_execution
FILE_TYPE	VARCHAR2 (64)	Type of the file
FILE_TEXT	CLOB	Content of the file
FORMAT	VARCHAR2	TEXT or HTML
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username

Table 1–19 ALL_RT_AUDIT_MAP_RUNS

Column Name	Data Type	Description
MAP_RUN_ID	NUMBER (22)	Internal primary key to audit_map_run
EXECUTION_AUDIT_ID	NUMBER (22)	Internal key to audit_execution
MAP_UOID	VARCHAR2 (255)	UOID of the mapping
MAP_NAME	VARCHAR2 (80)	Name of the mapping
MAP_TYPE	VARCHAR2 (30)	PLSQLMap or SQLLoaderControlFile
START_TIME	DATE	The time the mapping started
END_TIME	DATE	The time the mapping ended
ELAPSE_TIME	NUMBER (10)	Number of seconds elapsed
RUN_STATUS	VARCHAR2 (8)	RUNNING, FAILURE or COMPLETE
PHYSICAL_NAME	VARCHAR2 (80)	Full hierarchic name of .dat file for a SQL*Loader run
LOAD_DATE	VARCHAR2 (30)	Load date for a SQL*Loader run
LOAD_TIME	VARCHAR2 (30)	Load time for a SQL*Loader run
NUMBER_ERRORS	NUMBER (10)	Number of errors detected
NUMBER_RECORDS_SELECTED	NUMBER (10)	Number of records selected from source tables
NUMBER_RECORDS_INSERTED	NUMBER (10)	Number of records inserted into target tables
NUMBER_RECORDS_UPDATED	NUMBER (10)	Number of records updated in target tables
NUMBER_RECORDS_DELETED	NUMBER (10)	Number of records deleted in target tables
NUMBER_RECORDS_DISCARDED	NUMBER (10)	Number of records discarded in SQL*Loader run
NUMBER_RECORDS_MERGED	NUMBER (10)	Number of records merged in target tables
NUMBER_RECORDS_CORRECTED	NUMBER (10)	Number of records corrected in target tables
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time the audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1–20 ALL_RT_AUDIT_MAP_RUN_SOURCES

Column Name	Data Type	Description
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
SOURCE_NAME	VARCHAR2 (2000)	Name of mapping operator representing source table
SOURCE_DBLINK	VARCHAR2 (2000)	Name of database link for mapping operator representing source table

Table 1–21 ALL_RT_AUDIT_MAP_RUN_TARGETS

Column Name	Data Type	Description
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
TARGET_NAME	VARCHAR2 (2000)	Name of mapping operator representing target table

Table 1–22 ALL_RT_AUDIT_STEP_RUNS

Column Name	Data Type	Description
STEP_ID	NUMBER (22)	Internal primary key to audit_step_run
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
MAP_STEP	NUMBER (22)	Step number 0 or 1 For a PL/SQL mapping, this number is normally 0 for a set-based run, and 1 for a row-based, or row-based-target run
STEP_NAME	VARCHAR2 (80)	Name of the mapping for a set-based run, or the name of a mapping object for a set-based or set-based-target run
STEP_TYPE	VARCHAR2 (18)	Set-based, Row-based or Row-based target
START_TIME	DATE	The time the mapping step started
END_TIME	DATE	The time the mapping step ended
ELAPSE_TIME	NUMBER (10)	Number of seconds taken
RUN_STATUS	VARCHAR2 (8)	RUNNING or COMPLETE
NUMBER_ERRORS	NUMBER (10)	Number of errors detected
NUMBER_RECORDS_SELECTED	NUMBER (10)	Number of records selected from source tables
NUMBER_RECORDS_INSERTED	NUMBER (10)	Number of records inserted into target tables
NUMBER_RECORDS_UPDATED	NUMBER (10)	Number of records updated in target tables
NUMBER_RECORDS_DELETED	NUMBER (10)	Number of records deleted in target tables
NUMBER_RECORDS_DISCARDED	NUMBER (10)	Number of records discarded in a SQL*Loader run
NUMBER_RECORDS_MERGED	NUMBER (10)	Number of records merged in target tables
NUMBER_RECORDS_CORRECTED	NUMBER (10)	Number of records corrected in target tables
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time the audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1–23 ALL_RT_AUDIT_STEP_RUN_SOURCES

Column Name	Data Type	Description
STEP_ID	NUMBER (22)	Internal key to audit_step_run
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
MAP_STEP	NUMBER (22)	Step number 0 or 1 For a PL/SQL mapping, this number is normally 0 for a set-based run, and 1 for a row-based, or row-based-target run
SOURCE_NAME	VARCHAR2 (2000)	Name of mapping operator representing source table
SOURCE_DBLINK	VARCHAR2 (2000)	Name of database link for mapping operator representing source table

Table 1–24 ALL_RT_AUDIT_STEP_RUN_TARGETS

Column Name	Data Type	Description
STEP_ID	NUMBER (22)	Internal key to audit_step_run
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
MAP_STEP	NUMBER (22)	Step number 0 or 1 For a PL/SQL mapping, this number is normally 0 for a set-based run, and 1 for a row-based, or row-based-target run
TARGET_NAME	VARCHAR2 (2000)	Name of mapping operator representing target table

Table 1–25 ALL_RT_AUDIT_MAP_RUN_ERRORS

Column Name	Data Type	Description
RUN_ERROR_ID	NUMBER (22)	Internal primary key for map_run_error
STEP_ID	NUMBER (22)	Internal key to audit_step_run
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
MAP_STEP	NUMBER (22)	Step number 0 or 1 For a PL/SQL mapping, this number is normally 0 for a set-based run, and 1 for a row-based, or row-based-target run
CURSOR_ROWKEY	NUMBER (22)	Value identifying row returned by cursor. This is 0 for errors in a set-based run
RUN_ERROR_NUMBER	NUMBER (10)	Message number
RUN_ERROR_MESSAGE	VARCHAR2 (2000)	Message text
TARGET_NAME	VARCHAR2 (80)	Name of mapping operator representing target table
TARGET_COLUMN	VARCHAR2 (80)	Column name, or '*' if not known or not applicable
STATEMENT	VARCHAR2 (2000)	Value such as INSERT or BATCH INSERT, or a PL/SQL statement
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time the audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1–26 ALL_RT_AUDIT_MAP_RUN_TRACE

Column Name	Data Type	Description
TRACE_ID	NUMBER (22)	Internal primary key for map_run_trace
MAP_RUN_ID	NUMBER (22)	Internal key to audit_map_run
MAP_STEP	NUMBER (22)	Step number 0 or 1 For a PL/SQL mapping, this number is normally 0 for a set-based run, and 1 for a row-based, or row-based-target run
CURSOR_ROWKEY	NUMBER (22)	Value identifying error row returned by cursor. This is 0 for set-based run
TYPE	VARCHAR2 (30)	NEW for trace or ERROR for error
ROLE	VARCHAR2 (30)	S for source or T for target
ACTION	VARCHAR2 (30)	Value such as SELECT or a PL/SQL statement
TABLE_NAME	VARCHAR2 (80)	Name of mapping operator representing source/target table
CREATED_ON	DATE	The time the audit data was created
CREATED_BY	VARCHAR2 (30)	Database username
UPDATED_ON	DATE	The time the audit data was updated
UPDATED_BY	VARCHAR2 (30)	Database username

Table 1–27 ALL_RT_AUDIT_PROC_RUN_ERRORS

Column Name	Data Type	Description
RUN_ERROR_ID	NUMBER(22)	ID of the run error
MAP_RUN_ID	NUMBER(22)	ID of the map run
CURSOR_ROWKEY	NUMBER(22)	Rowkey of record returned by cursor when error reported
RUN_ERROR_NUMBER	NUMBER(10)	Error number
RUN_ERROR_MESSAGE	VARCHAR2(2000)	Error message
TARGET_NAME	VARCHAR2(2000)	Name of target
TARGET_COLUMN	VARCHAR2(80)	Name of target column
STATEMENT	VARCHAR2(2000)	Statement when error is reported
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 1–28 ALL_RT_AUDIT_STEP_RUN_STRUCTS

Column Name	Data Type	Description
STRUCT_ID	NUMBER(22)	ID of the structure
STEP_ID	NUMBER(22)	ID of the step in the map
MAP_RUN_ID	NUMBER(22)	ID of the run of the map
PARENT_OPERATOR_UOID	VARCHAR2(32)	UOID of the map-operator being audited

Table 1–28 (Cont.) ALL_RT_AUDIT_STEP_RUN_STRUCTS

Column Name	Data Type	Description
PARENT_OBJECT_UOID	VARCHAR2(32)	UUID of the parent object that the object is related to
PARENT_OBJECT_TYPE	VARCHAR2(30)	Type of the parent object
PARENT_OBJECT_LOCATION_UOID	VARCHAR2(32)	UUID of the location where the parent object has been deployed
PARENT_OBJECT_NAME	VARCHAR2(80)	Name of the parent object
OBJECT_UOID	VARCHAR2(32)	UUID of the object
OBJECT_TYPE	VARCHAR2(30)	Type of the object
OBJECT_LOCATION_UOID	VARCHAR2(32)	UUID of the location where the object has been deployed
OBJECT_NAME	VARCHAR2(80)	Name of the object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Public Views for the Design Environment

The Warehouse Builder provides a set of pre-built views for both the design and runtime environments. These views are called the Warehouse Builder Public Views and are the API alternative to using the Repository Browser described in the *Oracle Warehouse Builder User's Guide*.

Use these views to access to metadata and data stored in Warehouse Builder repositories. This chapter contains a catalog of the Public Views to the design environment.

- [General Model Views](#)
- [Data Model Views](#)
- [Flat Files Views](#)
- [Collections Views](#)
- [Function Model Views](#)
- [Configuration Model Views](#)
- [Deployment Model Views](#)
- [Mapping Model Views](#)
- [Process Flow Model Views](#)
- [Data Profiling Views](#)
- [Data Rules Views](#)
- [User Defined Object Views](#)
- [Expert Views](#)
- [Business Intelligence Views](#)
- [Real Time Views](#)
- [Scheduling Views](#)
- [Others](#)

Warehouse Builder Design Repository Public Views

The design repository contains all of the design metadata. Use these Public Views to access data about the design of your system. These views are used by Warehouse Builder Browser to provide metadata reporting.

[General Model Views](#)

- [ALL_IV_ALL_OBJECTS](#) on page 2-7
- [ALL_IV_OBJECTS](#) on page 2-7
- [ALL_IV_OBJECT_PROPERTIES](#) on page 2-8
- [ALL_IV_MLS_OBJECTS](#) on page 2-8
- [ALL_IV_SUPPORTED_LANGUAGES](#) on page 2-8
- [ALL_IV_MODULES](#) on page 2-8
- [ALL_IV_PROJECTS](#) on page 2-9
- [ALL_IV_INFORMATION_SYSTEMS](#) on page 2-9
- [ALL_IV_INSTALLATIONS](#) on page 2-10
- [ALL_IV_FILE_MODULES](#) on page 2-10
- [ALL_IV_GATEWAY_MODULES](#) on page 2-11
- [ALL_IV_PACKAGED_APPS_MODULES](#) on page 2-12
- [ALL_IV_PREDEFINED_MODULES](#) on page 2-13
- [ALL_IV_PROCESS_MODULES](#) on page 2-13
- [ALL_IV_WAREHOUSE_MODULES](#) on page 2-14
- [ALL_IV_BUSINESS_DEF_MODULES](#) on page 2-15
- [ALL_IV_BUSINESS_PRES_MODULES](#) on page 2-15
- [ALL_IV_CALENDAR_MODULES](#) on page 2-16
- [ALL_IV_CMIV_DEFINITIONS](#) on page 2-16
- [ALL_IV_CMIV_MODULES](#) on page 2-17
- [ALL_IV_DATA_RULE_MODULES](#) on page 2-18
- [ALL_IV_EXPERT_MODULES](#) on page 2-18
- [ALL_IV_PF_CORRECTED_MODULES](#) on page 2-19
- [ALL_IV_SAP_MODULES](#) on page 2-19
- [ALL_IV_TM_MODULES](#) on page 2-20
- [ALL_IV_UDO_MODULES](#) on page 2-20
- [ALL_IV_CMIV_VIEWS](#) on page 2-21
- [ALL_IV_FIRSTCLASS_OBJECTS](#) on page 2-21
- [ALL_IV_DB_FUNCTIONS](#) on page 2-21

Data Model Views

- [ALL_IV_ADVANCED_QUEUES](#) on page 2-22
- [ALL_IV_ATTR_GROUPS](#) on page 2-22
- [ALL_IV_ATTR_GROUP_ITEM_USES](#) on page 2-23
- [ALL_IV_CHECK_CONSTRAINTS](#) on page 2-23
- [ALL_IV_COLUMNS](#) on page 2-23
- [ALL_IV_CONSTRAINTS](#) on page 2-24
- [ALL_IV_CUBES](#) on page 2-24

- [ALL_IV_CUBE_DIMENSIONS](#) on page 2-25
- [ALL_IV_CUBE_MEASURES](#) on page 2-25
- [ALL_IV_CUBE_MEASURE_DIM_USES](#) on page 2-26
- [ALL_IV_DIMENSIONS](#) on page 2-26
- [ALL_IV_DIM_HIERARCHIES](#) on page 2-27
- [ALL_IV_DIM_HIERARCHY_LEVELS](#) on page 2-28
- [ALL_IV_DIM_LEVELS](#) on page 2-28
- [ALL_IV_DIM_LEVEL_ATTRIBUTES](#) on page 2-28
- [ALL_IV_EXTERNAL_COLUMNS](#) on page 2-29
- [ALL_IV_EXTERNAL_TABLES](#) on page 2-30
- [ALL_IV_FOREIGN_KEYS](#) on page 2-30
- [ALL_IV_KEYS](#) on page 2-31
- [ALL_IV_KEY_COLUMN_USES](#) on page 2-31
- [ALL_IV_MATERIALIZED_VIEWS](#) on page 2-31
- [ALL_IV_OBJECT_TYPES](#) on page 2-32
- [ALL_IV_RECORD_FIELDS](#) on page 2-32
- [ALL_IV_RELATIONS](#) on page 2-33
- [ALL_IV_SEQUENCES](#) on page 2-33
- [ALL_IV_VIEWS](#) on page 2-34
- [ALL_IV_TABLES](#) on page 2-34
- [ALL_IV_CALENDARS](#) on page 2-34
- [ALL_IV_VARRAYS](#) on page 2-35
- [ALL_IV_SCHEMAS](#) on page 2-35
- [ALL_IV_PROCEDURES](#) on page 2-36
- [ALL_IV_REF_CURSORS](#) on page 2-36
- [ALL_IV_DIM_ATTRIBUTES](#) on page 2-36
- [ALL_IV_DIM_ROLES](#) on page 2-37
- [ALL_IV_TM_SCHEMAS](#) on page 2-37
- [ALL_IV_TM_TABLESPACES](#) on page 2-38
- [ALL_IV_CUBE_IMPLS](#) on page 2-38
- [ALL_IV_DIM_IMPLS](#) on page 2-39
- [ALL_IV_DIM_LEVEL_IMPLS](#) on page 2-39
- [ALL_IV_NESTED_TABLES](#) on page 2-40

Flat Files Views

- [ALL_IV_FIELDS](#) on page 2-40
- [ALL_IV_FILES](#) on page 2-41
- [ALL_IV_RECORDS](#) on page 2-42

Collections Views

- [ALL_IV_COLLECTIONS](#) on page 2-42
- [ALL_IV_COLLECTION_REFERENCES](#) on page 2-43

Function Model Views

- [ALL_IV_FUNCTIONS](#) on page 2-43
- [ALL_IV_FUNCTION_LIBRARIES](#) on page 2-44
- [ALL_IV_FUNCTION_PARAMETERS](#) on page 2-44
- [ALL_IV_TABLE_FUNCTIONS](#) on page 2-45
- [ALL_IV_FUNCTION_IMPLS](#) on page 2-45

Configuration Model Views

- [ALL_IV_OBJECT_CONFIGURATIONS](#) on page 2-46
- [ALL_IV_CONFIGURATIONS](#) on page 2-46
- [ALL_IV_CONTROL_CENTERS](#) on page 2-46

Deployment Model Views

- [ALL_IV_CONNECTORS](#) on page 2-47
- [ALL_IV_LOCATIONS](#) on page 2-47
- [ALL_IV_RUNTIME_REPOSITORIES](#) on page 2-48

Mapping Model Views

- [ALL_IV_XFORM_MAPS](#) on page 2-48
- [ALL_IV_XFORM_MAP_COMPONENTS](#) on page 2-49
- [ALL_IV_XFORM_MAP_PARAMETERS](#) on page 2-50
- [ALL_IV_XFORM_MAP_PROPERTIES](#) on page 2-50
- [ALL_IV_XFORM_MAP_DETAILS](#) on page 2-51
- [ALL_IV_PLUGGABLE_MAPS](#) on page 2-51
- [ALL_IV_PLUGGABLE_MAP_LIBRARIES](#) on page 2-51
- [ALL_IV_PLUG_MAP_PARAMETERS](#) on page 2-52
- [ALL_IV_PLUG_MAP_COMPONENTS](#) on page 2-52

Process Flow Model Views

- [ALL_IV_PACKAGES](#) on page 2-53
- [ALL_IV_PROCESSES](#) on page 2-53
- [ALL_IV_PROCESS_ACTIVITIES](#) on page 2-54
- [ALL_IV_PROCESS_PARAMETERS](#) on page 2-54
- [ALL_IV_PROCESS_TRANSITIONS](#) on page 2-55
- [ALL_IV_PROCESS_VARIABLES](#) on page 2-55
- [ALL_IV_SUB_PROCESSES](#) on page 2-56

Data Profiling Views

- [ALL_IV_PROFILES](#) on page 2-56

- [ALL_IV_PROFILE_COLUMNS](#) on page 2-57
- [ALL_IV_PROFILE_DOMAIN_VALUES](#) on page 2-59
- [ALL_IV_FUNCTIONAL_DEPENDENCIES](#) on page 2-59
- [ALL_IV_PROFILE_FOREIGN_KEYS](#) on page 2-60
- [ALL_IV_PROFILE_KEY_COLUMN_USES](#) on page 2-61
- [ALL_IV_PROFILE_OBJECTS](#) on page 2-61
- [ALL_IV_PROFILE_PATTERN_VALUES](#) on page 2-62
- [ALL_IV_PROFILE_RULES](#) on page 2-62
- [ALL_IV_PROFILE_UNIQUE_KEYS](#) on page 2-62

Data Rules Views

- [ALL_IV_DATA_RULES](#) on page 2-63
- [ALL_IV_DATA_RULE_ATTRIBUTES](#) on page 2-64
- [ALL_IV_DATA_RULE_ATTR_USAGES](#) on page 2-64
- [ALL_IV_DATA_RULE_DOMAINS](#) on page 2-65
- [ALL_IV_DATA_RULE_GROUPS](#) on page 2-65
- [ALL_IV_DATA_RULE_GROUP_USAGES](#) on page 2-66
- [ALL_IV_DATA_RULE_PROPERTIES](#) on page 2-66
- [ALL_IV_DATA_RULE_USAGES](#) on page 2-67

User Defined Object Views

- [ALL_IV_UDO_FCOS](#) on page 2-67
- [ALL_IV_UDO_FOLDERS](#) on page 2-68
- [ALL_IV_UDO_SCOS](#) on page 2-68
- [ALL_IV_UDO_ASSOCIATIONS](#) on page 2-69

Expert Views

- [ALL_IV_EXPERTS](#) on page 2-69
- [ALL_IV_EXPERT_PARAMETERS](#) on page 2-69
- [ALL_IV_EXPERT_TASKS](#) on page 2-70
- [ALL_IV_EXPERT_TRANSITIONS](#) on page 2-70
- [ALL_IV_EXPERT_VARIABLES](#) on page 2-71
- [ALL_IV_NESTED_EXPERTS](#) on page 2-71

Business Intelligence Views

- [ALL_IV_ALTERNATIVE_SORT_ORDERS](#) on page 2-72
- [ALL_IV_BUSINESS_AREAS](#) on page 2-73
- [ALL_IV_BUSINESS_AREA_FOLDERS](#) on page 2-73
- [ALL_IV_PRESENTATION_TEMPLATES](#) on page 2-74
- [ALL_IV_DRILLS_TO_DETAIL](#) on page 2-74
- [ALL_IV_DRILL_LEVELS](#) on page 2-75

- [ALL_IV_ITEM_FOLDERS](#) on page 2-75
- [ALL_IV_ITEM_FOLDER_JOIN_USAGES](#) on page 2-76
- [ALL_IV_ITEMS](#) on page 2-76
- [ALL_IV_ITEM_FORMULA_REFS](#) on page 2-77
- [ALL_IV_DATA_ITEMS](#) on page 2-78
- [ALL_IV_EDGE_ITEMS](#) on page 2-78
- [ALL_IV_DRILL_PATHS](#) on page 2-79
- [ALL_IV_DRILL_LEVEL_ITEMS](#) on page 2-79
- [ALL_IV_DRILL_PATH_JOIN_USAGES](#) on page 2-80
- [ALL_IV_LISTS_OF_VALUES](#) on page 2-80
- [ALL_IV_REGISTERED_FUNCTIONS](#) on page 2-81
- [ALL_IV_CONDITION_FORMULA_REFS](#) on page 2-81
- [ALL_IV_JOIN_COMPONENTS](#) on page 2-82
- [ALL_IV_JOINS](#) on page 2-82
- [ALL_IV_CONDITIONS](#) on page 2-83

Real Time Views

- [ALL_IV_STREAMS_QUEUES](#) on page 2-84
- [ALL_IV_QUEUES](#) on page 2-84
- [ALL_IV_QUEUE_PROPAGATIONS](#) on page 2-85
- [ALL_IV_QUEUE_TABLES](#) on page 2-85
- [ALL_IV_STREAMS_CAPTURE](#) on page 2-85
- [ALL_IV_CAPTURE_RELATIONS](#) on page 2-86

Scheduling Views

- [ALL_IV_SCHEDULABLE](#) on page 2-86
- [ALL_IV_CALENDAR_SCHEDULES](#) on page 2-87

Others

- [ALL_IV_ACTIVITY_FOLDERS](#) on page 2-87
- [ALL_IV_ACTIVITY_TEMPLATES](#) on page 2-87
- [ALL_IV_PLS_COLLECTIONS](#) on page 2-88
- [ALL_IV_PLS_RECORDS](#) on page 2-88
- [ALL_IV_ROW_RELATIONSHIPS](#) on page 2-89

Note: In addition to the listed views, Warehouse Builder also contains the public view `ALL_IV_TABLE_FUNC_PROPERTIES`, which is invalid.

General Model Views

Table 2-1 ALL_IV_ALL_OBJECTS

Column Name	Data Type	Description
OBJECT_ID	NUMBER(9)	ID of the object
OBJECT_UOID	VARCHAR2(255)	UOID of the object
OBJECT_TYPE	VARCHAR2(4000)	Type of the object
OBJECT_NAME	VARCHAR2(4000)	Physical name of the object
BUSINESS_NAME	VARCHAR2(4000)	Business name of the object
CONTEXT_NAME	VARCHAR2(4000)	Name of the object, prefixed with its module's name, and project's name if existed
DESCRIPTION	VARCHAR2(4000)	Description of the object
PARENT_OBJECT_ID	NUMBER(9)	Container object ID for the object. Container object could be a module, for example, for a dimension, or a table for a column
PARENT_OBJECT_TYPE	VARCHAR2(4000)	Type of the parent object
PARENT_OBJECT_NAME	VARCHAR2(4000)	Name of the parent object
IS_VALID	VARCHAR2(13)	Is the object valid? It only makes sense for the objects that can be validated
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-2 ALL_IV_OBJECTS

Column Name	Data Type	Description
OBJECT_ID	NUMBER(9)	ID of the object (the difference between this view and 2.1 view is that 2.1 view includes all objects in this view, PLUS all archived snapshot objects (for MCM service))
OBJECT_TYPE	VARCHAR2(4000)	Type of the object
OBJECT_NAME	VARCHAR2(4000)	Physical name of the object
BUSINESS_NAME	VARCHAR2(4000)	Business name of the object
DESCRIPTION	VARCHAR2(4000)	Description of the object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–3 ALL_IV_OBJECT_PROPERTIES

Column Name	Data Type	Description
OBJECT_ID	NUMBER(9)	ID of the object
OBJECT_TYPE	VARCHAR2(4000)	Type of the object
OBJECT_NAME	VARCHAR2(255)	Physical name of the object
PROPERTY_ID	NUMBER(9)	ID of the object's property
PROPERTY_NAME	VARCHAR2(255)	ID of the property name
PROPERTY_VALUE	VARCHAR2(4000)	Value of the property

Table 2–4 ALL_IV_MLS_OBJECTS

Column Name	Data Type	Description
OBJECT_ID	NUMBER(9)	ID of the object (covers the same set as 2.2 view)
LANGUAGE_ID	VARCHAR2(255)	ID of the language (predefined internally by Warehouse Builder). To get language name, please join with 2.5 view
BUSINESS_NAME	VARCHAR2(4000)	Business name of the object
DESCRIPTION	VARCHAR2(4000)	Description of the object

Table 2–5 ALL_IV_SUPPORTED_LANGUAGES

Column Name	Data Type	Description
LANGUAGE_ID	VARCHAR2(255)	ID of the language
LANGUAGE_NAME	VARCHAR2(64)	Name of the language
ISBASELANGUAGE	VARCHAR2(1)	Is it a base language (for example, EN or FR)

Table 2–6 ALL_IV_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the module
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of the module
SCHEMA_ID	NUMBER(9)	ID of the module (repeated column, just to keep backward compatibility)
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
BUSINESS_NAME	VARCHAR2(4000)	Business name of the module
DESCRIPTION	VARCHAR2(4000)	Description of the module
STATUS	VARCHAR2(40)	Module status (dev, QA, prod)
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module
LOCATION_NAME	VARCHAR2(255)	Name of associated location for this module
UPDATED_ON	DATE	Update timestamp

Table 2-6 (Cont.) ALL_IV_MODULES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-7 ALL_IV_PROJECTS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
BUSINESS_NAME	VARCHAR2(4000)	Business name of the project
DESCRIPTION	VARCHAR2(4000)	Description of the project
VERSION_LABEL	VARCHAR2(255)	Version of the project
IS_VALID	VARCHAR2(13)	Is this project valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-8 ALL_IV_INFORMATION_SYSTEMS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the module
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of the module
INFORMATION_SYSTEM_TYPE	VARCHAR2(4000)	Type of the module
BUSINESS_NAME	VARCHAR2(4000)	Business name of the module
DESCRIPTION	VARCHAR2(4000)	Description of the module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (shown by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module
VENDOR	VARCHAR2(40)	Vendor name
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module. It is meaningful only for database applications
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external data source for the module
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module

Table 2–8 (Cont.) ALL_IV_INFORMATION_SYSTEMS

Column Name	Data Type	Description
LOCATION_NAME	VARCHAR2(255)	Physical name of the associated location
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–9 ALL_IV_INSTALLATIONS

Column Name	Data Type	Description
INSTALLATION_ID	NUMBER(9)	ID of the Warehouse Builder repository
INSTALLATION_NAME	VARCHAR2(255)	Physical name of the Warehouse Builder repository
BUSINESS_NAME	VARCHAR2(4000)	Business name of the Warehouse Builder repository
DESCRIPTION	VARCHAR2(4000)	Description of the Warehouse Builder repository
INSTALLED_VERSION	VARCHAR2(40)	Version of the Warehouse Builder repository
RELEASE	VARCHAR2(40)	Version of the Warehouse Builder Client
REPOSITORY_MODEL_VERSION	NUMBER(9)	Version of the Warehouse Builder model
PUBLIC_VIEW_VERSION	CHAR(5)	Version of the Warehouse Builder Public Views
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–10 ALL_IV_FILE_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of this file module
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of this file module
BUSINESS_NAME	VARCHAR2(4000)	Business name of this file module
DESCRIPTION	VARCHAR2(4000)	Description of this file module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (represented by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module
VENDOR	VARCHAR2(40)	Vendor name

Table 2–10 (Cont.) ALL_IV_FILE_MODULES

Column Name	Data Type	Description
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module. It is meaningful only for database applications
DIRECTORY	VARCHAR2(4000)	Name of the directory this file module connects to
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external file system for the module
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module
LOCATION_NAME	VARCHAR2(255)	Physical name of the associated location
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–11 ALL_IV_GATEWAY_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of this module
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of this module
BUSINESS_NAME	VARCHAR2(4000)	Business name of this module
DESCRIPTION	VARCHAR2(4000)	Description of this module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (represented by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module
VENDOR	VARCHAR2(40)	Vendor name
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module. It is meaningful only for database applications
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external data system for the module
STRONG_TYPE_NAME	VARCHAR2(255)	Used to differentiate which gateway component being employed, for example, Informix or Sybase
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module
LOCATION_NAME	VARCHAR2(255)	Physical name of the associated location

Table 2–11 (Cont.) ALL_IV_GATEWAY_MODULES

Column Name	Data Type	Description
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–12 ALL_IV_PACKAGED_APPS_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of this module (basically, the views wraps Oracle Applications, SAP)
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of this module
BUSINESS_NAME	VARCHAR2(4000)	Business name of this module
DESCRIPTION	VARCHAR2(4000)	Description of this module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (represented by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module
VENDOR	VARCHAR2(40)	Vendor name
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module. It is meaningful only for database applications
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external data system for the module
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module
LOCATION_NAME	VARCHAR2(255)	Physical name of the associated location
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–13 ALL_IV_PREDEFINED_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of this module (basically, the views wraps Oracle Pre-defined Transformations and Public Transformations)
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of this module
BUSINESS_NAME	VARCHAR2(4000)	Business name of this module
DESCRIPTION	VARCHAR2(4000)	Description of this module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (represented by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module
VENDOR	VARCHAR2(40)	Vendor name
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module. It is meaningful only for database applications
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external data system for the module
IS_VALID	VARCHAR2(13)	Is this module valid
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–14 ALL_IV_PROCESS_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of this module (basically, the views wraps Oracle Process Flow Module)
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of this module
BUSINESS_NAME	VARCHAR2(4000)	Business name of this module
DESCRIPTION	VARCHAR2(4000)	Description of this module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (represented by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module

Table 2–14 (Cont.) ALL_IV_PROCESS_MODULES

Column Name	Data Type	Description
VENDOR	VARCHAR2(40)	Vendor name
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module. It is meaningful only for database applications.
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external data system for the module
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module
LOCATION_NAME	VARCHAR2(255)	Physical name of the associated location
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–15 ALL_IV_WAREHOUSE_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project that this module belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of this module (basically, the views wraps Oracle Warehouse Module)
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of this module
BUSINESS_NAME	VARCHAR2(4000)	Business name of this module
DESCRIPTION	VARCHAR2(4000)	Description of this module
PRODUCT_TYPE	VARCHAR2(255)	Application type of the module (for example, Oracle apps or File based apps)
SYSTEM_TYPE	VARCHAR2(255)	Type of system that holds this application (represented by PRODUCT_TYPE)
VERSION_LABEL	NUMBER(9)	Version of the module
VENDOR	VARCHAR2(40)	Vendor name
DATABASE_LINK	VARCHAR2(40)	Name of the database link that physical points to data storage of this module
INTEGRATOR_NAME	VARCHAR2(255)	The name of Warehouse Builder integrator component that is used to access external data system for the module
IS_VALID	VARCHAR2(13)	Is this module valid
LOCATION_ID	NUMBER(9)	ID of the associated location for this module
LOCATION_NAME	VARCHAR2(255)	Physical name of the associated location
STATUS	VARCHAR2(17)	Status (dev, QA, or prod)
UPDATED_ON	DATE	Update timestamp

Table 2–15 (Cont.) ALL_IV_WAREHOUSE_MODULES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–16 ALL_IV_BUSINESS_DEF_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the business definition module
DESCRIPTION	VARCHAR2(4000)	Description of the business definition module
PRODUCT_TYPE	VARCHAR2(255)	Type of product (This will be Oracle Discoverer Application)
SYSTEM_TYPE	VARCHAR2(255)	Internal
VERSION_LABEL	NUMBER(9)	Internal
VENDOR	VARCHAR2(40)	Name of the vendor (This will be Oracle)
DATABASE_LINK	VARCHAR2(40)	Not applicable
INTEGRATOR_NAME	VARCHAR2(255)	Internal
IS_VALID	VARCHAR2(13)	Validation status of the module
STATUS	VARCHAR2(40)	The module status (Development, Quality Assurance, or Production)
LOCATION_ID	NUMBER(9)	ID of the Location associated with the business definition module
LOCATION_NAME	VARCHAR2(255)	Name of the Location associated with the business definition module
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–17 ALL_IV_BUSINESS_PRES_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
MODULE_ID	NUMBER(9)	ID of the business presentation module
MODULE_NAME	VARCHAR2(255)	Name of the business presentation module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the business presentation module
DESCRIPTION	VARCHAR2(4000)	Description of the business presentation module

Table 2–17 (Cont.) ALL_IV_BUSINESS_PRES_MODULES

Column Name	Data Type	Description
PRODUCT_TYPE	VARCHAR2(255)	Type of product (This will be Oracle BI Beans Application)
SYSTEM_TYPE	VARCHAR2(255)	Internal
VERSION_LABEL	NUMBER(9)	Internal
VENDOR	VARCHAR2(40)	Name of the vendor (This will be Oracle)
DATABASE_LINK	VARCHAR2(40)	Not applicable
INTEGRATOR_NAME	VARCHAR2(255)	Internal
IS_VALID	VARCHAR2(13)	Validation status of the module
STATUS	VARCHAR2(40)	The module status (Development, Quality Assurance, or Production)
LOCATION_ID	NUMBER(9)	Id of the location with which the business presentation module is associated
LOCATION_NAME	VARCHAR2(255)	Name of the location with which the business presentation module is associated
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–18 ALL_IV_CALENDAR_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
CALENDAR_MODULE_ID	NUMBER(9)	ID of the calendar module
CALENDAR_MODULE_NAME	VARCHAR2(255)	Name of the calendar module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the calendar module
DESCRIPTION	VARCHAR2(4000)	Description of the calendar module
IS_VALID	VARCHAR2(13)	Is the calendar module valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–19 ALL_IV_CMIV_DEFINITIONS

Column Name	Data Type	Description
CMIV_ID	NUMBER(9)	ID of the CMI definition
NAME	VARCHAR2(255)	Name of the CMI definition
BUSINESS_NAME	VARCHAR2(1000)	Business name
MIVMODE	VARCHAR2(40)	Mode of CMI (SQL or XML File)

Table 2–19 (Cont.) ALL_IV_CMIV_DEFINITIONS

Column Name	Data Type	Description
TYPE	VARCHAR2(40)	Type
TABLE_FILTER_SUPPORTED	VARCHAR2(1)	Flag on supporting table filter
VIEW_FILTER_SUPPORTED	VARCHAR2(1)	Flag on supporting view filter
SEQUENCE_FILTER_SUPPORTED	VARCHAR2(1)	Flag on supporting sequence filter
TABLE_FKLEVEL_SUPPORTED	VARCHAR2(1)	Flag on supporting table foreign key level dependency
MULTI_TREE_SUPPORTED	VARCHAR2(1)	Flag on supporting multiple tree in business component navigation
REIMPORT_SUPPORTED	VARCHAR2(1)	Flag on supporting reimport
TEST_DB_LINK	VARCHAR2(255)	DB link used for testing the CMI definition
TESTDIRECTORY	VARCHAR2(255)	File directory used for testing the CMI definition
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–20 ALL_IV_CMIV_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the information system
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Name of the information system
BUSINESS_NAME	VARCHAR2(1000)	Business name of the module
DESCRIPTION	VARCHAR2(4000)	Description of the module
PRODUCT_TYPE	VARCHAR2(255)	Product type
SYSTEM_TYPE	VARCHAR2(255)	System type
VERSION_LABEL	NUMBER(9)	Version label
VENDOR	VARCHAR2(40)	Vendor
DATABASE_LINK	VARCHAR2(40)	Database link
INTEGRATOR_NAME	VARCHAR2(255)	Name of the integrator
IS_VALID	VARCHAR2(13)	Flag on valid module
STATUS	VARCHAR2(40)	Status
LOCATION_ID	NUMBER(9)	ID of the location to access data
LOCATION_NAME	VARCHAR2(255)	Name of the data location
METADATA_LOCATION_ID	NUMBER(9)	ID of the location to access metadata
METADATA_LOCATION_NAME	VARCHAR2(255)	Name of the metadata location
UPDATED_ON	DATE	Update timestamp

Table 2–20 (Cont.) ALL_IV_CMIV_MODULES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–21 ALL_IV_DATA_RULE_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project to which the data rule module belongs
PROJECT_NAME	VARCHAR2(255)	Name of the project to which the data rule module belongs
SCHEMA_ID	NUMBER(9)	ID of the data rule module
SCHEMA_NAME	VARCHAR2(255)	Name of the data rule module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the data rule module
DESCRIPTION	VARCHAR2(4000)	Description of the data rule module
STATUS	VARCHAR2(40)	Not used
IS_VALID	VARCHAR2(13)	Not used
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–22 ALL_IV_EXPERT_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the information system
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Name of the information system
BUSINESS_NAME	VARCHAR2(1000)	The business name of the expert module
DESCRIPTION	VARCHAR2(4000)	Description of the module
VERSION_LABEL	NUMBER(9)	The version for this module
IS_VALID	VARCHAR2(13)	Is this module valid
STATUS	VARCHAR2(40)	The status for this module
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–23 ALL_IV_PF_CORRECTED_MODULES

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
CORRECTED_MODULE_ID	NUMBER(9)	ID of the corrected module
CORRECTED_MODULE_NAME	VARCHAR2(255)	Name of the corrected module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the corrected module
DESCRIPTION	VARCHAR2(4000)	Description of the corrected module
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–24 ALL_IV_SAP_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the information system
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Name of the information system
BUSINESS_NAME	VARCHAR2(1000)	Business name of the SAP module
DESCRIPTION	VARCHAR2(4000)	Description of the SAP module
PRODUCT_TYPE	VARCHAR2(255)	Product type
SYSTEM_TYPE	VARCHAR2(255)	System type
VERSION_LABEL	NUMBER(9)	Version label
VENDOR	VARCHAR2(40)	Vendor
INTEGRATOR_NAME	VARCHAR2(255)	Integrator name
IS_VALID	VARCHAR2(13)	Flag if the module is valid
STATUS	VARCHAR2(40)	Status
LOCATION_ID	NUMBER(9)	Data location ID
LOCATION_NAME	VARCHAR2(255)	Data location name
METADATA_LOCATION_ID	NUMBER(9)	Metadata location ID
METADATA_LOCATION_NAME	VARCHAR2(255)	Metadata location name
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–25 ALL_IV_TM_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
TM_ID	NUMBER(9)	ID of the transportable module
TM_NAME	VARCHAR2(255)	Name of the transportable module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the transportable module
DESCRIPTION	VARCHAR2(4000)	Description of the transportable module
IS_VALID	VARCHAR2(13)	Whether the transportable module is valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user
CHARSET	VARCHAR2(40)	Not used
ORACLEHOME	VARCHAR2(40)	Not used
HOST	VARCHAR2(40)	The source database host name
DEFAULTPORT	NUMBER(9)	The access port of source database
SERVICE	VARCHAR2(40)	The database service name of source database
SID	VARCHAR2(40)	Not used

Table 2–26 ALL_IV_UDO_MODULES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the information system
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Name of the information system
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
MODULE_TYPE	VARCHAR2(255)	Type of the module
BUSINESS_NAME	VARCHAR2(1000)	Business name of the module
DESCRIPTION	VARCHAR2(4000)	Description of the module
STATUS	VARCHAR2(40)	Status of the module
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–27 ALL_IV_CMIV_VIEWS

Column Name	Data Type	Description
MIV_DEFINITION_ID	NUMBER(9)	ID of the CMI definition
MIV_DEFINITION	VARCHAR2(255)	Name of the CMI definition
MIV_VIEW_ID	NUMBER(9)	ID of the CMI View
MIV_VIEW	VARCHAR2(255)	Name of the CMI View
BUSINESS_NAME	VARCHAR2(1000)	Business name of the CMI view
VIEW_TYPE	VARCHAR2(40)	Type of the CMI view
IS_DEFAULT	VARCHAR2(1)	Flag if the view definition is the default
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–28 ALL_IV_FIRSTCLASS_OBJECTS

Column Name	Data Type	Description
OBJECT_ID	NUMBER(9)	ID of the object
OBJECT_NAME	VARCHAR2(255)	Name of the object
BUSINESS_NAME	VARCHAR2(1000)	Business name of the object
DESCRIPTION	VARCHAR2(4000)	Description of the object
CLASSNAME	VARCHAR2(255)	Internal type of the object
OBJECT_TYPE	VARCHAR2(4000)	Type of object
SCRIPTING_TYPE	VARCHAR2(4000)	Type of object exposed in scripting
OBJECT_UOID	VARCHAR2(255)	UOID of the object
OWNINGFOLDER	NUMBER	Folder that owns the object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–29 ALL_IV_DB_FUNCTIONS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
FUNCTION_LIBRARY_ID	NUMBER(9)	ID of the function library
FUNCTION_LIBRARY_NAME	VARCHAR2(255)	Name of the function library
FUNCTION_ID	NUMBER(9)	ID of the function
FUNCTION_NAME	VARCHAR2(255)	Name of the function
BUSINESS_NAME	VARCHAR2(1000)	Business name of the function

Table 2–29 (Cont.) ALL_IV_DB_FUNCTIONS

Column Name	Data Type	Description
DESCRIPTION	VARCHAR2(4000)	Description of the function
SIGNATURE	VARCHAR2(4000)	Signature of the function
FUNCTION_TYPE	CHAR(8)	Function or Procedure
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Data Model Views

Table 2–30 ALL_IV_ADVANCED_QUEUES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this queue belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
QUEUE_ID	NUMBER(9)	ID of this queue
QUEUE_NAME	VARCHAR2(255)	Physical name of this queue
BUSINESS_NAME	VARCHAR2(4000)	Business name of this queue
DESCRIPTION	VARCHAR2(4000)	Description of this queue
LOAD_TYPE_ID	NUMBER(9)	ID of the load type
LOAD_TYPE_NAME	VARCHAR2(255)	Name of the load type
QUEUE_TABLE_NAME	VARCHAR2(40)	Name of the queue table
IS_VALID	VARCHAR2(13)	Is this queue valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–31 ALL_IV_ATTR_GROUPS

Column Name	Data Type	Description
DATA_ENTITY_ID	NUMBER(9)	ID of the data entity this attribute group belongs to
DATA_ENTITY_TYPE	VARCHAR2(4000)	Type of the data entity
DATA_ENTITY_NAME	VARCHAR2(255)	Physical name of the data entity
ATTRIBUTE_GROUP_NAME	VARCHAR2(255)	Physical name of this attribute group
ATTRIBUTE_GROUP_ID	NUMBER(9)	ID of this attribute group
BUSINESS_NAME	VARCHAR2(4000)	Business name of this attribute group
DESCRIPTION	VARCHAR2(4000)	Description of this attribute group
ATTRIBUTE_GROUP_TYPE	VARCHAR2(40)	Type of attribute group

Table 2–31 (Cont.) ALL_IV_ATTR_GROUPS

Column Name	Data Type	Description
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–32 ALL_IV_ATTR_GROUP_ITEM_USES

Column Name	Data Type	Description
ATTRIBUTE_GROUP_ID	NUMBER(9)	ID of the attribute group that this data item belongs to
ATTRIBUTE_GROUP_NAME	VARCHAR2(255)	Name of the attribute group
DATA_ITEM_ID	NUMBER(9)	ID of this data item
DATA_ITEM_TYPE	VARCHAR2(4000)	Type of this data item
DATA_ITEM_NAME	VARCHAR2(255)	Physical name of this data item
POSITION	NUMBER(9)	Position of this data item in the attribute group

Table 2–33 ALL_IV_CHECK_CONSTRAINTS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this check constraint belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
RELATION_ID	NUMBER(9)	ID of the relation entity this check constraint belongs to
RELATION_NAME	VARCHAR2(255)	Physical name of the relation entity
CONSTRAINT_ID	NUMBER(9)	ID of this check constraint
CONSTRAINT_NAME	VARCHAR2(255)	Physical name of this check constraint
BUSINESS_NAME	VARCHAR2(4000)	Business name of this check constraint
DESCRIPTION	VARCHAR2(4000)	Description of this check constraint
CONSTRAINT_TEXT	VARCHAR2(255)	Textual expression of this check constraint
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–34 ALL_IV_COLUMNS

Column Name	Data Type	Description
ENTITY_ID	NUMBER(9)	ID of the data entity this column belongs to
ENTITY_TYPE	VARCHAR2(4000)	Type of the data entity
ENTITY_NAME	VARCHAR2(255)	Physical name of the data entity
COLUMN_ID	NUMBER(9)	ID of this column
COLUMN_NAME	VARCHAR2(255)	Physical name of this column

Table 2–34 (Cont.) ALL_IV_COLUMNS

Column Name	Data Type	Description
BUSINESS_NAME	VARCHAR2(4000)	Business name of this column
DESCRIPTION	VARCHAR2(4000)	Description of this column
POSITION	NUMBER(9)	Position of this column in the data entity
DATA_TYPE	VARCHAR2(255)	Data type of this column
LENGTH	NUMBER(9)	Data length of this column
PRECISION	NUMBER(9)	Data precision of this column
SCALE	NUMBER(9)	Data scale of this column
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–35 ALL_IV_CONSTRAINTS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this constraint belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
RELATION_ID	NUMBER(9)	ID of the relational entity this constraint belongs to
RELATION_NAME	VARCHAR2(255)	Physical name of the relational entity
CONSTRAINT_ID	NUMBER(9)	ID of this constraint
CONSTRAINT_NAME	VARCHAR2(255)	Physical name of this constraint
CONSTRAINT_TYPE	VARCHAR2(21)	Type of this constraint (check, primary, foreign key)
BUSINESS_NAME	VARCHAR2(4000)	Business name of this constraint
DESCRIPTION	VARCHAR2(4000)	Description of this constraint
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–36 ALL_IV_CUBES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this cube belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
CUBE_ID	NUMBER(9)	ID of this cube
CUBE_NAME	VARCHAR2(255)	Physical name of this cube
BUSINESS_NAME	VARCHAR2(4000)	Business name of this cube
DESCRIPTION	VARCHAR2(4000)	Description of this cube
IS_VALID	VARCHAR2(13)	Is this cube valid

Table 2–36 (Cont.) ALL_IV_CUBES

Column Name	Data Type	Description
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user
AUTOSOLVE	CHAR(1)	Sets the flag to say whether to measure is Auto Solve
AWNAME	VARCHAR2(255)	The analytical workspace name where the cube is implemented
AWOBJECTNAME	VARCHAR2(4000)	The analytical workspace object name
AWTABLESPACE	VARCHAR2(255)	The analytical workspace tablespace name
COMPRESSED	CHAR(1)	Flag to check whether the cube is compressed
CREATEBITMAPS	CHAR(1)	Flag to check whether to create a bitmap for the cube
CREATECONSTRAINTS	CHAR(1)	Flag to check whether to create a constraint for the cube
IMPLEMENTATION	VARCHAR2(255)	Whether the storage of a cube is AW or Relational
LOADPOLICY	VARCHAR2(255)	Not applicable for Paris release
OLAPUSERVISIBLE	CHAR(1)	Flag to check whether the Cube is visible to OLAP end user
STORAGEPROPERTYTYPE	VARCHAR2(255)	The storage of a cube can be AW or Relational
USEGLOBALINDEX	CHAR(1)	Whether to generate a composite for measure partition combination
PARTITIONHIERARCHY	NUMBER(9)	The hierarchy by which one should partition the cube
INSTALLEDMODULE	NUMBER(9)	The Oracle module to which the cube belongs
PARTITIONLEVEL	NUMBER(9)	The Level by which one should partition the cube
BINDINGFACT	NUMBER(9)	The element ID for binding fact table to the cube

Table 2–37 ALL_IV_CUBE_DIMENSIONS

Column Name	Data Type	Description
CUBE_ID	NUMBER(9)	ID of the cube this dimension has associated with
CUBE_NAME	VARCHAR2(255)	Physical name of the cube
DIMENSION_ID	NUMBER(9)	ID of this dimension
DIMENSION_NAME	VARCHAR2(255)	Physical name of this dimension
DIMENSION_ALIAS	VARCHAR2(255)	Alias of this dimension

Table 2–38 ALL_IV_CUBE_MEASURES

Column Name	Data Type	Description
CUBE_ID	NUMBER(9)	ID of the cube this measure belongs to
CUBE_NAME	VARCHAR2(255)	Physical name of the cube
MEASURE_ID	NUMBER(9)	ID of this measure
MEASURE_NAME	VARCHAR2(255)	Physical name of this measure

Table 2–38 (Cont.) ALL_IV_CUBE_MEASURES

Column Name	Data Type	Description
BUSINESS_NAME	VARCHAR2(4000)	Business name of this measure
DESCRIPTION	VARCHAR2(4000)	Description of this measure
POSITION	NUMBER(9)	Position of this measure within the cube
DATA_TYPE	VARCHAR2(255)	Data type of this measure
LENGTH	NUMBER(9)	Data length of this measure
PRECISION	NUMBER(9)	Data precision of this measure
SCALE	NUMBER(9)	Data scale of this measure
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user
TABLE_ID	NUMBER(9)	ID of the table
TABLE_NAME	VARCHAR2(255)	Name of the table

Table 2–39 ALL_IV_CUBE_MEASURE_DIM_USES

Column Name	Data Type	Description
CUBE_ID	NUMBER(9)	ID of the cube (Note, this view is redundant, it can be achieved by joining 2.23, 2.24 views. It will be removed in the future)
CUBE_NAME	VARCHAR2(255)	Physical name of the cube
MEASURE_ID	NUMBER(9)	ID of the measure belonging to this cube
MEASURE_NAME	VARCHAR2(255)	Physical name of the measure
DIMENSION_ID	NUMBER	ID of the dimension associated with this cube
DIMENSION_NAME	VARCHAR2(255)	Physical name of the dimension
DIMENSION_ALIAS	VARCHAR2(255)	Alias of the dimension

Table 2–40 ALL_IV_DIMENSIONS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this dimension belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
DIMENSION_ID	NUMBER(9)	ID of this dimension
DIMENSION_NAME	VARCHAR2(255)	Physical name of this dimension
BUSINESS_NAME	VARCHAR2(4000)	Business name of this dimension
DESCRIPTION	VARCHAR2(4000)	Description of this dimension
IS_VALID	VARCHAR2(13)	Is this dimension valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user

Table 2–40 (Cont.) ALL_IV_DIMENSIONS

Column Name	Data Type	Description
CREATED_BY	VARCHAR2(255)	Created by user
AWNAME	VARCHAR2(255)	The analytical workspace name where the dimension is implemented
AWOBJECTNAME	VARCHAR2(4000)	The analytical workspace object name
AWTABLESPACE	VARCHAR2(255)	The analytical workspace tablespace name
CREATECONSTRAINTS	CHAR(1)	Flag to check whether to create a constraint for the dimension
IMPLEMENTATION	VARCHAR2(255)	Type of implementation of the dimension (The storage of a dimension can be AW or Relational)
LOADPOLICY	VARCHAR2(255)	Data policy for loading dimension where Warehouse Builder mapping code relies on database constraints to detect the orphans (level records without parent)
OLAPPRIMARYSORTORDER	VARCHAR2(255)	The primary sorting order for dimension data in OLAP service
OLAPSECONDARYSORTORDER	VARCHAR2(255)	The secondary sorting order for dimension data in OLAP service
OLAPTYPE	VARCHAR2(255)	Type of OLAP Dimension Normal or Time
OLAPUSERVISIBLE	CHAR(1)	Whether the dimension is visible to OLAP end user
OWBTYPE	VARCHAR2(255)	Type of Dimension Normal or Time
REMOVEPOLICY	VARCHAR2(255)	Orphan Management (not applicable for Paris)
SLOWLYCHANGINGTYPE	NUMBER(9)	Slowly changing policy to be applied on the dimension
STORAGEPROPERTYTYPE	VARCHAR2(255)	The storage of a cube can be AW or Relational
VALUEBASED	CHAR(1)	The flag to define a value based hierarchy, applicable for AW only
DEFAULTDISPLAYHIERARCHY	NUMBER(9)	The hierarchy is set as default display hierarchy
INSTALLEDMODULE	NUMBER(9)	The Oracle module to which the cube belongs
TIMEDIMPOPULATIONMAP	NUMBER(9)	The element ID of map for a time dimension
DIMENSIONKEYSEQUENCE	NUMBER(9)	Element ID for sequence to generate the dimension key

Table 2–41 ALL_IV_DIM_HIERARCHIES

Column Name	Data Type	Description
DIMENSION_ID	NUMBER(9)	ID of the dimension this hierarchy belongs to
DIMENSION_NAME	VARCHAR2(255)	Physical name of this dimension
HIERARCHY_ID	NUMBER(9)	ID of this hierarchy
HIERARCHY_NAME	VARCHAR2(255)	Physical name of this hierarchy
BUSINESS_NAME	VARCHAR2(1000)	Business name of this hierarchy
DESCRIPTION	VARCHAR2(4000)	Description of this hierarchy
UPDATED_ON	DATE	Update timestamp

Table 2–41 (Cont.) ALL_IV_DIM_HIERARCHIES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–42 ALL_IV_DIM_HIERARCHY_LEVELS

Column Name	Data Type	Description
LEVEL_USE_ID	NUMBER(9)	ID of the level relationship that this level and this parent level participates in
LEVEL_USE_NAME	VARCHAR2(255)	Level name used on Hierarchy
HIERARCHY_ID	NUMBER(9)	ID of the hierarchy that this level and this parent level belongs to
HIERARCHY_NAME	VARCHAR2(255)	Physical name of the hierarchy
LEVEL_ID	NUMBER(9)	ID of this level
LEVEL_NAME	VARCHAR2(255)	Physical name of this level
LEVEL_DESCRIPTION	VARCHAR2(4000)	Description of this level
PARENT_LEVEL_ID	NUMBER(9)	ID of this parent level
PARENT_LEVEL_NAME	VARCHAR2(255)	Physical name of this parent level
POSITION	NUMBER(9)	Position of this level

Table 2–43 ALL_IV_DIM_LEVELS

Column Name	Data Type	Description
DIMENSION_ID	NUMBER(9)	ID of the dimension this level belongs to
DIMENSION_NAME	VARCHAR2(255)	Physical name of the dimension
LEVEL_ID	NUMBER(9)	ID of this level
LEVEL_NAME	VARCHAR2(255)	Physical name of this level
BUSINESS_NAME	VARCHAR2(4000)	Business name of this level
DESCRIPTION	VARCHAR2(4000)	Description of this level
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–44 ALL_IV_DIM_LEVEL_ATTRIBUTES

Column Name	Data Type	Description
LEVEL_ID	NUMBER(9)	ID of the level this attribute belongs to
LEVEL_NAME	VARCHAR2(255)	Physical name of the level
ATTRIBUTE_ID	NUMBER(9)	ID of this attribute
ATTRIBUTE_NAME	VARCHAR2(255)	Physical name of this attribute

Table 2–44 (Cont.) ALL_IV_DIM_LEVEL_ATTRIBUTES

Column Name	Data Type	Description
DIMENSION_ATTRIBUTE	VARCHAR2(255)	Name of the dimension attribute that the level attribute implements
BUSINESS_NAME	VARCHAR2(1000)	Business name of this attribute
DESCRIPTION	VARCHAR2(4000)	Description of this attribute
POSITION	NUMBER(9)	Position of this attribute within the level
DATA_TYPE	VARCHAR2(255)	Data type of this attribute
LENGTH	NUMBER(9)	Data length of this attribute
PRECISION	NUMBER(9)	Data precision of this attribute
SCALE	NUMBER(9)	Data scale of this attribute
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–45 ALL_IV_EXTERNAL_COLUMNS

Column Name	Data Type	Description
ENTITY_ID	NUMBER(9)	ID of the external table that this column belongs to
ENTITY_NAME	VARCHAR2(255)	Name of the external table
COLUMN_ID	NUMBER(9)	ID of this column
COLUMN_NAME	VARCHAR2(255)	Physical name of this column
BUSINESS_NAME	VARCHAR2(4000)	Business name of this column
DESCRIPTION	VARCHAR2(4000)	Description of this column
POSITION	NUMBER(9)	Position of this column within the external table
DATA_TYPE	VARCHAR2(255)	Data type of this column
LENGTH	NUMBER(9)	Data length of this column
PRECISION	NUMBER(9)	Data precision of this column
SCALE	NUMBER(9)	Data scale of this column
SOURCE_FIELD_ID	NUMBER(9)	ID of the field that this column maps to
SOURCE_FIELD_NAME	VARCHAR2(255)	Physical name of the source field
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–46 ALL_IV_EXTERNAL_TABLES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this external table belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
LOCATION_ID	NUMBER(9)	ID of the location where the module is deployed to
LOCATION_NAME	VARCHAR2(255)	Physical name of the location
TABLE_ID	NUMBER(9)	ID of the external table
TABLE_NAME	VARCHAR2(255)	Physical name of the external table
BUSINESS_NAME	VARCHAR2(4000)	Business name of the external table
DESCRIPTION	VARCHAR2(4000)	Description of the external table
SOURCE_RECORD_ID	NUMBER(9)	ID of the record that this external table maps to
SOURCE_RECORD_NAME	VARCHAR2(255)	Physical name of the source record
SOURCE_FILE_NAME	VARCHAR2(255)	Physical name of the file that this source record belongs to
ACCESS_PARAMETERS	VARCHAR2(4000)	Expression for parameters that are used to access the source record
IS_VALID	VARCHAR2(13)	Is this external table valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–47 ALL_IV_FOREIGN_KEYS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module that this foreign key belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
ENTITY_ID	NUMBER(9)	ID of the data entity this foreign key belongs to
ENTITY_NAME	VARCHAR2(255)	Physical name of the data entity
ENTITY_TYPE	VARCHAR2(4000)	Type of the data type (for example, table, view)
FOREIGN_KEY_ID	NUMBER(9)	ID of this foreign key
FOREIGN_KEY_NAME	VARCHAR2(255)	Physical name of this foreign key
BUSINESS_NAME	VARCHAR2(4000)	Business name of this foreign key
DESCRIPTION	VARCHAR2(4000)	Description of this foreign key
KEY_ID	NUMBER(9)	ID of the associated key for this foreign key
KEY_NAME	VARCHAR2(255)	Physical name of the key
IS_DISABLED	CHAR(1)	Is this foreign key disabled
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–48 ALL_IV_KEYS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this key belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of this module
ENTITY_ID	NUMBER(9)	ID of the data entity this key belongs to
ENTITY_NAME	VARCHAR2(255)	Physical name of the data entity
ENTITY_TYPE	VARCHAR2(4000)	Type of the data entity (for example, table, view)
KEY_ID	NUMBER(9)	ID of this key
KEY_NAME	VARCHAR2(255)	Physical name of this key
BUSINESS_NAME	VARCHAR2(4000)	Business of this key
DESCRIPTION	VARCHAR2(4000)	Description of this key
IS_PRIMARY	VARCHAR2(9)	Is this key primary key
IS_DISABLED	CHAR(1)	Is this key disabled
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–49 ALL_IV_KEY_COLUMN_USES

Column Name	Data Type	Description
KEY_ID	NUMBER(9)	ID of the key that this column is associated with
KEY_NAME	VARCHAR2(255)	Physical name of the key
KEY_TYPE	VARCHAR2(11)	Type of the key (primary, unique, foreign)
COLUMN_ID	NUMBER(9)	ID of this column
COLUMN_NAME	VARCHAR2(255)	Physical name of this column
POSITION	NUMBER(9)	Position of this column with the key

Table 2–50 ALL_IV_MATERIALIZED_VIEWS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this materialized view belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
VIEW_ID	NUMBER(9)	ID of this materialized view
VIEW_NAME	VARCHAR2(255)	Physical name of this materialized view
BUSINESS_NAME	VARCHAR2(4000)	Business name of this materialized view
DESCRIPTION	VARCHAR2(4000)	Description of this materialized view
QUERY_TEXT	VARCHAR2(4000)	Textual expression of query statement for this materialized view
IS_VALID	VARCHAR2(13)	Is this materialized view valid
UPDATED_ON	DATE	Update timestamp

Table 2–50 (Cont.) ALL_IV_MATERIALIZED_VIEWS

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–51 ALL_IV_OBJECT_TYPES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this object type belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
FOLDER_ID	NUMBER(9)	ID of the folder this object type belongs to
FOLDER_NAME	VARCHAR2(255)	Physical name of the folder
OBJECT_TYPE_ID	NUMBER(9)	ID of this object type
OBJECT_TYPE_NAME	VARCHAR2(255)	Physical name of this object type
BUSINESS_NAME	VARCHAR2(4000)	Business name of this object
DESCRIPTION	VARCHAR2(4000)	Description of this object type
TYPE	VARCHAR2(40)	Type of this object type
IS_VALID	VARCHAR2(13)	Is this object type valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–52 ALL_IV_RECORD_FIELDS

Column Name	Data Type	Description
FIRSTCLASS_OBJECT_ID	NUMBER(9)	ID of the first class object that this record field belongs to (normally, this ID will be the same as the following relational ID)
FRISTCLASS_OBJECT_NAME	VARCHAR2(255)	Physical name of the first class object
RELATION_ID	NUMBER(9)	ID of the relational entity this record field belongs to
RELATION_NAME	VARCHAR2(255)	Physical name of the relational entity
RECORDFIELD_ID	NUMBER(9)	ID of this record field
RECORDFIELD_NAME	VARCHAR2(255)	Physical name of this record field
BUSINESS_NAME	VARCHAR2(4000)	Business name of this record field
DESCRIPTION	VARCHAR2(4000)	Description of this record field
POSITION	NUMBER(9)	Position of this record field
DATA_TYPE	VARCHAR2(255)	Data type of this record field
LENGTH	NUMBER(9)	Data length of this record field
PRECISION	NUMBER(9)	Data precision of this record field
SCALE	NUMBER(9)	Data scale of this record field

Table 2–52 (Cont.) ALL_IV_RECORD_FIELDS

Column Name	Data Type	Description
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–53 ALL_IV_RELATIONS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this relational entity belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
RELATION_ID	NUMBER(9)	ID of this relational entity
RELATION_NAME	VARCHAR2(255)	Physical name of this relational entity
RELATION_TYPE	VARCHAR2(16)	Type of this relational entity (such as table, view, sequence and materialized view)
BUSINESS_NAME	VARCHAR2(4000)	Business name of this relational entity
DESCRIPTION	VARCHAR2(4000)	Description of this relational entity
IS_VALID	VARCHAR2(13)	Is this relational entity valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–54 ALL_IV_SEQUENCES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this sequence belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module the sequence belongs to
SEQUENCE_ID	NUMBER(9)	ID of the sequence
SEQUENCE_NAME	VARCHAR2(255)	Physical name of the sequence
BUSINESS_NAME	VARCHAR2(4000)	Business name of the sequence
DESCRIPTION	VARCHAR2(4000)	Description of the sequence
IS_VALID	VARCHAR2(13)	Is this sequence valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–55 ALL_IV_VIEWS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this view belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
VIEW_ID	NUMBER(9)	ID of this view
VIEW_NAME	VARCHAR2(255)	Physical name of this view
QUERY_TEXT	VARCHAR2(4000)	Textual expression of the query for this view
BUSINESS_NAME	VARCHAR2(4000)	Business name of this view
DESCRIPTION	VARCHAR2(4000)	Description of this view
IS_VALID	VARCHAR2(13)	Is this view valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–56 ALL_IV_TABLES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this table belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
TABLE_ID	NUMBER(9)	ID of this table
TABLE_NAME	VARCHAR2(255)	Physical name of this table
BUSINESS_NAME	VARCHAR2(4000)	Business name of this table
DESCRIPTION	VARCHAR2(4000)	Description of this table
IS_VALID	VARCHAR2(13)	Is this table valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–57 ALL_IV_CALENDARS

Column Name	Data Type	Description
CALENDAR_MODULE_ID	NUMBER(9)	ID of the owning calendar module
CALENDAR_MODULE_NAME	VARCHAR2(255)	Name of the owning calendar module
CALENDAR_ID	NUMBER(9)	ID of the calendar
CALENDAR_NAME	VARCHAR2(255)	Name of the calendar
BUSINESS_NAME	VARCHAR2(1000)	Business name of the calendar
DESCRIPTION	VARCHAR2(4000)	Description of the calendar
IS_VALID	VARCHAR2(13)	Is the calendar valid
UPDATED_ON	DATE	Update timestamp

Table 2–57 (Cont.) ALL_IV_CALENDARS

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–58 ALL_IV_VARRAYS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
VARRAY_ID	NUMBER(9)	ID of the Varray
VARRAY_NAME	VARCHAR2(255)	Name of the Varray
BUSINESS_NAME	VARCHAR2(1000)	Business name of the Varray
DESCRIPTION	VARCHAR2(4000)	Description of the Varray
IS_VALID	VARCHAR2(13)	Is the Varray valid
BASE_ELEMENT_NAME	VARCHAR2(767)	Name of the base element
BASE_ELEMENT_PRECISION	NUMBER	Precision of the base element
BASE_ELEMENT_SCALE	NUMBER	Scale of the base element
BASE_ELEMENT_LENGTH	NUMBER	Length of the base element
ARRAY_LENGTH	NUMBER(9)	Number of elements in the Varray
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–59 ALL_IV_SCHEMAS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
BUSINESS_NAME	VARCHAR2(1000)	Business name of the schema
DESCRIPTION	VARCHAR2(4000)	Description of the schema
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the information system
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Name of the information system
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Create timestamp

Table 2-60 ALL_IV_PROCEDURES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
FUNCTION_LIBRARY_ID	NUMBER(9)	ID of the function library
FUNCTION_LIBRARY_NAME	VARCHAR2(255)	Name of the function library
FUNCTION_ID	NUMBER(9)	ID of the procedure
FUNCTION_NAME	VARCHAR2(255)	Physical name of the procedure
BUSINESS_NAME	VARCHAR2(1000)	Business name of the procedure
DESCRIPTION	VARCHAR2(4000)	Description of the procedure
SIGNATURE	VARCHAR2(4000)	Signature of procedure
FUNCTION_TYPE	CHAR(9)	Type of the function
IS_VALID	VARCHAR2(13)	Is procedure valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2-61 ALL_IV_REF_CURSORS

Column Name	Data Type	Description
LIBRARY_ID	NUMBER(9)	ID of the library
LIBRARY_NAME	VARCHAR2(255)	Name of the library
CURSOR_ID	NUMBER(9)	ID of the cursor
CURSOR_NAME	VARCHAR2(255)	Name of the cursor
BUSINESS_NAME	VARCHAR2(1000)	Business name of the cursor
DESCRIPTION	VARCHAR2(4000)	Description of the cursor
CURSOR_TYPE	VARCHAR2(40)	Type of the cursor
RETURN_RECORD_ID	NUMBER(9)	Ref Cursor Row Type Id
RETURN_RECORD_NAME	VARCHAR2(255)	Ref Cursor Row Type Name
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2-62 ALL_IV_DIM_ATTRIBUTES

Column Name	Data Type	Description
DIMENSION_ID	NUMBER(9)	ID of the dimension
DIMENSION_NAME	VARCHAR2(255)	Name of the dimension
ATTRIBUTE_ID	NUMBER(9)	ID of the attribute

Table 2–62 (Cont.) ALL_IV_DIM_ATTRIBUTES

Column Name	Data Type	Description
ATTRIBUTE_NAME	VARCHAR2(255)	Name of the attribute
BUSINESS_NAME	VARCHAR2(1000)	Business name of the attribute
DESCRIPTION	VARCHAR2(4000)	Description of the attribute
POSITION	NUMBER(9)	Position of the dimension attribute
DATA_TYPE	VARCHAR2(255)	Data type of the dimension attribute
LENGTH	NUMBER(9)	Length for data types of the dimension attribute
PRECISION	NUMBER(9)	Precision for data types of the dimension attribute
SCALE	NUMBER(9)	Scale for data types of the dimension attribute
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–63 ALL_IV_DIM_ROLES

Column Name	Data Type	Description
DIMENSION_ID	NUMBER(9)	ID of the dimension
DIMENSION_NAME	VARCHAR2(255)	Name of the dimension
ROLE_ID	NUMBER(9)	ID of the role
ROLE_NAME	VARCHAR2(255)	Name of the role
BUSINESS_NAME	VARCHAR2(1000)	Business name of the role
DESCRIPTION	VARCHAR2(4000)	Description of the role
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–64 ALL_IV_TM_SCHEMAS

Column Name	Data Type	Description
TM_TABLESPACE_ID	NUMBER(9)	ID of the owning tablespace
TM_TABLESPACE_NAME	VARCHAR2(255)	Internal name of the owning tablespace
TM_SCHEMA_ID	NUMBER(9)	ID of the schema
TM_SCHEMA_NAME	VARCHAR2(255)	Internal unique name for the schema
TM_SCHEMA_UI_NAME	VARCHAR2(1002)	Schema name of the schema in source database. This may not be unique
BUSINESS_NAME	VARCHAR2(1000)	Business name of the schema
DESCRIPTION	VARCHAR2(4000)	Description of the schema
IS_VALID	VARCHAR2(13)	Not used

Table 2–64 (Cont.) ALL_IV_TM_SCHEMAS

Column Name	Data Type	Description
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–65 ALL_IV_TM_TABLESPACES

Column Name	Data Type	Description
TM_ID	NUMBER(9)	ID of the transportable module
TM_NAME	VARCHAR2(255)	Name of the transportable module
TM_TABLESPACE_ID	NUMBER(9)	Internal ID assigned to tablespace within the transportable module
TM_TABLESPACE_NAME	VARCHAR2(255)	Internal name assigned to tablespace within the transportable module. The internal tablespace name is unique
BUSINESS_NAME	VARCHAR2(1000)	The full specification of the tablespace. The format of this value is <Internal tablespace name>:<Source Host>:<Source Port>:<Source Service>:<Source Tablespace Name>
TM_TABLESPACE_UI_NAME	VARCHAR2(1000)	The source tablespace name that this tablespace represent. The tablespace UI name is not guaranteed to be unique
DESCRIPTION	VARCHAR2(4000)	Description of the tablespace
IS_VALID	VARCHAR2(13)	Not used
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user
EXPANDABLE	CHAR(1)	Not used
BLOCKSIZE	NUMBER(9)	Not used

Table 2–66 ALL_IV_CUBE_IMPLS

Column Name	Data Type	Description
IMPLEMENTATION_ID	NUMBER(9)	ID of this cube (this column will be updated in future)
ITEM_ID	NUMBER(9)	ID of the item belonging to this cube
ITEM_TYPE	VARCHAR2(18)	Two kinds of type: cube measures and foreign keys pointing to dimension (called: cube dimension use)
ITEM_NAME	VARCHAR2(255)	Physical name of the item
CUBE_ID	NUMBER(9)	ID of this cube
CUBE_NAME	VARCHAR2(255)	Physical name of this cube
DIMENSION_ID	NUMBER(9)	ID of the associated dimension
DIMENSION_NAME	VARCHAR2(255)	Physical name of the associated name

Table 2–66 (Cont.) ALL_IV_CUBE_IMPLS

Column Name	Data Type	Description
DIMENSION_ALIAS	VARCHAR2(255)	Alias of the associated dimension (name of the foreign key in the cube)
COLUMN_ID	NUMBER(9)	ID of the implementing column for the item
COLUMN_NAME	VARCHAR2(255)	Physical name of the implementing column for the item
POSITION	NUMBER	Position of the implementing column
TABLE_ID	NUMBER(9)	ID of the implementing table for this cube
TABLE_NAME	VARCHAR2(255)	Physical name of the implementing table
FOREIGN_KEY_ID	NUMBER(9)	ID of the foreign key pointing to the dimension
FOREIGN_KEY_NAME	VARCHAR2(255)	Physical name of the foreign key
DIM_IMPLEMENTATION_ID	NUMBER(9)	Current value set to NULL, will be updated in future

Table 2–67 ALL_IV_DIM_IMPLS

Column Name	Data Type	Description
IMPLEMENTATION_ID	NUMBER(9)	ID of the item belonging to this dimension (this column will be updated in future)
LEVEL_ID	NUMBER(9)	ID of the level
DIMENSION_ID	NUMBER(9)	ID of the dimension
DIMENSION_NAME	VARCHAR2(255)	Physical name of the dimension
LEVEL_NAME	VARCHAR2(255)	Name of the level
TABLE_NAME	VARCHAR2(255)	Name of the table

Table 2–68 ALL_IV_DIM_LEVEL_IMPLS

Column Name	Data Type	Description
IMPLEMENTATION_ID	NUMBER(9)	ID of the item belonging to this level (this column will be updated in future)
ITEM_ID	NUMBER(9)	ID of the item belonging to this level
ITEM_TYPE	VARCHAR2(18)	Type of the item (constant value: Level Attribute)
ITEM_NAME	VARCHAR2(255)	Physical name of the item
DIMENSION_ID	NUMBER(9)	ID of the dimension
DIMENSION_NAME	VARCHAR2(255)	Physical name of the dimension
LEVEL_ID	NUMBER(9)	ID of this level
LEVEL_NAME	VARCHAR2(255)	Physical name of this level
COLUMN_ID	NUMBER(9)	ID of the implementation column
COLUMN_NAME	VARCHAR2(255)	Physical name of the implementation column
TABLE_ID	NUMBER(9)	ID of the implementation table for this level
TABLE_NAME	VARCHAR2(255)	Physical name of the implementation table for this level

Table 2-69 ALL_IV_NESTED_TABLES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
NESTED_TABLE_ID	NUMBER(9)	ID of the nested table
NESTED_TABLE_NAME	VARCHAR2(255)	Name of the nested table
BUSINESS_NAME	VARCHAR2(1000)	Business name of the nested table
DESCRIPTION	VARCHAR2(4000)	Description of the nested table
IS_VALID	VARCHAR2(13)	Is the nested table valid
BASE_ELEMENT_NAME	VARCHAR2(767)	Name of the base element
BASE_ELEMENT_PRECISION	NUMBER	Precision of the base element
BASE_ELEMENT_SCALE	NUMBER	Scale of the base element
BASE_ELEMENT_LENGTH	NUMBER	Length of the base element
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Flat Files Views

Table 2-70 ALL_IV_FIELDS

Column Name	Data Type	Description
RECORD_ID	NUMBER(9)	ID of the record this field belongs to
RECORD_NAME	VARCHAR2(255)	Physical name of the record
FIELD_ID	NUMBER(9)	ID of this field
FIELD_NAME	VARCHAR2(255)	Physical name of this field
BUSINESS_NAME	VARCHAR2(4000)	Business name of this field
DESCRIPTION	VARCHAR2(4000)	Description of this field
POSITION	NUMBER(9)	Position of this field
DATA_TYPE	VARCHAR2(255)	Data type of this field
LENGTH	NUMBER(9)	Data length of this field
PRECISION	NUMBER(9)	Data precision of this field
SCALE	NUMBER(9)	Data scale of this field
PICTURE	VARCHAR2(40)	Picture of the field
SIGN_TYPE	NUMBER(9)	Sign type of the field
USAGE	VARCHAR2(40)	Usage of the field
MASK	VARCHAR2(255)	Mask of the field
NULLIF	VARCHAR2(40)	Nullif value of the field
DEFAULTIF	VARCHAR2(40)	Defaultif value of the field

Table 2–70 (Cont.) ALL_IV_FIELDS

Column Name	Data Type	Description
SQL_DATA_TYPE	VARCHAR2(40)	SQL data type of the field
SQL_LENGTH	NUMBER(9)	SQL data length of the field
SQL_PRECISION	NUMBER(9)	SQL precision of the field
SQL_SCALE	NUMBER(9)	SQL data scale of the field
START_POSITION	VARCHAR2(40)	Start position of the field
OCCURS	NUMBER(9)	Occurs of the attribute array within the structure (identified by next column)
STRUCTURE_ID	NUMBER(9)	ID of the structure containing field
STRUCTURE_NAME	VARCHAR2(255)	Physical name of the structure
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–71 ALL_IV_FILES

Column Name	Data Type	Description
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the module this file belongs to
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of the module
FILE_ID	NUMBER(9)	ID of this file
FILE_NAME	VARCHAR2(255)	Physical name of this file
BUSINESS_NAME	VARCHAR2(4000)	Business name of this file
DESCRIPTION	VARCHAR2(4000)	Description of the file
FILE_FORMAT	VARCHAR2(10)	Format of this file
IS_VALID	VARCHAR2(13)	Is this file valid
RECORD_CLASSIFIER_POSITION	NUMBER(9)	Record classifier position of this file
RECORD_CLASSIFIER_LENGTH	NUMBER(9)	Record classifier length of this file
RECORD_SIZE	VARCHAR2(40)	Record size of this file
N_PHYSICAL_RECORDS_IN_LOGICAL	NUMBER(9)	Number of physical records for each logical record
CONTINUATION_AT_END	CHAR(1)	Continuation at end or not
CONTINUATION_DELIMITER	VARCHAR2(40)	Continuation delimiter symbol
RECORD_DELIMITER	VARCHAR2(40)	Record delimiter symbol
FIELD_DELIMITER	VARCHAR2(40)	Field delimiter symbol
TEXT_START_DELIMITER	VARCHAR2(1)	Text start delimiter symbol
TEXT_END_DELIMITER	VARCHAR2(1)	Text end delimiter symbol
SOURCE_FROM	VARCHAR2(4000)	Directory path of this file
UPDATED_ON	DATE	Update timestamp

Table 2–71 (Cont.) ALL_IV_FILES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–72 ALL_IV_RECORDS

Column Name	Data Type	Description
FILE_ID	NUMBER(9)	ID of the file this record belongs to
FILE_NAME	VARCHAR2(255)	Physical name of the file
RECORD_ID	NUMBER(9)	ID of this record
RECORD_NAME	VARCHAR2(255)	Physical name of this record
BUSINESS_NAME	VARCHAR2(4000)	Business name of this record
DESCRIPTION	VARCHAR2(4000)	Description of this record
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Collections Views

Table 2–73 ALL_IV_COLLECTIONS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project this collection belongs to (this view replaces old classification view)
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
COLLECTION_ID	NUMBER(9)	ID of this collection
COLLECTION_NAME	VARCHAR2(255)	Physical name of this collection
BUSINESS_NAME	VARCHAR2(4000)	Business name of this collection
DESCRIPTION	VARCHAR2(4000)	Description of this collection
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-74 ALL_IV_COLLECTION_REFERENCES

Column Name	Data Type	Description
COLLECTION_ID	NUMBER(9)	ID of the collection this reference belongs to (this view replaces old classification_item view)
COLLECTION_NAME	VARCHAR2(255)	Physical name of the collection
COLLECTION_REFERENCE_ID	NUMBER(9)	ID of this collection reference
COLLECTION_REFERENCE_TYPE	VARCHAR2(4000)	Type of this collection reference
COLLECTION_REFERENCE_NAME	VARCHAR2(255)	Physical name of the collection reference
BUSINESS_NAME	VARCHAR2(4000)	Business name of this collection reference
DESCRIPTION	VARCHAR2(4000)	Description of this collection reference
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Function Model Views

Table 2-75 ALL_IV_FUNCTIONS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this function belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
FUNCTION_LIBRARY_ID	NUMBER(9)	ID of the function library this function belongs to
FUNCTION_LIBRARY_NAME	VARCHAR2(255)	Physical name of the function library
FUNCTION_ID	NUMBER(9)	ID of this function
FUNCTION_NAME	VARCHAR2(255)	Physical name of this function
FUNCTION_TYPE	VARCHAR2(13)	Type of this function (function, procedure, table function)
SIGNATURE	VARCHAR2(4000)	Signature of this function
IS_VALID	VARCHAR2(13)	Is this function valid
BUSINESS_NAME	VARCHAR2(4000)	Business name of this function
DESCRIPTION	VARCHAR2(4000)	Description of this function
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-76 ALL_IV_FUNCTION_LIBRARIES

Column Name	Data Type	Description
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the module this function library belongs to
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of the module
FUNCTION_LIBRARY_ID	NUMBER(9)	ID of this function library
FUNCTION_LIBRARY_NAME	VARCHAR2(255)	Physical name of this function library
FUNCTION_LIBRARY_TYPE	VARCHAR2(40)	Type of this function library
IS_VALID	VARCHAR2(13)	Is this function library valid
BUSINESS_NAME	VARCHAR2(4000)	Business name of this function library
DESCRIPTION	VARCHAR2(4000)	Description of this function library
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-77 ALL_IV_FUNCTION_PARAMETERS

Column Name	Data Type	Description
FUNCTION_ID	NUMBER(9)	ID of the function this parameter belongs to
FUNCTION_NAME	VARCHAR2(255)	Physical name of the function
PARAMETER_ID	NUMBER(9)	ID of this parameter
PARAMETER_NAME	VARCHAR2(255)	Physical name of this parameter
PARAMETER_TYPE	VARCHAR2(40)	Type of this parameter
BUSINESS_NAME	VARCHAR2(4000)	Business name of this parameter
DESCRIPTION	VARCHAR2(4000)	Description of this parameter
POSITION	NUMBER(9)	Position of this parameter within the function
DATA_TYPE	VARCHAR2(255)	Data type of this parameter
LENGTH	NUMBER(9)	Data length of this parameter
PRECISION	NUMBER(9)	Data precision of this parameter
SCALE	NUMBER(9)	Data scale of this parameter
DEFAULT_VALUE	VARCHAR2(4000)	Default value of this parameter
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-78 ALL_IV_TABLE_FUNCTIONS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this table function belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
FUNCTION_LIBRARY_ID	NUMBER(9)	ID of the function library this table function belongs to
FUNCTION_LIBRARY_NAME	VARCHAR2(255)	Physical name of the function library
FUNCTION_ID	NUMBER(9)	ID of this table function
FUNCTION_NAME	VARCHAR2(255)	Physical name of this table function
FUNCTION_TYPE	VARCHAR2(13)	Type of this table function (constant value: Table Function)
SIGNATURE	VARCHAR2(4000)	Signature of this table function
IS_VALID	VARCHAR2(13)	Is this table function valid
BUSINESS_NAME	VARCHAR2(4000)	Business name of this table function
DESCRIPTION	VARCHAR2(4000)	Description of this table function
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2-79 ALL_IV_FUNCTION_IMPLS

Column Name	Data Type	Description
FUNCTION_ID	NUMBER(9)	ID of the function
FUNCTION_NAME	VARCHAR2(255)	Physical name of the function
FUNCTION_IMPLEMENTATION_ID	NUMBER(9)	ID of the function implementation for this function
FUNCTION_IMPLEMENTATION_NAME	VARCHAR2(255)	Physical name of the function implementation
LANGUAGE	VARCHAR2(255)	Name of the language being used in the implementation
SCRIPT	VARCHAR2(4000)	Implementation script for this function
BUSINESS_NAME	VARCHAR2(4000)	Business name for this implementation
DESCRIPTION	VARCHAR2(4000)	Description for this implementation
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Configuration Model Views

Table 2–80 ALL_IV_OBJECT_CONFIGURATIONS

Column Name	Data Type	Description
CONFIGURED_OBJECT_ID	NUMBER(9)	ID of the object being configured
CONFIGURED_OBJECT_NAME	VARCHAR2(255)	Physical name of the object
CONFIGURED_OBJECT_TYPE	VARCHAR2(4000)	Type of the object
CONFIGURATION_PARAMETER_KEY	VARCHAR2(128)	Key of configuration parameter
CONFIGURATION_PARAMETER_NAME	VARCHAR2(64)	Name of configuration parameter
PARAMETER-NLSKEY	VARCHAR2(64)	National Language Support (NLS) key of the parameter
CONFIGURATION_PARAMETER_TYPE	CHAR(23)	Type of the configuration parameter
ARGUMENT	VARCHAR2(128)	Value of the configuration parameter
GROUP_NAME	VARCHAR2(322)	Name of the configuration group
GROUP-NLSKEY	VARCHAR2(64)	National Language Support (NLS) key of the configuration group
LANGUAGE	VARCHAR2(64)	Name of the language being used for this configuration

Table 2–81 ALL_IV_CONFIGURATIONS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
CONFIGURATION_ID	NUMBER(9)	ID of the configuration
CONFIGURATION_NAME	VARCHAR2(255)	Name of the configuration
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–82 ALL_IV_CONTROL_CENTERS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
CONTROL_CENTER_ID	NUMBER(9)	ID of the control center
CONTROL_CENTER_NAME	VARCHAR2(255)	Name of the control center
BUSINESS_NAME	VARCHAR2(1000)	Business name of the control center
DESCRIPTION	VARCHAR2(4000)	Description of the control center
HOST	VARCHAR2(255)	Host of the control center
SERVICE_NAME	VARCHAR2(4000)	Service name of the control center
PORT	NUMBER	Port of the control center

Table 2–82 (Cont.) ALL_IV_CONTROL_CENTERS

Column Name	Data Type	Description
USERNAME	VARCHAR2(4000)	User name who will connect to the control center
SCHEMA	VARCHAR2(40)	Schema for the control center
IS_VALID	VARCHAR2(13)	The result of the last validation performed against the control center
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Deployment Model Views

Table 2–83 ALL_IV_CONNECTORS

Column Name	Data Type	Description
LOCATION_ID	NUMBER(9)	ID of the location owning this connector
LOCATION_NAME	VARCHAR2(255)	Physical name of the location
CONNECTOR_ID	NUMBER(9)	ID of this connector
CONNECTOR_NAME	VARCHAR2(255)	Physical name of this connector
BUSINESS_NAME	VARCHAR2(4000)	Business name of this connector
DESCRIPTION	VARCHAR2(4000)	Description of this connector
REFERENCED_LOCATION_ID	NUMBER(9)	ID of the location this connector references to
REFERENCED_LOCATION_NAME	VARCHAR2(255)	Physical name of the location this connector references to
IS_VALID	VARCHAR2(13)	Is this connector valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–84 ALL_IV_LOCATIONS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project this location belongs to
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
LOCATION_ID	NUMBER(9)	ID of this location
LOCATION_NAME	VARCHAR2(255)	Physical name of this location
LOCATION_TARGET_TYPE	VARCHAR2(40)	Target type of this location
LOCATION_TARGET_VERSION	VARCHAR2(40)	Target version of this location
APPLICATION_TYPE	VARCHAR2(255)	Application type of the location connected to
SYSTEM_TYPE	VARCHAR2(255)	System type of this location connected to

Table 2–84 (Cont.) ALL_IV_LOCATIONS

Column Name	Data Type	Description
IS_VALID	VARCHAR2(13)	Is this location valid
BUSINESS_NAME	VARCHAR2(4000)	Business name of this location
DESCRIPTION	VARCHAR2(4000)	Description of this location
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–85 ALL_IV_RUNTIME_REPOSITORIES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project this repository belongs to (also called runtime location, or simply, location)
PROJECT_NAME	VARCHAR2(255)	Physical name of the project
LOCATION_ID	NUMBER(9)	ID of this runtime location
LOCATION_NAME	VARCHAR2(255)	Physical name of this runtime location
LOCATION_TYPE	VARCHAR2(255)	Type of this runtime location
APPLICATION_TYPE	VARCHAR2(255)	Type of the application this location connected to
SYSTEM_TYPE	VARCHAR2(255)	Type of the system this location connected to
BUSINESS_NAME	VARCHAR2(4000)	Business name of this runtime location
DESCRIPTION	VARCHAR2(4000)	Description of this runtime location
HOST	VARCHAR2(40)	Host name of the connection for this location
SERVICE_NAME	VARCHAR2(40)	Service name of the connection for this location
PORT	NUMBER(9)	Port of the connection for this location
USERNAME	VARCHAR2(40)	User name of the connection for this location
SCHEMA	VARCHAR2(40)	Schema name of the connection for this location
IS_VALID	VARCHAR2(13)	Is this runtime location valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Mapping Model Views

Table 2–86 ALL_IV_XFORM_MAPS

Column Name	Data Type	Description
INFORMATION_SYSTEM_ID	NUMBER(9)	ID of the module this map belongs to
INFORMATION_SYSTEM_NAME	VARCHAR2(255)	Physical name of the module
MAP_ID	NUMBER(9)	ID of this map

Table 2–86 (Cont.) ALL_IV_XFORM_MAPS

Column Name	Data Type	Description
MAP_NAME	VARCHAR2(255)	Physical name of this map
BUSINESS_NAME	VARCHAR2(4000)	Business name of this map
DESCRIPTION	VARCHAR2(4000)	Description of this map
COMPOSITE_MAP_COMPONENT_ID	NUMBER(9)	ID of this map (this column is redundant, will be removed in future)
COMPOSITE_MAP_COMPONENT_NAME	VARCHAR2(255)	Physical name of this map (this column is redundant, will be removed in future)
IS_VALID	VARCHAR2(13)	Is this map valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–87 ALL_IV_XFORM_MAP_COMPONENTS

Column Name	Data Type	Description
MAP_ID	NUMBER(9)	ID of the map this map component belongs to
MAP_NAME	VARCHAR2(255)	Physical name of the map
MAP_COMPONENT_ID	NUMBER(9)	ID of this map component (also called map operator)
MAP_COMPONENT_NAME	VARCHAR2(255)	Physical name of this map component
BUSINESS_NAME	VARCHAR2(4000)	Business name of this map component
DESCRIPTION	VARCHAR2(4000)	Description of this map component
OPERATOR_TYPE	VARCHAR2(4000)	Type of this map component (for example, Filter, Joiner, Table)
COMPOSITE_MAP_COMPONENT_ID	NUMBER(9)	ID of the map (this column is redundant, will be removed in future)
COMPOSITE_MAP_COMPONENT_NAME	VARCHAR2(255)	Physical name of the map (this column is redundant, will be removed in future)
DATA_ENTITY_ID	NUMBER(9)	ID of the data entity this map component synchronized to
DATA_ENTITY_NAME	VARCHAR2(255)	Physical name of the data entity
DATA_ENTITY_TYPE	VARCHAR2(4000)	Type of the data entity
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–88 ALL_IV_XFORM_MAP_PARAMETERS

Column Name	Data Type	Description
MAP_COMPONENT_ID	NUMBER(9)	ID of the map component this parameter belongs to
MAP_COMPONENT_NAME	VARCHAR2(255)	Physical name of the map component
PARAMETER_ID	NUMBER(9)	ID of this parameter
PARAMETER_NAME	VARCHAR2(255)	Physical name of this parameter
BUSINESS_NAME	VARCHAR2(4000)	Business name of this parameter
DESCRIPTION	VARCHAR2(4000)	Description of this parameter
MAP_ID	NUMBER(9)	ID of the map containing the map component
MAP_NAME	VARCHAR2(255)	Physical name of the map
PARAMETER_GROUP_NAME	VARCHAR2(255)	Physical name of the parameter group name
PARAMETER_GROUP_ID	NUMBER(9)	ID of the parameter group
PARAMETER_TYPE	VARCHAR2(5)	Type of the parameter (IN, OUT, INOUT)
POSITION	NUMBER(9)	Position of the parameter within the group
DATA_TYPE	VARCHAR2(40)	Data type of the parameter
TRANSFORMATION_EXPRESSION	VARCHAR2(4000)	Textual expression of the transformation for this parameter
DATA_ITEM_ID	NUMBER(9)	ID of the data item this parameter synchronized to
DATA_ITEM_TYPE	VARCHAR2(40)	Type of the data item
DATA_ITEM_NAME	VARCHAR2(255)	Physical name of the data item
SOURCE_PARAMETER_ID	NUMBER(9)	ID of the source parameter (where this parameter connected from)
SOURCE_PARAMETER_NAME	VARCHAR2(255)	Physical name of the source parameter
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–89 ALL_IV_XFORM_MAP_PROPERTIES

Column Name	Data Type	Description
MAP_COMPONENT_ID	NUMBER(9)	ID of the map component this property belongs to
MAP_COMPONENT_NAME	VARCHAR2(255)	Physical name of the map component
PROPERTY_ID	NUMBER(9)	ID of this property
PROPERTY_NAME	VARCHAR2(255)	Physical name of this property
BUSINESS_NAME	VARCHAR2(4000)	Business name of this property
DESCRIPTION	VARCHAR2(4000)	Description of this property
PROPERTY_GROUP_NAME	VARCHAR2(255)	Physical name of this property group
PROPERTY_VALUE	VARCHAR2(4000)	Value of this property

Table 2–90 ALL_IV_XFORM_MAP_DETAILS

Column Name	Data Type	Description
MAP_COMPONENT_ID	NUMBER(9)	ID of the map component
MAP_COMPONENT_NAME	VARCHAR2(255)	Name of the map component
PARAMETER_NAME	VARCHAR2(255)	Name of the parameter
PARAMETER_ID	NUMBER(9)	ID of the parameter
POSITION	NUMBER(9)	Position
BUSINESS_NAME	VARCHAR2(1000)	Business name of map component
TRANSFORMATION_EXPRESSION	VARCHAR2(4000)	Transformation expression
DESCRIPTION	VARCHAR2(4000)	Description of map component
SOURCE_EXPRESSION	VARCHAR2(4000)	Source expression

Table 2–91 ALL_IV_PLUGGABLE_MAPS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
MAP_LIBRARY_ID	NUMBER	ID of the map library
MAP_LIBRARY_NAME	VARCHAR2(255)	Name of the map library
MAP_ID	NUMBER(9)	ID of the map
MAP_NAME	VARCHAR2(255)	Name of the map
BUSINESS_NAME	VARCHAR2(1000)	Business name of the map
DESCRIPTION	VARCHAR2(4000)	Description of the map
IS_VALID	VARCHAR2(13)	Is the map valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–92 ALL_IV_PLUGGABLE_MAP_LIBRARIES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
MAP_LIBRARY_ID	NUMBER(9)	ID of the map library
MAP_LIBRARY_NAME	VARCHAR2(255)	Name of the map library
BUSINESS_NAME	VARCHAR2(1000)	Business name of the map library
DESCRIPTION	VARCHAR2(4000)	Description of the map library
IS_VALID	VARCHAR2(13)	Is the library valid
UPDATED_ON	DATE	Update timestamp

Table 2–92 (Cont.) ALL_IV_PLUGGABLE_MAP_LIBRARIES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–93 ALL_IV_PLUG_MAP_PARAMETERS

Column Name	Data Type	Description
MAP_COMPONENT_ID	NUMBER(9)	ID of the map component
MAP_COMPONENT_NAME	VARCHAR2(255)	Name of the map component
PARAMETER_ID	NUMBER(9)	ID of the parameter
PARAMETER_NAME	VARCHAR2(255)	Name of the parameter
BUSINESS_NAME	VARCHAR2(1000)	Business name of the parameter
DESCRIPTION	VARCHAR2(4000)	Description of the parameter
MAP_ID	NUMBER(9)	ID of the map
MAP_NAME	VARCHAR2(255)	Name of the map
PARAMETER_GROUP_NAME	VARCHAR2(255)	Name of the parameter group
PARAMETER_GROUP_ID	NUMBER(9)	ID of the parameter group
PARAMETER_TYPE	VARCHAR2(5)	Type of the parameter
POSITION	NUMBER(9)	The position of the parameter
DATA_TYPE	VARCHAR2(4000)	The data type of the parameter
TRANSFORMATION_EXPRESSION	VARCHAR2(4000)	The expression of the parameter
DATA_ITEM_ID	NUMBER(9)	ID of the data item
DATA_ITEM_TYPE	VARCHAR2	Type of the data item
DATA_ITEM_NAME	VARCHAR2(255)	Name of the data item
SOURCE_PARAMETER_ID	NUMBER(9)	The ID of the parameter where this parameter is connected from
SOURCE_PARAMETER_NAME	VARCHAR2(255)	The name of the parameter where this parameter is connected from
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–94 ALL_IV_PLUG_MAP_COMPONENTS

Column Name	Data Type	Description
MAP_ID	NUMBER(9)	ID of the map
MAP_NAME	VARCHAR2(255)	Name of the map
MAP_COMPONENT_ID	NUMBER(9)	ID of the map component
MAP_COMPONENT_NAME	VARCHAR2(255)	Name of the map component

Table 2–94 (Cont.) ALL_IV_PLUG_MAP_COMPONENTS

Column Name	Data Type	Description
BUSINESS_NAME	VARCHAR2(1000)	Business name of the map component
DESCRIPTION	VARCHAR2(4000)	Description of the map component
OPERATOR_TYPE	VARCHAR2(4000)	The operator type of the map component
COMPOSITE_COMPONENT_ID	NUMBER(9)	ID of the map component
COMPOSITE_COMPONENT_NAME	VARCHAR2(255)	Name of the map component
DATA_ENTITY_ID	NUMBER(9)	ID of the data entity
DATA_ENTITY_TYPE	VARCHAR2(4000)	Type of data entity
DATA_ENTITY_NAME	VARCHAR2(255)	Name of the data entity
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Process Flow Model Views

Table 2–95 ALL_IV_PACKAGES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the module this process package belongs to
SCHEMA_NAME	VARCHAR2(255)	Physical name of the module
PACKAGE_ID	NUMBER(9)	ID of this process package
PACKAGE_NAME	VARCHAR2(255)	Physical name of this process package
BUSINESS_NAME	VARCHAR2(4000)	Business name of this process package
DESCRIPTION	VARCHAR2(4000)	Description of this process package
IS_VALID	VARCHAR2(13)	Is this process package valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–96 ALL_IV_PROCESSES

Column Name	Data Type	Description
PACKAGE_ID	NUMBER(9)	ID of the process package this process belongs to
PACKAGE_NAME	VARCHAR2(255)	Physical name of the process package
PARENT_PROCESS_ID	NUMBER(9)	ID of the parent process for this process
PARENT_PROCESS_NAME	VARCHAR2(255)	Physical name of the parent process for this process
PROCESS_ID	NUMBER(9)	ID of this process
PROCESS_NAME	VARCHAR2(255)	Physical name of this process
BUSINESS_NAME	VARCHAR2(4000)	Business name of this process

Table 2–96 (Cont.) ALL_IV_PROCESSES

Column Name	Data Type	Description
DESCRIPTION	VARCHAR2(4000)	Description of this process
BOUND_OBJECT_ID	NUMBER(9)	ID of the bound object
BOUND_OBJECT_NAME	VARCHAR2(255)	Name of the bound object
IS_VALID	VARCHAR2(13)	Is this process valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–97 ALL_IV_PROCESS_ACTIVITIES

Column Name	Data Type	Description
PROCESS_ID	NUMBER(9)	ID of the process this activity belongs to
PROCESS_NAME	VARCHAR2(255)	Physical name of the process
ACTIVITY_ID	NUMBER(9)	ID of this process activity
ACTIVITY_NAME	VARCHAR2(255)	Physical name of this activity
ACTIVITY_TYPE	VARCHAR2(4000)	Type of this activity
BOUND_OBJECT_ID	NUMBER(9)	ID of the bound object
BOUND_OBJECT_NAME	VARCHAR2(255)	Name of the bound object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–98 ALL_IV_PROCESS_PARAMETERS

Column Name	Data Type	Description
PARAMETER_OWNER_ID	NUMBER(9)	ID of the owning object for this parameter
PARAMETER_OWNER_NAME	VARCHAR2(255)	Physical name of the owning object for this parameter
PARAMETER_OWNER_TYPE	CHAR(14)	Type of the owning object
PARAMETER_ID	NUMBER(9)	ID of this parameter
PARAMETER_NAME	VARCHAR2(255)	Physical name of this parameter
BUSINESS_NAME	VARCHAR2(4000)	Business name of this parameter
DESCRIPTION	VARCHAR2(4000)	Description of this parameter
POSITION	NUMBER(9)	Position of this parameter
DATA_TYPE	VARCHAR2(40)	Data type of this parameter
DEFAULT_VALUE	VARCHAR2(4000)	Default value for this parameter
DIRECTION	VARCHAR2(3)	Direction of this parameter (IN, OUT)

Table 2–98 (Cont.) ALL_IV_PROCESS_PARAMETERS

Column Name	Data Type	Description
IS_FINAL	CHAR(1)	Is process final
BOUNDDATA_ID	NUMBER(9)	ID of the bound data for this parameter
BOUNDDATA_NAME	VARCHAR2(255)	Physical name of the bound data
BOUNDDATA_TYPE	VARCHAR2(40)	Type of the bound data
BOUNDDATA_VALUE	VARCHAR2(4000)	Value of the bound data
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–99 ALL_IV_PROCESS_TRANSITIONS

Column Name	Data Type	Description
PROCESS_ID	NUMBER(9)	ID of the process this transition belongs to
PROCESS_NAME	VARCHAR2(255)	Physical name of the process
TRANSITION_ID	NUMBER(9)	ID of this transition
TRANSITION_NAME	VARCHAR2(255)	Physical name of this transition
BUSINESS_NAME	VARCHAR2(4000)	Business name of this transition
DESCRIPTION	VARCHAR2(4000)	Description of this transition
CONDITION	VARCHAR2(255)	Condition of this transition
TRANSITION_ORDER	NUMBER(9)	Order of this transition
SOURCE_ACTIVITY_ID	NUMBER(9)	ID of the source activity for this transition
SOURCE_ACTIVITY_NAME	VARCHAR2(255)	Physical name of the source activity
TARGET_ACTIVITY_ID	NUMBER(9)	ID of the target activity for this transition
TARGET_ACTIVITY_NAME	VARCHAR2(255)	Physical name of the target activity
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–100 ALL_IV_PROCESS_VARIABLES

Column Name	Data Type	Description
PROCESS_ID	NUMBER(9)	ID of the process this variable belongs to
PROCESS_NAME	VARCHAR2(255)	Physical name of the process
VARIABLE_ID	NUMBER(9)	ID of this process variable
VARIABLE_NAME	VARCHAR2(255)	Physical name of this process variable
BUSINESS_NAME	VARCHAR2(4000)	Business name of this process variable
DESCRIPTION	VARCHAR2(4000)	Description of this process variable

Table 2–100 (Cont.) ALL_IV_PROCESS_VARIABLES

Column Name	Data Type	Description
POSITION	NUMBER(9)	Position of this variable
DATA_TYPE	VARCHAR2(40)	Data type of this variable
DEFAULT_VALUE	VARCHAR2(4000)	Default value of this variable
IS_FINAL	CHAR(1)	Is process final
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(255)	Updated by user
CREATED_BY	VARCHAR2(255)	Created by user

Table 2–101 ALL_IV_SUB_PROCESSES

Column Name	Data Type	Description
PACKAGE_ID	NUMBER(9)	ID of the package
PACKAGE_NAME	VARCHAR2(255)	Name of the package
PARENT_PROCESS_ID	NUMBER(9)	ID of the parent process
PARENT_PROCESS_NAME	VARCHAR2(255)	Name of the parent process
PROCESS_ID	NUMBER(9)	ID of the process
PROCESS_NAME	VARCHAR2(255)	Name of the process
BUSINESS_NAME	VARCHAR2(1000)	Business name of the condition
DESCRIPTION	VARCHAR2(4000)	Description of the condition
BOUND_OBJECT_ID	NUMBER(9)	ID of the bound object
BOUND_OBJECT_NAME	VARCHAR2(40)	Name of the bound object
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Data Profiling Views

Table 2–102 ALL_IV_PROFILES

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
BUSINESS_NAME	VARCHAR2(1000)	Business name of the profile
DESCRIPTION	VARCHAR2(4000)	Description of the profile
UPDATED_ON	DATE	Update timestamp

Table 2–102 (Cont.) ALL_IV_PROFILES

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–103 ALL_IV_PROFILE_COLUMNS

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
PROFILE_OBJECT_ID	NUMBER(9)	ID of the profile object
ENTITY_ID	NUMBER(9)	ID of the entity
ENTITY_NAME	VARCHAR2(255)	Name of the entity
PROFILE_COLUMN_ID	NUMBER(9)	ID of the profile column
COLUMN_ID	NUMBER(9)	ID of the column
COLUMN_NAME	VARCHAR2(255)	Name of the column
BUSINESS_NAME	VARCHAR2(1000)	Business name of column
AVG_VALUE	VARCHAR2(40)	Average value of column if column has a numeric data type
MAX_VALUE	VARCHAR2(4000)	Maximum value stored in the column
MIN_VALUE	VARCHAR2(4000)	Minimum value stored in the column
MEDIAN_VALUE	VARCHAR2(4000)	Median value of column if column has a numeric data type
STDDEV_VALUE	VARCHAR2(40)	Standard deviation of column if column has a numeric data type
NUM_NULLS	NUMBER	Number of null values stored in the column
NUM_DISTINCT	NUMBER	Number of distinct values stored in the column
CONSENSUS_DATATYPE	VARCHAR2(40)	The discovered data type for the column
CONSENSUS_DATATYPE_CNT	VARCHAR2(40)	The number of rows that have the consensus data type
CONSENSUS_LENGTH	NUMBER	The predominant length of the column
CONSENSUS_LENGTH_CNT	NUMBER	The number of rows with the predominant length
CONSENSUS_PRECISION	NUMBER	The predominant precision of the column
CONSENSUS_PRECISION_CNT	NUMBER	The number of rows with the predominant precision
CONSENSUS_SCALE	NUMBER	The predominant scale of the column
CONSENSUS_SCALE_CNT	NUMBER	The number of rows with the predominant scale
COMMON_FORMAT	VARCHAR2(40)	The discovered common format

Table 2–103 (Cont.) ALL_IV_PROFILE_COLUMNS

Column Name	Data Type	Description
COMMON_FORMAT_CNT	VARCHAR2(40)	The number of rows that have this discovered common format
DOMINANT_CHARACTER_PATTERN	VARCHAR2(4000)	The discovered pattern at the character level of the column
DOMINANT_CHARACTER_PATTERN_CNT	VARCHAR2(40)	The number of rows that satisfy the character level pattern
DOMINANT_WORD_PATTERN	VARCHAR2(4000)	The discovered pattern at the word level of the column
DOMINANT_WORD_PATTERN_CNT	VARCHAR2(40)	The number of rows that satisfy the word level pattern
MAX_LENGTH	NUMBER	Maximum length of the values stored in the column
MIN_LENGTH	NUMBER	Minimum length of the values stored in the column
MAX_PRECISION	NUMBER	Maximum precision of the values stored in the column
MIN_PRECISION	NUMBER	Minimum precision of the values stored in the column
MAX_SCALE	NUMBER	Maximum scale of the values stored in the column
MIN_SCALE	NUMBER	Minimum scale of the values stored in the column
CFORMAT_COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy the discovered common format
CFORMAT_DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies the discovered common format and which does not
CFORMAT_NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered common format
DATATYPE_COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy the discovered data type
DATATYPE_DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies the discovered data type and which does not
DATATYPE_NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered data type
DOMAIN_COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that contain the discovered domain values
DOMAIN_DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row contains the discovered domain values and which does not
DOMAIN_NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not contain the discovered domain values
NULL_COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that are null based on the configured null value

Table 2–103 (Cont.) ALL_IV_PROFILE_COLUMNS

Column Name	Data Type	Description
NULL_DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row is null and which isn't based on the configured null value
NULL_NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered common format
CHAR_PATTERN_COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy the discovered character pattern
CHAR_PATTERN_DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies the discovered character pattern and which does not
CHAR_PATTERN_NONCOM_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered character pattern
WORD_PATTERN_COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy the discovered word pattern
WORD_PATTERN_DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies the discovered word pattern and which does not
WORD_PATTERN_NONCOM_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered word pattern
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–104 ALL_IV_PROFILE_DOMAIN_VALUES

Column Name	Data Type	Description
PROFILE_OBJECT_ID	NUMBER(9)	ID of the profile object
PROFILE_COLUMN_ID	NUMBER(9)	ID of the profile column
COLUMN_NAME	VARCHAR2(255)	Name of the column
VALUE	VARCHAR2(4000)	Domain value
VALUE_COUNT	VARCHAR2(4000)	Number of rows that have this domain value
COMPLIANT	VARCHAR2(1)	Whether this domain value satisfies the domain value configuration

Table 2–105 ALL_IV_FUNCTIONAL_DEPENDENCIES

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
ENTITY_ID	NUMBER(9)	ID of the entity
ENTITY_NAME	VARCHAR2(255)	Name of the entity

Table 2–105 (Cont.) ALL_IV_FUNCTIONAL_DEPENDENCIES

Column Name	Data Type	Description
FUNCTIONAL_DEPENDENCY_ID	NUMBER(9)	ID of the functional dependency
FUNCTIONAL_DEPENDENCY_NAME	VARCHAR2(255)	Name of the functional dependency
BUSINESS_NAME	VARCHAR2(1000)	Business name of the functional dependency
TYPE	VARCHAR2(15)	Type of functional dependency
DEPENDENT_COLUMN_ID	NUMBER(9)	ID of the dependent column in the functional dependency
FD_ERROR	NUMBER	The number of rows that do not satisfy this functional dependency
COMPLIANT_QUERY	VARCHAR2(4000)	The query that is used to retrieve the rows that satisfy this functional dependency
COMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy this functional dependency
NONCOMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies this functional dependency and which does not
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–106 ALL_IV_PROFILE_FOREIGN_KEYS

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
ENTITY_ID	NUMBER(9)	ID of the entity
ENTITY_NAME	VARCHAR2(255)	Name of the entity
FOREIGN_KEY_ID	NUMBER(9)	ID of the row relationship
FOREIGN_KEY_NAME	VARCHAR2(255)	Name of the row relationship
BUSINESS_NAME	VARCHAR2(1000)	Business name of the foreign key
UNIQUE_KEY_ID	NUMBER(9)	ID of the unique key
IS_DISCOVERED	VARCHAR2(3)	Should a foreign key be created on the column
IS_DOCUMENTED	VARCHAR2(3)	Indicates if a foreign key exists in the data dictionary for the column
LOCAL_MAX_CARDINALITY	VARCHAR2(40)	Maximum number of values found on the local side
LOCAL_MIN_CARDINALITY	VARCHAR2(40)	Minimum number of values found on the local side
REMOTE_MAX_CARDINALITY	VARCHAR2(40)	Maximum number of values found on the remote side
REMOTE_MIN_CARDINALITY	VARCHAR2(40)	Minimum number of values found on the remote side

Table 2–106 (Cont.) ALL_IV_PROFILE_FOREIGN_KEYS

Column Name	Data Type	Description
NUM_ORPHANS	VARCHAR2(40)	Number of distinct values found in the local column but not in the remote column
COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy the discovered row relationship
COMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered common format
NONCOMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies the discovered row relationship and which does not
CHILDLESS_QUERY	VARCHAR2(4000)	Not used
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–107 ALL_IV_PROFILE_KEY_COLUMN_USES

Column Name	Data Type	Description
KEY_ID	NUMBER(9)	ID of the key
KEY_TYPE	VARCHAR2(4000)	Type of the key
KEY_NAME	VARCHAR2(255)	Name of the key
COLUMN_ID	NUMBER(9)	ID of the column
COLUMN_NAME	VARCHAR2(255)	Physical name of the column
BUSINESS_NAME	VARCHAR2(1000)	Business name of the column
POSITION	NUMBER(9)	Position of the column in the set

Table 2–108 ALL_IV_PROFILE_OBJECTS

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
PROFILE_OBJECT_ID	NUMBER(9)	ID of the profile object
OBJECT_ID	NUMBER(9)	ID of the object
PROFILE_OBJECT_NAME	VARCHAR2(255)	Name of the profile object
OBJECT_NAME	VARCHAR2(255)	Physical name of the object
OBJECT_TYPE	VARCHAR2(4000)	Type of the object
BUSINESS_NAME	VARCHAR2(1000)	Business name of the object
DESCRIPTION	VARCHAR2(4000)	Description of the profile object
IS_VALID	VARCHAR2(13)	Is the profile object valid

Table 2–108 (Cont.) ALL_IV_PROFILE_OBJECTS

Column Name	Data Type	Description
ROW_COUNT	NUMBER	Number of rows in the object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–109 ALL_IV_PROFILE_PATTERN_VALUES

Column Name	Data Type	Description
PROFILE_OBJECT_ID	NUMBER(9)	ID of the profile object
PROFILE_COLUMN_ID	NUMBER(9)	ID of the profile column
COLUMN_NAME	VARCHAR2(255)	Name of the column
VALUE	VARCHAR2(4000)	Pattern value
VALUE_COUNT	VARCHAR2(4000)	The number of rows that satisfy this discovered pattern value
TYPE	VARCHAR2(9)	0 for Word Pattern, 1 for Character Pattern
COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy this pattern value
NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy this pattern value
DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies this pattern value and which does not

Table 2–110 ALL_IV_PROFILE_RULES

Column Name	Data Type	Description
PROFILE_OBJECT_ID	NUMBER(9)	ID of the profile object
DATA_RULE_USAGE_ID	NUMBER(9)	ID of the data rule usage
DATA_RULE_USAGE_NAME	VARCHAR2(255)	Name of the data rule usage
COMPLIANT_ROW_COUNT	VARCHAR2(4000)	The number of rows that comply with the data rule

Table 2–111 ALL_IV_PROFILE_UNIQUE_KEYS

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
ENTITY_ID	NUMBER(9)	ID of the entity
ENTITY_NAME	VARCHAR2(255)	Name of the entity
UNIQUE_KEY_ID	NUMBER(9)	ID of the unique key
UNIQUE_KEY_NAME	VARCHAR2(255)	Name of the unique key
BUSINESS_NAME	VARCHAR2(1000)	Business name of the unique key

Table 2–111 (Cont.) ALL_IV_PROFILE_UNIQUE_KEYS

Column Name	Data Type	Description
IS_DISCOVERED	VARCHAR2(3)	Should a unique key be created on the column
IS_DOCUMENTED	VARCHAR2(3)	Indicates if a unique key exists on the column in the data dictionary
UK_ERROR	VARCHAR2(40)	Number of rows that do not satisfy this unique key
UK_PARTITION	VARCHAR2(40)	Not used
COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy this unique key
COMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy this unique key
NONCOMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies this unique key and which does not
CHILDLESS_QUERY	VARCHAR2(4000)	Not used
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Data Rules Views

Table 2–112 ALL_IV_DATA_RULES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
DATARULE_ID	NUMBER(9)	ID of the data rule
DATARULE_NAME	VARCHAR2(255)	Physical name of the data rule
BUSINESS_NAME	VARCHAR2(1000)	Business name of the data rule
DATARULE_TYPE	VARCHAR2(4000)	Type of data rule
DESCRIPTION	VARCHAR2(4000)	Description of the data rule
IS_VALID	VARCHAR2(13)	Not used
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–113 ALL_IV_DATA_RULE_ATTRIBUTES

Column Name	Data Type	Description
DATARULE_ID	NUMBER(9)	ID of the data rule
DATARULE_NAME	VARCHAR2(255)	Name of the data rule
DATARULEGROUP_ID	NUMBER(9)	ID of the owning data rule group
DATARULEGROUP_NAME	VARCHAR2(255)	Name of the owning data rule group
DATARULEATTR_ID	NUMBER(9)	ID of the data rule attribute
NAME	VARCHAR2(255)	Name of the data rule attribute
BUSINESS_NAME	VARCHAR2(1000)	Business name of data rule attribute
DESCRIPTION	VARCHAR2(4000)	Description of data rule attribute
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–114 ALL_IV_DATA_RULE_ATTR_USAGES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
RELATION_ID	NUMBER(9)	ID of owning relation
RELATION_NAME	VARCHAR2(255)	Name of owning relation
DATARULE_USAGE_ID	NUMBER(9)	ID of owning data rule usage
DATARULE_USAGE_NAME	VARCHAR2(255)	Name of owning data rule usage
DATARULE_GROUP_USAGE_ID	NUMBER(9)	ID of owning data rule usage group
DATARULE_GROUP_USAGE_NAME	VARCHAR2(255)	Name of owning data rule usage group
DATARULE_ATTR_USAGE_ID	NUMBER(9)	ID of data rule attribute usage
NAME	VARCHAR2(255)	Name of data rule attribute usage
BUSINESS_NAME	VARCHAR2(1000)	Business name of data rule attribute usage
DESCRIPTION	VARCHAR2(4000)	Description of data rule attribute usage
DATARULE_SCHEMA_ID	NUMBER(9)	ID of data rule folder
DATARULE_SCHEMA_NAME	VARCHAR2(255)	Name of data rule folder
DATARULE_ID	NUMBER(9)	ID of data rule
DATARULE_NAME	VARCHAR2(255)	Name of data rule
DATARULE_GROUP_ID	NUMBER(9)	ID of data rule group
DATARULE_GROUP_NAME	VARCHAR2(255)	Name of data rule group
DATARULE_ATTR_ID	NUMBER(9)	ID of data rule attribute
DATARULE_ATTR_NAME	VARCHAR2(255)	Name of data rule attribute
REFERENCED_SCHEMA	NUMBER(9)	ID of referenced schema of date rule usage group

Table 2–114 (Cont.) ALL_IV_DATA_RULE_ATTR_USAGES

Column Name	Data Type	Description
REFERENCE_SCHEMA_NAME	VARCHAR2(255)	Name of referenced schema of data rule usage group
REFERENCED_RELATION_ID	NUMBER(9)	ID if referenced relation of data rule usage group
REFERENCE_RELATION_NAME	VARCHAR2(255)	Name of referenced relation of data rule usage group
ATTRIBUTE_ID	NUMBER(9)	ID of the attribute
ATTRIBUTE_NAME	VARCHAR2(255)	Name of the attribute
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–115 ALL_IV_DATA_RULE_DOMAINS

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
DATARULE_ID	NUMBER(9)	ID of the data rule
DATARULE_NAME	VARCHAR2(255)	Name of the data rule
PROPERTY_ID	NUMBER(9)	ID of domain property
NAME	VARCHAR2(255)	Value of domain property
BUSINESS_NAME	VARCHAR2(1000)	Not used
DESCRIPTION	VARCHAR2(4000)	Description of the data rule
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–116 ALL_IV_DATA_RULE_GROUPS

Column Name	Data Type	Description
DATARULE_ID	NUMBER(9)	ID of the data rule
DATARULE_NAME	VARCHAR2(255)	Physical name of the data rule
DATARULEGROUP_ID	NUMBER(9)	ID of data rule group
DATARULEGROUP_NAME	VARCHAR2(255)	Name of data rule group
BUSINESS_NAME	VARCHAR2(1000)	Business name of data rule group
DESCRIPTION	VARCHAR2(4000)	Description of the data rule group
UPDATED_ON	DATE	Update timestamp

Table 2–116 (Cont.) ALL_IV_DATA_RULE_GROUPS

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–117 ALL_IV_DATA_RULE_GROUP_USAGES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
RELATION_ID	NUMBER(9)	ID of owning relation
RELATION_NAME	VARCHAR2(255)	Name of owning relation
DATARULE_USAGE_ID	NUMBER(9)	ID of data rule usage
DATARULE_USAGE_NAME	VARCHAR2(255)	Name of data rule usage
DATARULE_GROUP_USAGE_ID	NUMBER(9)	ID of data rule usage group
NAME	VARCHAR2(255)	Name of data rule usage group
BUSINESS_NAME	VARCHAR2(1000)	Business name of data rule usage group
DESCRIPTION	VARCHAR2(4000)	Description of data rule usage group
DATARULE_SCHEMA_ID	NUMBER(9)	ID of referenced data rule folder
DATARULE_SCHEMA_NAME	VARCHAR2(255)	Name of referenced data rule folder
DATARULE_ID	NUMBER(9)	ID of referenced data rule
DATARULE_NAME	VARCHAR2(255)	Name of referenced data rule
DATARULE_GROUP_ID	NUMBER(9)	ID of referenced data rule group
DATARULE_GROUP_NAME	VARCHAR2(255)	Name of referenced data rule group
REFERENCED_SCHEMA	NUMBER(9)	ID of referenced schema
REFERENCE_SCHEMA_NAME	VARCHAR2(255)	Name of referenced schema
REFERENCED_RELATION_ID	NUMBER(9)	ID of referenced relation
REFERENCE_RELATION_NAME	VARCHAR2(255)	Name of referenced relation
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–118 ALL_IV_DATA_RULE_PROPERTIES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
DATARULE_ID	NUMBER(9)	ID of the data rule
DATARULE_NAME	VARCHAR2(255)	Physical name of the data rule

Table 2–118 (Cont.) ALL_IV_DATA_RULE_PROPERTIES

Column Name	Data Type	Description
PROPERTY_ID	NUMBER(9)	ID of data rule property
NAME	VARCHAR2(255)	Name of data rule property
BUSINESS_NAME	VARCHAR2(1000)	Business name of the data rule property
VALUE	VARCHAR2(4000)	Value of data rule property
DESCRIPTION	VARCHAR2(4000)	Description of the data rule property
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–119 ALL_IV_DATA_RULE_USAGES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the owning schema
SCHEMA_NAME	VARCHAR2(255)	Name of the owning schema
RELATION_ID	NUMBER(9)	Owning relation of data rule usage
RELATION_NAME	VARCHAR2(255)	Name of owning relation of data rule usage
DATARULE_USAGE_ID	NUMBER(9)	ID of data rule usage
NAME	VARCHAR2(255)	Name of data rule usage
BUSINESS_NAME	VARCHAR2(1000)	Business name of data rule usage
DESCRIPTION	VARCHAR2(4000)	Description of data rule usage
DATARULE_SCHEMA_ID	NUMBER(9)	Referenced data rule folder ID
DATARULE_SCHEMA_NAME	VARCHAR2(255)	Referenced data rule folder name
DATARULE_ID	NUMBER(9)	Referenced data rule ID
DATARULE_NAME	VARCHAR2(255)	Referenced data rule name
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

User Defined Object Views

Table 2–120 ALL_IV_UDO_FCOS

Column Name	Data Type	Description
OWNER_ID	NUMBER(9)	ID of the owner
OWNER_NAME	VARCHAR2(255)	Name of the owner
FIRST_CLASS_OBJECT_ID	NUMBER(9)	ID of the First Class Object
FIRST_CLASS_OBJECT_NAME	VARCHAR2(255)	Name of the First Class Object
FIRST_CLASS_OBJECT_TYPE	VARCHAR2(255)	Type of the First Class Object

Table 2–120 (Cont.) ALL_IV_UDO_FCOS

Column Name	Data Type	Description
BUSINESS_NAME	VARCHAR2(1000)	Business name of the First Class Object
DESCRIPTION	VARCHAR2(4000)	Description of the First Class Object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–121 ALL_IV_UDO_FOLDERS

Column Name	Data Type	Description
OWNER_ID	NUMBER(9)	ID of the owner
OWNER_NAME	VARCHAR2(255)	Name of the owner
FOLDER_ID	NUMBER(9)	ID of the folder
FOLDER_NAME	VARCHAR2(255)	Name of the folder
FOLDER_TYPE	VARCHAR2(255)	Type of folder
BUSINESS_NAME	VARCHAR2(1000)	Business name of the folder
DESCRIPTION	VARCHAR2(4000)	Description of the folder
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–122 ALL_IV_UDO_SCOS

Column Name	Data Type	Description
OWNER_ID	NUMBER(9)	ID of the owner
OWNER_NAME	VARCHAR2(255)	Name of the owner
SECOND_CLASS_OBJECT_ID	NUMBER(9)	ID of the Second Class Object
SECOND_CLASS_OBJECT_NAME	VARCHAR2(255)	Name of the Second Class Object
SECOND_CLASS_OBJECT_TYPE	VARCHAR2(255)	Type of Second Class Object
BUSINESS_NAME	VARCHAR2(1000)	Business name of the Second Class Object
DESCRIPTION	VARCHAR2(4000)	Description of the Second Class Object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–123 ALL_IV_UDO_ASSOCIATIONS

Column Name	Data Type	Description
OWNER_ID	NUMBER(9)	ID of the owner
OWNER_NAME	VARCHAR2(255)	Name of the owner
OWNER_TYPE	VARCHAR2(255)	Type of the owner
ASSOCIATION_ID	NUMBER(9)	ID of the association
SOURCE_ROLE	VARCHAR2(255)	Role of the owner
TARGET_ROLE	VARCHAR2(255)	Role of the associated object
TARGET_ID	NUMBER(9)	ID of the target
TARGET_NAME	VARCHAR2(255)	Name of the target
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Expert Views

Table 2–124 ALL_IV_EXPERTS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
EXPERT_ID	NUMBER(9)	ID of the expert
EXPERT_NAME	VARCHAR2(255)	Name of the expert
BUSINESS_NAME	VARCHAR2(1000)	Business name of the expert
DESCRIPTION	VARCHAR2(4000)	Description of the expert
BOUND_OBJECT_ID	NUMBER(9)	ID of the bound object
BOUND_OBJECT_NAME	VARCHAR2(40)	Name of the bound object
IS_VALID	VARCHAR2(13)	Is this expert valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–125 ALL_IV_EXPERT_PARAMETERS

Column Name	Data Type	Description
PARAMETER_OWNER_ID	NUMBER(9)	ID of the parameter owner
PARAMETER_OWNER_NAME	VARCHAR2(255)	Name of the parameter owner
PARAMETER_OWNER_TYPE	VARCHAR2(6)	Type of the parameter owner
PARAMETER_ID	NUMBER(9)	ID of the parameter

Table 2–125 (Cont.) ALL_IV_EXPERT_PARAMETERS

Column Name	Data Type	Description
PARAMETER_NAME	VARCHAR2(255)	Name of the parameter
BUSINESS_NAME	VARCHAR2(1000)	Business name of the parameter
DESCRIPTION	VARCHAR2(4000)	Description of the parameter
POSITION	NUMBER(9)	The position of the parameter
DATA_TYPE	VARCHAR2(40)	Data type for this parameter
DEFAULT_VALUE	VARCHAR2(4000)	Default value
DIRECTION	VARCHAR2(5)	Direction of this parameter
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–126 ALL_IV_EXPERT_TASKS

Column Name	Data Type	Description
EXPERT_ID	NUMBER(9)	ID of the expert
EXPERT_NAME	VARCHAR2(255)	Name of the expert
TASK_ID	NUMBER(9)	ID of the task
TASK_NAME	VARCHAR2(255)	Name of the task
BUSINESS_NAME	VARCHAR2(1000)	Business name of the task
DESCRIPTION	VARCHAR2(4000)	Description of the task
TASK_TYPE	VARCHAR2(4000)	Type of the task
BOUND_OBJECT_ID	NUMBER(9)	ID of the bound object
BOUND_OBJECT_NAME	VARCHAR2(40)	Name of the bound object
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user
MAIN	VARCHAR2(4000)	The main processing script of the task
PREPROCESSING	VARCHAR2(4000)	The pre-processing script of the task
POSTPROCESSING	VARCHAR2(4000)	The post-processing script of the task
INSTRUCTION	VARCHAR2(4000)	Instruction for running the task

Table 2–127 ALL_IV_EXPERT_TRANSITIONS

Column Name	Data Type	Description
EXPERT_ID	NUMBER(9)	ID of the expert
EXPERT_NAME	VARCHAR2(255)	Name of the expert
TRANSITION_ID	NUMBER(9)	ID of the transition

Table 2–127 (Cont.) ALL_IV_EXPERT_TRANSITIONS

Column Name	Data Type	Description
TRANSITION_NAME	VARCHAR2(255)	Name of the transition
BUSINESS_NAME	VARCHAR2(1000)	Business name of the transition
DESCRIPTION	VARCHAR2(4000)	Description of the transition
CONDITION	VARCHAR2(4000)	Transition condition
TRANSITION_ORDER	NUMBER(9)	Transition order
SOURCE_ACTIVITY_ID	NUMBER(9)	ID of the source task
SOURCE_ACTIVITY_NAME	VARCHAR2(255)	Name of the source task
TARGET_ACTIVITY_ID	NUMBER(9)	ID of the target task
TARGET_ACTIVITY_NAME	VARCHAR2(255)	Name of the target task
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–128 ALL_IV_EXPERT_VARIABLES

Column Name	Data Type	Description
EXPERT_ID	NUMBER(9)	ID of the expert
EXPERT_NAME	VARCHAR2(255)	Name of the expert
VARIABLE_ID	NUMBER(9)	ID of the variable
VARIABLE_NAME	VARCHAR2(255)	Name of the variable
BUSINESS_NAME	VARCHAR2(1000)	Business name of the variable
DESCRIPTION	VARCHAR2(4000)	Description of the variable
POSITION	NUMBER(9)	Position of the variable
DATA_TYPE	VARCHAR2(40)	Data type of the variable
DEFAULT_VALUE	VARCHAR2(4000)	Default value of the variable
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–129 ALL_IV_NESTED_EXPERTS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
PARENT_EXPERT_ID	NUMBER(9)	ID of the parent expert
PARENT_EXPERT_NAME	VARCHAR2(255)	Name of the parent expert
EXPERT_ID	NUMBER(9)	ID of the expert

Table 2–129 (Cont.) ALL_IV_NESTED_EXPERTS

Column Name	Data Type	Description
EXPERT_NAME	VARCHAR2(255)	Name of the expert
BUSINESS_NAME	VARCHAR2(1000)	Business name of the expert
DESCRIPTION	VARCHAR2(4000)	Description of the expert
BOUND_OBJECT_ID	NUMBER(9)	ID of the bound object
BOUND_OBJECT_NAME	VARCHAR2(40)	Name of the bound object
IS_VALID	VARCHAR2(13)	Is this nested expert valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Business Intelligence Views

Table 2–130 ALL_IV_ALTERNATIVE_SORT_ORDERS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
ALT_SORT_ORDER_ID	NUMBER(9)	ID of the alternative sort order
ALT_SORT_ORDER_NAME	VARCHAR2(255)	Name of the alternative sort order
BUSINESS_NAME	VARCHAR2(1000)	Business name of the alternative sort order
DESCRIPTION	VARCHAR2(4000)	Description of the alternative sort order
IS_DRILL_TO_DETAIL	CHAR(1)	Whether it acts as a Drill to Detail when deployed to Discoverer
IS_CACHE_VALUES	CHAR(1)	Indicates whether Discoverer should cache the list of values in memory for the current session, once it has been displayed for the first time
IS_REQUIRE_SEARCH	CHAR(1)	Causes Discoverer to request the end user to enter search criteria to reduce the list of values to a subset of the whole list
IS_SHOW_IN_NAVIGATOR	CHAR(1)	Indicates that Discoverer should show the values in the "Select Items" page of the Worksheet Wizard (the item navigator).
IS_SORTED_DISTINCT	CHAR(1)	Indicates that the values should be displayed alphabetically sorted and with duplicates hidden
RETRIEVE_VALUE_GROUP_SIZE	NUMBER(9)	The maximum number of rows to be fetched from the database at a time
VALUES_ITEM_ID	NUMBER(9)	The ID of the item that contains the values to be sorted
VALUES_ITEM_NAME	VARCHAR2(255)	The name of the item that contains the values to be sorted
ORDER_ITEM_ID	NUMBER(9)	The identifier of the item that defines the order in which the values in the VALUES_ITEM_ID field are to be sorted

Table 2–130 (Cont.) ALL_IV_ALTERNATIVE_SORT_ORDERS

Column Name	Data Type	Description
ORDER_ITEM_NAME	VARCHAR2(255)	The name of the item that defines the order in which the values in the VALUES_ITEM_ID field are to be sorted
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–131 ALL_IV_BUSINESS_AREAS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
BUSINESS_AREA_ID	NUMBER(9)	Id of the business area
BUSINESS_AREA_NAME	VARCHAR2(255)	Name of the business area
BUSINESS_NAME	VARCHAR2(1000)	Business name of the business area
DESCRIPTION	VARCHAR2(4000)	Description of the business area
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–132 ALL_IV_BUSINESS_AREA_FOLDERS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
BUSINESS_AREA_ID	NUMBER(9)	ID of the business area
BUSINESS_AREA_NAME	VARCHAR2(255)	Name of the business area
ITEM_FOLDER_ID	NUMBER(9)	Identifier of the Item Folder present in the Business Area
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder present in the business area
BUSINESS_NAME	VARCHAR2(1000)	Business name of the item folder present in the business area
DESCRIPTION	VARCHAR2(4000)	Description of the item folder present in the business area
FOLDER_TYPE	VARCHAR2(40)	The item folder type (simple or complex) of the item folder present in the business area
IS_VISIBLE	NUMBER	The visibility of the item folder to the end-user
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp

Table 2–132 (Cont.) ALL_IV_BUSINESS_AREA_FOLDERS

Column Name	Data Type	Description
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–133 ALL_IV_PRESENTATION_TEMPLATES

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
TEMPLATE_ID	NUMBER(9)	ID of the template
TEMPLATE_NAME	VARCHAR2(255)	Name of the template
BUSINESS_NAME	VARCHAR2(1000)	Business name of the presentation template
DESCRIPTION	VARCHAR2(4000)	Description of the presentation template
PRESENTATION_TYPE	VARCHAR2(40)	The presentation type (CROSSTAB, PIE, BAR_VERT_CLUSTER)
CUBE_ID	NUMBER(9)	ID of the cube that is referenced in the presentation template
CUBE_NAME	VARCHAR2(255)	Name of the cube that is referenced in the presentation template
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–134 ALL_IV_DRILLS_TO_DETAIL

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
DRILL_TO_DETAIL_ID	NUMBER(9)	ID of the drill to detail
DRILL_TO_DETAIL_NAME	VARCHAR2(255)	Name of the drill to detail
BUSINESS_NAME	VARCHAR2(1000)	Business name of the drill to detail
DESCRIPTION	VARCHAR2(4000)	Description of the drill to detail
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–135 ALL_IV_DRILL_LEVELS

Column Name	Data Type	Description
DRILL_PATH_ID	NUMBER(9)	The identifier of the owning drill path
DRILL_PATH_NAME	VARCHAR2(255)	The name of the owning drill path
DRILL_LEVEL_ID	NUMBER(9)	The identifier of the drill level
DRILL_LEVEL_NAME	VARCHAR2(255)	The name of the drill level
BUSINESS_NAME	VARCHAR2(1000)	The business name of the drill level
DESCRIPTION	VARCHAR2(4000)	The description of the drill level
PARENT_DRILL_LEVEL_ID	NUMBER(9)	The identifier of the parent level in the drill path hierarchy
PARENT_DRILL_LEVEL_NAME	VARCHAR2(255)	The name of the parent level in the drill path hierarchy
RELATED_LEVEL_ID	NUMBER(9)	The identifier of the hierarchy level that the drill level was derived from
RELATED_LEVEL_NAME	VARCHAR2(255)	The name of the hierarchy level that the drill level was derived from
ITEM_FOLDER_ID	NUMBER(9)	The identifier of the item folder that the drill level is based on
ITEM_FOLDER_NAME	VARCHAR2(255)	The name of the item folder that the drill level is based on
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–136 ALL_IV_ITEM_FOLDERS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
ITEM_FOLDER_ID	NUMBER(9)	ID of the item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder
BUSINESS_NAME	VARCHAR2(1000)	Business name of the item folder
DESCRIPTION	VARCHAR2(4000)	Description of the item folder
FOLDER_TYPE	VARCHAR2(40)	Type of item folder (simple or complex)
IS_VISIBLE	NUMBER	Whether the item folder is visible to the end user
SOURCE_OBJECT_ID	NUMBER(9)	ID of the source object
SOURCE_OBJECT_TYPE	VARCHAR2(4000)	Type of the source object (Table, Cube, Dimension)
SOURCE_OBJECT_NAME	VARCHAR2(255)	Name of the source object
DIMENSION_ROLE_ID	NUMBER(9)	If this item folder was derived for a dimension role, then this returns the identifier of that dimension role
DIMENSION_ROLE_NAME	VARCHAR2(255)	If this item folder was derived for a dimension role, then this returns the name of that dimension role

Table 2–136 (Cont.) ALL_IV_ITEM_FOLDERS

Column Name	Data Type	Description
RELATED_LEVEL_ID	NUMBER(9)	If this item folder was derived for a level, then this returns the identifier of the level
RELATED_LEVEL_NAME	VARCHAR2(255)	If this item folder was derived for a level then this returns the name of the level
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–137 ALL_IV_ITEM_FOLDER_JOIN_USAGES

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the complex item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the complex item folder
JOIN_ID	NUMBER(9)	ID of the join between two base item folders of the complex item folder
JOIN_NAME	VARCHAR2(255)	Name of the join between two base item folders of the complex item folder

Table 2–138 ALL_IV_ITEMS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder
ITEM_ID	NUMBER(9)	ID of the item
ITEM_NAME	VARCHAR2(255)	Name of the item
BUSINESS_NAME	VARCHAR2(1000)	Business name of the item
DESCRIPTION	VARCHAR2(4000)	Description of the item
IS_COLUMN_ITEM	CHAR(1)	Whether this item is based on a column
DATATYPE	VARCHAR2(40)	The data type of the item
ALIGNMENT	VARCHAR2(40)	Alignment for display (Default, Left, Center, Right)
CASE_STORAGE	VARCHAR2(40)	How alphabetic characters are stored in the database (Unknown, Lower, Upper, Mixed)
CONTENT_TYPE	VARCHAR2(40)	Specifies whether the item can be used to launch an external application
DEFAULT_AGGREGATE	VARCHAR2(255)	Name of default rollup function
DEFAULT_POSITION	VARCHAR2(40)	Default placement for query item (Unknown, Measure, Axis, X-axis, Y-axis, Z-axis)
DEFAULT_WIDTH	NUMBER(9)	Default number of characters in display

Table 2–138 (Cont.) ALL_IV_ITEMS

Column Name	Data Type	Description
DISPLAY_CASE	VARCHAR2(40)	How alphabetic characters should be displayed (Unchanged, Lower, Upper, InitCapped)
FORMAT_MASK	VARCHAR2(255)	The format of the way that the item is displayed
FORMULA	CLOB	The item's expression
HEADING	VARCHAR2(255)	The value of the default heading in a report
IS_VISIBLE	NUMBER	Whether the item is visible to the end user
IS_WORD_WRAP	CHAR(1)	Whether word wrap is allowed when displaying values in a report
MAX_CHAR_FETCHED	NUMBER(9)	Maximum number of characters retrieved from the database
REPLACE_NULL_WITH	VARCHAR2(255)	Value to be displayed for null values
RELATED_ATTRIBUTE_ID	NUMBER(9)	If this item was derived, then the identifier of the attribute it was derived from
RELATED_ATTRIBUTE_TYPE	VARCHAR2(4000)	If this item was derived, then the type of the attribute it was derived from
RELATED_ATTRIBUTE_NAME	VARCHAR2(255)	If this item was derived, then the name of the attribute it was derived from
LIST_OF_VALUES_ID	NUMBER	If this item has a list of values, then the list of values identifier
LIST_OF_VALUES_NAME	VARCHAR2(255)	If this item has a list of values, then the list of values name
ALTERNATIVE_SORT_ORDER_ID	NUMBER	If this item has an alternative sort order then the ID of the alternative sort order
ALTERNATIVE_SORT_ORDER_NAME	VARCHAR2(255)	If this item has an alternative sort order then the name of the alternative sort order
DRILL_TO_DETAIL_ID	NUMBER	If this item has a drill to detail then the ID of the drill to detail
DRILL_TO_DETAIL_NAME	VARCHAR2(255)	If this item has a drill to detail then the name of the drill to detail
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–139 ALL_IV_ITEM_FORMULA_REFS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder
ITEM_ID	NUMBER(9)	ID of the item
ITEM_NAME	VARCHAR2(255)	Name of the item

Table 2–139 (Cont.) ALL_IV_ITEM_FORMULA_REFS

Column Name	Data Type	Description
TAG	NUMBER(9)	Refers to the tag value used in the formula of the item
REFERENCED_FUNCTION_ID	NUMBER(9)	ID of the referenced function
REFERENCED_FUNCTION_TYPE	VARCHAR2(4000)	Type of the referenced function
REFERENCED_FUNCTION_NAME	VARCHAR2(255)	Name of the referenced function
REFERENCED_ITEM_FOLDER_ID	NUMBER(9)	ID of the referenced item folder
REFERENCED_ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the referenced item folder
REFERENCED_ITEM_ID	NUMBER(9)	ID of the referenced item
REFERENCED_ITEM_NAME	VARCHAR2(255)	Name of the referenced item

Table 2–140 ALL_IV_DATA_ITEMS

Column Name	Data Type	Description
TEMPLATE_ID	NUMBER(9)	ID of the presentation template
TEMPLATE_NAME	VARCHAR2(255)	Name of the presentation template
DATA_ITEM_ID	NUMBER(9)	ID of the data item
DATA_ITEM_NAME	VARCHAR2(255)	Name of the data item
BUSINESS_NAME	VARCHAR2(1000)	Business name of the data item
DESCRIPTION	VARCHAR2(4000)	Description of the data item
MEASURE_ID	NUMBER(9)	The identifier of the measure used as the data item
MEASURE_NAME	VARCHAR2(255)	The name of the measure used as the data item
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–141 ALL_IV_EDGE_ITEMS

Column Name	Data Type	Description
TEMPLATE_ID	NUMBER(9)	ID of the presentation template
TEMPLATE_NAME	VARCHAR2(255)	Name of the presentation template
EDGE_ITEM_ID	NUMBER(9)	Identifier of the edge item
EDGE_ITEM_NAME	VARCHAR2(255)	Name of the edge item
BUSINESS_NAME	VARCHAR2(1000)	Business name of the edge item
DESCRIPTION	VARCHAR2(4000)	Description of the edge item
PLACEMENT	VARCHAR2(40)	The axis the edge item is on
DIMENSION_ROLE_ID	NUMBER(9)	ID of the dimension role
DIMENSION_ROLE_NAME	VARCHAR2(255)	Name of the dimension role
DIMENSION_ID	NUMBER(9)	ID of the dimension

Table 2–141 (Cont.) ALL_IV_EDGE_ITEMS

Column Name	Data Type	Description
DIMENSION_NAME	VARCHAR2(255)	Name of the dimension
HIERARCHY_ID	NUMBER(9)	Not used
HIERARCHY_NAME	VARCHAR2(255)	Not used
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–142 ALL_IV_DRILL_PATHS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
DRILL_PATH_ID	NUMBER(9)	ID of the drill path
DRILL_PATH_NAME	VARCHAR2(255)	Name of the drill path
BUSINESS_NAME	VARCHAR2(1000)	Business name of the drill path
DESCRIPTION	VARCHAR2(4000)	Description of the drill path
DIMENSION_ID	NUMBER(9)	If the drill path was derived from a dimension hierarchy, then the ID of the dimension
DIMENSION_NAME	VARCHAR2(255)	If the drill path was derived from a dimension hierarchy then the name of the dimension
DIMENSION_ROLE_ID	NUMBER(9)	If the drill path was derived from a dimension role's hierarchy then its identifier
DIMENSION_ROLE_NAME	VARCHAR2(255)	If the drill path was derived from a dimension role's hierarchy then its name
HIERARCHY_ID	NUMBER(9)	The identifier of the hierarchy the drill path was derived from
HIERARCHY_NAME	VARCHAR2(255)	The name of the hierarchy the drill path was derived from
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–143 ALL_IV_DRILL_LEVEL_ITEMS

Column Name	Data Type	Description
DRILL_PATH_ID	NUMBER(9)	Owning drill path identifier
DRILL_PATH_NAME	VARCHAR2(255)	Owning drill path name
DRILL_LEVEL_ID	NUMBER(9)	ID of the drill level
DRILL_LEVEL_NAME	VARCHAR2(255)	Name of the drill level

Table 2–143 (Cont.) ALL_IV_DRILL_LEVEL_ITEMS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	Item folder ID
ITEM_FOLDER_NAME	VARCHAR2(255)	Item folder name
ITEM_ID	NUMBER(9)	ID of the item
ITEM_NAME	VARCHAR2(255)	Name of the item

Table 2–144 ALL_IV_DRILL_PATH_JOIN_USAGES

Column Name	Data Type	Description
DRILL_PATH_ID	NUMBER(9)	ID of the drill path
DRILL_PATH_NAME	VARCHAR2(255)	Name of the drill path
PARENT_DRILL_LEVEL_ID	NUMBER(9)	ID of the parent drill level
PARENT_DRILL_LEVEL_NAME	VARCHAR2(255)	Name of the parent drill level
CHILD_DRILL_LEVEL_ID	NUMBER(9)	ID of the child drill level
CHILD_DRILL_LEVEL_NAME	VARCHAR2(255)	Name of the child drill level
JOIN_ID	NUMBER(9)	ID of the join
JOIN_NAME	VARCHAR2(255)	Name of the join

Table 2–145 ALL_IV_LISTS_OF_VALUES

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
LIST_OF_VALUES_ID	NUMBER(9)	ID of the list of values
LIST_OF_VALUES_NAME	VARCHAR2(255)	Name of the list of values
BUSINESS_NAME	VARCHAR2(1000)	Business name of the list of values
DESCRIPTION	VARCHAR2(4000)	Description of the list of values
IS_DRILL_TO_DETAIL	CHAR(1)	Whether it acts as a drill to detail when deployed to Discoverer
IS_CACHE_VALUES	CHAR(1)	Indicates whether Discoverer should cache the list of values in memory for the current session, once it has been displayed for the first time
IS_REQUIRE_SEARCH	CHAR(1)	Value 1 causes Discoverer to request the end user to enter search criteria to reduce the list of values to a subset of the whole list
IS_SHOW_IN_NAVIGATOR	CHAR(1)	Indicates whether Discoverer should show the values in the "Select Items" page of the Worksheet Wizard (the item navigator)
IS_SORTED_DISTINCT	CHAR(1)	Value 1 indicates that the values should be displayed alphabetically sorted and with duplicates hidden
RETRIEVE_VALUE_GROUP_SIZE	NUMBER(9)	The maximum number of rows to be fetched from the database at a time
VALUES_ITEM_ID	NUMBER(9)	The identifier of the item that supplies the values

Table 2–145 (Cont.) ALL_IV_LISTS_OF_VALUES

Column Name	Data Type	Description
VALUES_ITEM_NAME	VARCHAR2(255)	The name of the item that supplies the values
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–146 ALL_IV_REGISTERED_FUNCTIONS

Column Name	Data Type	Description
MODULE_ID	NUMBER(9)	ID of the module
MODULE_NAME	VARCHAR2(255)	Name of the module
REGISTERED_FUNCTION_ID	NUMBER(9)	ID of the registered function
REGISTERED_FUNCTION_NAME	VARCHAR2(255)	Name of the registered function
BUSINESS_NAME	VARCHAR2(1000)	Business name of the registered function
DESCRIPTION	VARCHAR2(4000)	Description of the registered function
SIGNATURE	VARCHAR2(4000)	The signature of the registered function
IS_AVAILABLE	CHAR(1)	Whether a Discoverer end user may use this function in calculations
SOURCE_FUNCTION_ID	NUMBER(9)	The identifier of the function it was derived from
SOURCE_FUNCTION_NAME	VARCHAR2(255)	The name of the function it was derived from
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–147 ALL_IV_CONDITION_FORMULA_REFS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder
ITEM_ID	NUMBER(9)	ID of the condition
ITEM_NAME	VARCHAR2(255)	Name of the condition
TAG	NUMBER(9)	Refers to the tag value used in the condition's formula
REFERENCED_FUNCTION_ID	NUMBER(9)	ID of the referenced function
REFERENCED_FUNCTION_TYPE	VARCHAR2(4000)	Type of the referenced function
REFERENCED_ITEM_ID	NUMBER(9)	ID of the referenced item

Table 2–147 (Cont.) ALL_IV_CONDITION_FORMULA_REFS

Column Name	Data Type	Description
REFERENCED_ITEM_NAME	VARCHAR2(255)	Name of the referenced item
REFERENCED_CONDITION_ID	NUMBER(9)	ID of the referenced condition
REFERENCED_CONDITION_NAME	VARCHAR2(255)	Name of the referenced condition

Table 2–148 ALL_IV_JOIN_COMPONENTS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder
JOIN_ID	NUMBER(9)	ID of the join
JOIN_NAME	VARCHAR2(255)	Name of the join
JOIN_COMPONENT_ID	NUMBER(9)	Identifier of the join component
JOIN_COMPONENT_NAME	VARCHAR2(255)	Name of the join component
BUSINESS_NAME	VARCHAR2(1000)	Business Name of the join component
DESCRIPTION	VARCHAR2(4000)	Description of the join component
JOIN_OPERATOR	VARCHAR2(40)	The operator for the join component
DETAIL_ITEM_ID	NUMBER(9)	The identifier of the item referenced in the detail item folder
DETAIL_ITEM_NAME	VARCHAR2(255)	The name of the item referenced in the detail item folder
MASTER_ITEM_ID	NUMBER(9)	The identifier of the item referenced in the master item folder
MASTER_ITEM_NAME	VARCHAR2(255)	The name of the item referenced in the master item folder
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–149 ALL_IV_JOINS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the detail item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the detail item folder
JOIN_ID	NUMBER(9)	ID of the join
JOIN_NAME	VARCHAR2(255)	Name of the join
BUSINESS_NAME	VARCHAR2(1000)	Business name of the join
DESCRIPTION	VARCHAR2(4000)	Description of the join
IS_OUTER_JOIN_ON_MASTER	CHAR(1)	Indicates whether to perform an outer join on the master item folder. If this is set, any detail rows that have no related master row will be included in the results of the join

Table 2–149 (Cont.) ALL_IV_JOINS

Column Name	Data Type	Description
IS_OUTER_JOIN_ON_DETAIL	CHAR(1)	Indicates whether to perform an outer join on the detail item folder. If this is set, any master rows that have no related detail rows will be included in the results of the join
IS_DETAIL_EXISTS_ON_MASTER	CHAR(1)	Indicates whether every detail row must refer to a valid master row
IS_ONE_TO_ONE	CHAR(1)	Indicates whether each master row only ever has a single detail row
REFERENCED_ITEM_FOLDER_ID	NUMBER(9)	The identifier of the master item folder referenced from the join
REFERENCED_ITEM_FOLDER_NAME	VARCHAR2(255)	The name of the master item folder referenced from the join
RELATED_FOREIGN_KEY_ID	NUMBER(9)	The identifier of the foreign key that this join was derived from
RELATED_FOREIGN_KEY_NAME	VARCHAR2(255)	The name of the foreign key that this join was derived from
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–150 ALL_IV_CONDITIONS

Column Name	Data Type	Description
ITEM_FOLDER_ID	NUMBER(9)	ID of the item folder
ITEM_FOLDER_NAME	VARCHAR2(255)	Name of the item folder
CONDITION_ID	NUMBER(9)	ID of the condition
CONDITION_NAME	VARCHAR2(255)	Name of the condition
BUSINESS_NAME	VARCHAR2(1000)	Business name of the condition
DESCRIPTION	VARCHAR2(4000)	Description of the condition
IS_MANDATORY	CHAR(1)	Whether the condition is mandatory
FORMULA	CLOB	Formula of the condition
IS_MATCH_CASE	CHAR(1)	Whether the alphabetic character case must match exactly
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Real Time Views

Table 2–151 ALL_IV_STREAMS_QUEUES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
QUEUE_ID	NUMBER(9)	ID of the queue
QUEUE_NAME	VARCHAR2(255)	Name of the queue
BUSINESS_NAME	VARCHAR2(1000)	Business name of the queue
DESCRIPTION	VARCHAR2(4000)	Description of the queue
QUEUE_TABLE	VARCHAR2(255)	Physical name of the queue table
QUEUE_TABLE_ID	NUMBER(9)	ID of the queue table
PAYLOAD_TYPE	CHAR(11)	Type of payload
IS_VALID	VARCHAR2(13)	Is the queue valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–152 ALL_IV_QUEUES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
QUEUE_ID	NUMBER(9)	ID of the queue
QUEUE_NAME	VARCHAR2(255)	Name of the queue
BUSINESS_NAME	VARCHAR2(1000)	Business name of the queue
DESCRIPTION	VARCHAR2(4000)	Description of the queue
QUEUE_TABLE	VARCHAR2(255)	Physical name of the queue table
QUEUE_TABLE_ID	NUMBER(9)	ID of the queue table
PAYLOAD_TYPE	VARCHAR2(255)	Type of payload for the queue
PAYLOAD_TYPE_ID	NUMBER	ID of the payload type
IS_VALID	VARCHAR2(13)	Is the queue valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–153 ALL_IV_QUEUE_PROPAGATIONS

Column Name	Data Type	Description
QUEUEPROPAGATION_ID	NUMBER(9)	ID of the queue propagation
QUEUETABLE_NAME	VARCHAR2(255)	Physical name of the queue table
BUSINESS_NAME	VARCHAR2(1000)	Business name of the queue table
DESCRIPTION	VARCHAR2(4000)	Description of the queue table
SOURCE_QUEUE	VARCHAR2(255)	Name of the source queue
SOURCE_QUEUE_ID	NUMBER(9)	ID of the source queue
TARGET_QUEUE	VARCHAR2(255)	Name of the target queue
TARGET_QUEUE_ID	NUMBER(9)	ID of the target queue
IS_VALID	VARCHAR2(13)	Is the queue table valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–154 ALL_IV_QUEUE_TABLES

Column Name	Data Type	Description
SCHEMA_ID	NUMBER(9)	ID of the schema
SCHEMA_NAME	VARCHAR2(255)	Name of the schema
QUEUETABLE_ID	NUMBER(9)	ID of the queue table
QUEUETABLE_NAME	VARCHAR2(255)	Name of the queue table
BUSINESS_NAME	VARCHAR2(1000)	Business name of the queue table
DESCRIPTION	VARCHAR2(4000)	Description of the queue table
PAYLOAD_TYPE	VARCHAR2(767)	Type of payload
PAYLOAD_TYPE_ID	NUMBER(9)	ID of payload type
IS_VALID	VARCHAR2(13)	Is queue table valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–155 ALL_IV_STREAMS_CAPTURE

Column Name	Data Type	Description
STREAMSCAPTURE_ID	NUMBER(9)	ID of the streams capture
STREAMSCAPTURE_NAME	VARCHAR2(255)	Name of the streams capture
BUSINESS_NAME	VARCHAR2(1000)	Business name of the streams capture
DESCRIPTION	VARCHAR2(4000)	Description of the streams capture
STREAMS_QUEUE	VARCHAR2(255)	Streams queue

Table 2–155 (Cont.) ALL_IV_STREAMS_CAPTURE

Column Name	Data Type	Description
STREAMS_QUEUE_ID	NUMBER(9)	ID of the streams queue
IS_VALID	VARCHAR2(13)	Validation status
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–156 ALL_IV_CAPTURE_RELATIONS

Column Name	Data Type	Description
STREAMS_CAPTURE_ID	NUMBER(9)	ID of the streams capture
STREAMS_CAPTURE_NAME	VARCHAR2(255)	Name of the streams capture
TABLE_ID	NUMBER(9)	ID of the table
TABLE_NAME	VARCHAR2(255)	Name of the table
CAPTURERELATION_ID	NUMBER(9)	ID of the capture relation
CAPTUREREALTION_NAME	VARCHAR2(255)	Name of the capture relation
BUSINESS_NAME	VARCHAR2(1000)	Business name of the capture relation
DESCRIPTION	VARCHAR2(4000)	Description of the capture relation
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Scheduling Views

Table 2–157 ALL_IV_SCHEDULABLE

Column Name	Data Type	Description
OBJECT_ID	NUMBER(9)	ID of the object
OBJECT_TYPE	VARCHAR2(4000)	Type of the object
OBJECT_NAME	VARCHAR2(255)	Name of the object
CONFIGURATION_ID	NUMBER(9)	ID of the configuration
CONFIGURATION_NAME	VARCHAR2(255)	Name of the configuration
SCHEDULE_ID	NUMBER(9)	ID of the applied schedule
SCHEDULE_NAME	VARCHAR2(255)	Name of the applied schedule

Table 2–158 ALL_IV_CALENDAR_SCHEDULES

Column Name	Data Type	Description
CALENDAR_ID	NUMBER(9)	ID of the calendar
CALENDAR_NAME	VARCHAR2(255)	Name of the calendar
SCHEDULE_ID	NUMBER(9)	ID of the schedule
SCHEDULE_NAME	VARCHAR2(255)	Name of the schedule
BUSINESS_NAME	VARCHAR2(1000)	Business name of the schedule
DESCRIPTION	VARCHAR2(4000)	Description of the schedule
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user
STARTTIME	DATE	Start time of the schedule
ENDTIME	DATE	End time of the schedule
REPEATEXPRESSSION	VARCHAR2(4000)	Expression defining how often the schedule is active
TIMEZONE	VARCHAR2(255)	Time zone that the start and end time refer to

Others

Table 2–159 ALL_IV_ACTIVITY_FOLDERS

Column Name	Data Type	Description
PROJECT_ID	NUMBER(9)	ID of the project
PROJECT_NAME	VARCHAR2(255)	Name of the project
ACTIVITY_FOLDER_ID	NUMBER(9)	ID of the activity folder
ACTIVITY_FOLDER_NAME	VARCHAR2(255)	Name of the activity folder
BUSINESS_NAME	VARCHAR2(1000)	Business name of the activity folder
DESCRIPTION	VARCHAR2(4000)	Description of the activity folder
IS_VALID	VARCHAR2(13)	Is the activity folder valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–160 ALL_IV_ACTIVITY_TEMPLATES

Column Name	Data Type	Description
ACTIVITY_FOLDER_ID	NUMBER(9)	ID of the activity folder
ACTIVITY_FOLDER_NAME	VARCHAR2(255)	Name of the activity folder
ACTIVITY_TEMPLATE_ID	NUMBER(9)	ID of the activity template
ACTIVITY_TEMPLATE_NAME	VARCHAR2(255)	Name of the activity template

Table 2–160 (Cont.) ALL_IV_ACTIVITY_TEMPLATES

Column Name	Data Type	Description
BUSINESS_NAME	VARCHAR2(1000)	Business name of the activity template
DESCRIPTION	VARCHAR2(4000)	Description of the activity template
IS_VALID	VARCHAR2(13)	Is the activity template valid
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–161 ALL_IV_PLS_COLLECTIONS

Column Name	Data Type	Description
LIBRARY_ID	NUMBER(9)	ID of the library
LIBRARY_NAME	VARCHAR2(255)	Name of the library
COLLECTION_ID	NUMBER(9)	ID of the collection
COLLECTION_NAME	VARCHAR2(255)	Name of the collection
BUSINESS_NAME	VARCHAR2(1000)	Business name of the collection
DESCRIPTION	VARCHAR2(4000)	Description of the collection
COLLECTION_TYPE	VARCHAR2(255)	Type of the collection
RELATED_RECORD_ID	NUMBER(9)	ID of the related PLS record
RELATED_RECORD_NAME	VARCHAR2(255)	Name of the related PLS record
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–162 ALL_IV_PLS_RECORDS

Column Name	Data Type	Description
LIBRARY_ID	NUMBER(9)	ID of the library
LIBRARY_NAME	VARCHAR2(255)	Name of the library
RECORD_ID	NUMBER(9)	ID of the record
RECORD_NAME	VARCHAR2(255)	Name of the record
BUSINESS_NAME	VARCHAR2(1000)	Business name of the record
DESCRIPTION	VARCHAR2(4000)	Description of the record
RECORD_TYPE	VARCHAR2(40)	Type of the record
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Table 2–163 ALL_IV_ROW_RELATIONSHIPS

Column Name	Data Type	Description
PROFILE_ID	NUMBER(9)	ID of the profile
PROFILE_NAME	VARCHAR2(255)	Name of the profile
ENTITY_ID	NUMBER(9)	ID of the entity
ENTITY_NAME	VARCHAR2(255)	Name of the entity
ROW_RELATIONSHIP_ID	NUMBER(9)	ID of the row relationship
ROW_RELATIONSHIP_NAME	VARCHAR2(255)	Name of the row relationship
BUSINESS_NAME	VARCHAR2(1000)	Business name of the row relationship
REMOTE_KEY_ID	NUMBER(9)	ID of the other row relationship
IS_DISCOVERED	CHAR(3)	If this row relationship was discovered
IS_DOCUMENTED	CHAR(2)	If this row relationship is documented
LOCAL_MAX_CARDINALITY	VARCHAR2(40)	Maximum number of values found on the local side
LOCAL_MIN_CARDINALITY	VARCHAR2(40)	Minimum number of values found on the local side
REMOTE_MAX_CARDINALITY	VARCHAR2(40)	Maximum number of values found on the remote side
REMOTE_MIN_CARDINALITY	VARCHAR2(40)	Minimum number of values found on the remote side
NUM_ORPHANS	VARCHAR2(40)	Number of distinct values found in the local column but not in the remote column
COMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that satisfy the discovered row relationship
COMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
NONCOMPLIANT_QUERY	VARCHAR2(4000)	The query used to get the rows that do not satisfy the discovered common format
NONCOMPLIANT_CNT_QUERY	VARCHAR2(4000)	Not used
DRILLDOWN_QUERY	VARCHAR2(4000)	The query used to get all the rows of the table but distinguishes which row satisfies the discovered row relationship and which does not
UPDATED_ON	DATE	Update timestamp
CREATED_ON	DATE	Creation timestamp
UPDATED_BY	VARCHAR2(40)	Updated by user
CREATED_BY	VARCHAR2(40)	Created by user

Using SQL*Plus to Schedule and Execute Jobs

After you design and configure the logical definitions of your target system, you can deploy and create the physical instance of your target. You can then start deployed mapping and process flow scripts to load or update your data.

This chapter contains the following topics:

- [Managing Jobs Using SQL Scripts](#)
- [Starting ETL Jobs in SQL*Plus](#)
- [Scheduling ETL Jobs in Oracle Enterprise Manager](#)
- [Managing a Control Center](#)

Managing Jobs Using SQL Scripts

Numerous SQL scripts are installed with Warehouse Builder so that you can manage jobs using other administrative tools.

[Table 3–1](#) describes the scripts that you can use to manage deployment jobs, execution jobs, and Control Centers from SQL*Plus. The scripts are located in `\owb\rtpl\sql` in the Oracle home directory for Warehouse Builder. Comments in the scripts explain how to use them.

Table 3–1 SQL Scripts for Managing Jobs and Control Centers

Script	Description
<code>abort_exec_request</code>	Stops an execution that is currently busy.
<code>abort_unit_request</code>	Stops a deployment job at the unit level. A deployment unit is a collection of objects that are being deployed to the same location with the same deployment action.
<code>deactivate_deployment</code>	Deactivates a deployment job.
<code>deactivate_execution</code>	Deactivates an execution job.
<code>delete_warehouse_object</code>	Deletes an object from a Warehouse Builder repository.
<code>display_platform_property</code>	Displays the value of a platform property. These properties control how the Control Center service behaves.
<code>expedite_exec_request</code>	Moves a deployment job to the top of the list of pending jobs.
<code>list_requests</code>	Lists the details of any active deployment or execution requests.

Table 3-1 (Cont.) SQL Scripts for Managing Jobs and Control Centers

Script	Description
oem_exec_background_template	Starts a specified job in the background. This template must be copied into a user-defined SQL*Plus job in Enterprise Manager. You can obtain the task status and return result from the public views. The views have names that begin ALL_RT_*.
oem_exec_template	Creates a new, parameterized job or submits a new job for immediate execution. This template must be copied into a user-defined SQL*Plus job in Enterprise Manager.
print_exec_details	Prints the audit execution hierarchy and details about the individual executions associated with an audit ID.
print_running_exec_details	Prints the audit execution hierarchy and details of all executions that are incomplete and were started since a provided date.
purge_audit_template	Purges deployment audit data or execution audit data.
reset_repository	Resets the registration details for a Control Center.
rtrepos_report	Displays the details of the runtime repository and its registered locations.
service_doctor	Displays diagnostics about the Control Center repository and its service.
set_oem_home	Sets the platform properties associated with an Enterprise Manager home directory. These properties enable the Control Center to locate Enterprise Manager components.
set_platform_property	Sets the value of a platform property. These properties control how the Control Center service behaves.
set_repository_password	Sets the repository password, which is used by the Control Center service at startup.
show_service	Displays the current status of the Control Center service.
sqlplus_exec_background_template	Starts the specified object in the background. You can obtain the task status and return result from the public views. The views have names that begin ALL_RT_*
sqlplus_exec_template	Starts a job as described in " Starting ETL Jobs in SQL*Plus " on page 3-2.
start_service	Starts a Control Center Service.
stop_service	Stops a Control Center Service.

Starting ETL Jobs in SQL*Plus

In addition to executing objects using the Control Center Manager, you can use SQL*Plus. To do this, use a script provided with Warehouse Builder named `sqlplus_exec_template`. Alternatively, you can use `sqlplus_exec_background_template` to run a job in the background.

Take these steps to run the `SQLPLUS_EXEC_TEMPLATE` script in SQL*Plus:

1. From the Warehouse Builder Tools menu, choose **SQL*Plus**.
The SQL*Plus window opens.
2. Connect as a Warehouse Builder user, not as a repository owner.
3. Start the script, using syntax such as the following.

```
@%ORACLE_HOME%\owb\rtpl\sql\sqlplus_exec_template MY_RUNTIME MY_WAREHOUSE PLSQL
MY_MAPPING " " " "
```

Refer to "[The SQLPLUS_EXEC_TEMPLATE SQL Script](#)" for a complete description of the syntax.

Scheduling ETL Jobs in Oracle Enterprise Manager

After you successfully deployed a mapping or a process flow, you can schedule it to run in Oracle Enterprise Manager. This is an alternative to using the Warehouse Builder scheduler described in the *User's Guide*.

See Also: *Oracle Enterprise Manager Concepts* and the Enterprise Manager Help system for information about creating jobs and schedules.

To schedule a mapping or process flow in Enterprise Manager:

1. Successfully deploy the mapping or process flow in Warehouse Builder.
2. Connect to Enterprise Manager as a Warehouse Builder repository user or owner.
3. Create a scheduler job that uses the `WB_RT_API_EXEC.RUN_TASK` function in a PL/SQL block.

For more information about this function, refer to "[The WB_RT_API_EXEC.RUN_TASK Function](#)".

4. Create a schedule for running the job.

The SQLPLUS_EXEC_TEMPLATE SQL Script

This script enables you to start the ETL process from SQL*Plus, and to use scheduling tools such as cron, AT, Autosys, and Tivoli.

The `sqlplus_exec_template.sql` script is located in the following directory:
`ORACLE_HOME/owb/rtp/sql`.

Return Value

- 1 = Success
- 2 = Warning
- 3 = Error

Syntax

```
SQLPLUS_EXEC_TEMPLATE rt_owner location task_type task_name
                        system_params custom_params
```

Arguments

Provide a value for each of the following arguments.

- **rt_owner:** The repository owner
- **location:** For PL/SQL mappings and process flows, specify the location you used for deployment.

For SQL*Loader and SAP mappings, set this parameter to `PlatformSchema`. This is a case-sensitive variable.

- **task_type:** Enter the appropriate task type for the mapping or the process flow:

- PLSQLMAP: PL/SQL mapping
- SQLLOADERCONTROLFILE: SQL*Loader mapping
- PROCESSFLOW: Process flow
- ABAPFILE: SAP mapping
- DATAAUDITOR: Data Auditor mapping
- SCHEDULEDJOB: Warehouse Builder scheduled job
- **task_name**: The physical name of the mapping or the process flow.
- **system_params**: Values of system parameters for this task type. These values override the default values. Enter the parameters in the form *name=value*. Separate multiple parameters with commas, and enclose the entire string in double quotes. A backslash (\) is the escape character, when you need to include commas or double quotes as literal text.

The following examples are correct:

```
" , "
"this_param=true"
"this_param=true, that_param=2"
```

- **custom_params**: Values of a custom parameter defined for this task. Refer to *system_params* for the syntax.

Examples

In each of the following examples, you may need to provide the path to `sqlplus.exe` and to `sqlplus_exec_template.sql`.

```
sqlplus user/password@tns_name @sqlplus_exec_template MY_RUNTIME MY_WAREHOUSE
PLSQL MY_MAPPING " , " , "
```

```
sqlplus user/password@tns_name @sqlplus_exec_template MY_RUNTIME PlatformSchema
SQL_LOADER MY_LOAD " , " , "
```

```
sqlplus user/password@tns_name @sqlplus_exec_template MY_RUNTIME MY_WORKFLOW
PROCESS MY_PROCESS " , " , "
```

```
sqlplus user/password@tns_name @sqlplus_exec_template MY_RUNTIME PlatformSchema
ABAP MY_SAP " , " , "
```

The WB_RT_API_EXEC.RUN_TASK Function

The `RUN_TASK` function of the `WB_RT_API_EXEC` PL/SQL package enables you to schedule and run the ETL process from Warehouse Builder.

Return Value

The return value is affected by the parameters of the function.

When `background=0` and `oem_friendly=0`:

```
1 = Success
2 = Warning
3 = Error
```

When `background=0` and `oem_friendly=1`:

```
0 = Success or Warning
```

3 = Error

When *background=1*:

0 = Task successfully submitted for execution

1 = Task not successfully submitted

Syntax

```

RUN_TASK
  ( location          IN   VARCHAR2,
    task_type         IN   VARCHAR2,
    task_name         IN   VARCHAR2,
    custom_params     IN   VARCHAR2   DEFAULT NULL,
    system_params     IN   VARCHAR2   DEFAULT NULL,
    oem_friendly      IN   NUMBER     DEFAULT 0,
    background        IN   NUMBER     DEFAULT 0
  )
RETURN NUMBER;

```

Provide a value for each of the following parameters:

- **location:** For PL/SQL mappings and process flows, specify the location you used for deployment.

For SQL*Loader and SAP mappings, set this parameter to `PlatformSchema`. This is a case-sensitive variable.

- **task_type:** Enter the appropriate task type for the mapping or the process flow:
 - PLSQLMAP: PL/SQL mapping
 - SQLLOADER: SQL*Loader mapping
 - PROCESSFLOW: Process flow
 - SAP: SAP mapping
 - DATAAUDITOR: Data auditor mapping
 - SCHEDULEDJOB: Warehouse Builder schedule object
- **task_name:** The name of the mapping or the process flow.
- **custom_params:** Values of a custom parameter defined for this task. Refer to `system_params` for the syntax.
- **system_params:** Values of system parameters for this task type. These values override the default values. Enter the parameters in the form *name=value*. Separate multiple parameters with commas, and enclose the entire string in double quotes. A backslash (\) is the escape character, when you need to include commas or double quotes as literal text.

The following examples are correct:

```

" "
"this_param=true"
"this_param=true, that_param=2"

```

- **oem_friendly:** Controls the return values. Set to 1 for execution in Enterprise Manager, or set to 0 for other environments.
- **background:** Controls execution of the task. Set to 1 for background, or set to 0 for foreground.

Example

The following example displays the return value of the function, which runs a mapping named `CUSTOMER_MAP` in `SALES_TARGET_LOCATION`.

```
BEGIN
DBMS_OUTPUT.PUT_LINE('Result: ' || TO_CHAR(gccrep.wb_rt_api_exec.run_task(
'SALES_TARGET_LOCATION','PLSQLMAP','CUSTOMER_MAP', null, null, 1)));
END;
```

Managing a Control Center

The repository owner may occasionally need to run a SQL script to manager a control center.

Disabling Automatic Recovery

Warehouse Builder automatically restarts jobs that were interrupted as the result of Oracle Database shutting down. You can control this behavior for all jobs associated with a particular Control Center by setting the `RECOVERY` platform property:

- `TRUE`: Automatically restarts jobs. (Default)
- `FALSE`: Jobs require manual restart.

To turn off automatic recovery:

1. Open SQL*Plus and connect as the repository owner.
2. Use the `SET_PLATFORM_PROPERTY` script to set the `RECOVERY` property to `FALSE`.

The following example sets the property on a Windows platform:

```
@%ORACLE_HOME%\owb\rtpl\sql\set_platform_property RECOVERY FALSE
```

`ORACLE_HOME` is the Oracle home directory for Warehouse Builder.

Unregistering Locations

If a location cannot be unregistered using the Control Center Manager or an OMB*Plus script, you can use a SQL script. This problem may occur if the control center becomes inaccessible for any reason, such as disk failure or simply deleting the control center without first unregistering the locations.

Note: Only use this method of unregistering locations when the usual methods are unsuccessful.

To unregister a location:

1. Open SQL*Plus and connect as the repository owner.
2. Run the `unregister_location` script. The following example shows the syntax on Windows:

```
@%ORACLE_HOME%\owb\misc\unregister_location
```

`ORACLE_HOME` is the Oracle home directory for Warehouse Builder.

Part II

OMB Plus Scripting Language

This part contains the following chapters:

- Chapter 4, "Introduction to OMB Plus"
- Chapter 5, "OMB Commands"
- Chapter 6, "OMBALTER to OMBALTER EXTERNAL_TABLE"
- Chapter 7, "OMBALTER FLAT_FILE to OMBALTER STREAMS_QUEUE"
- Chapter 8, "OMBALTER TABLE to OMBALTER VIEW"
- Chapter 9, "OMBCREATE to OMBCREATE PLSQL_TABLE_TYPE"
- Chapter 10, "OMBCREATE PLUGGABLE_MAPPING to OMBCREATE VIEW"
- Chapter 11, "OMBRETRIEVE to OMBRETRIEVE LOCATION"
- Chapter 12, "OMBRETRIEVE MAPPING to OMBRETRIEVE VIEW"
- Chapter 13, "OMBDROP"
- Chapter 14, "OMU Commands"
- Appendix A, "Additional and Optional Usages"

Introduction to OMB Plus

OMB Plus is a flexible, high-level command line metadata access tool for Oracle Warehouse Builder. Use OMB Plus to create, modify, delete, and retrieve object metadata in Warehouse Builder design and runtime repositories.

This chapter contains the following topics:

- [About the OMB Scripting Language](#) on page 4-1
- [OMB Plus Commands](#) on page 4-5
- [How to Read Syntax Diagrams](#) on page 4-10
- [Sample OMB Plus Scripts](#) on page 4-11
- [New to OMB Plus in This Release](#) on page 4-14

About the OMB Scripting Language

The Warehouse Builder scripting language, known as OMB Plus, is an extension of the Tcl programming language. With OMB Plus, you can write the syntactic constructs such as variable support, conditional and looping control structures, error handling, and standard library procedures.

Use OMB Plus to create, modify, delete, and retrieve object metadata in Warehouse Builder design and runtime repositories.

OMB Plus enables you to edit Warehouse Builder repository metadata using a scripting interface. You can use this scripting interface to:

- Perform complex actions directly in Warehouse Builder, without launching the client user interface.
- Define sets of routine operations to be executed in Warehouse Builder.
- Perform batch operations in Warehouse Builder.
- Automate a series of conditional operations in Warehouse Builder.

Using OMB Plus

To use OMB Plus, first launch OMB Plus and then connect to a repository. Type all commands and keywords in OMB Plus in uppercase.

Launching OMB Plus

To launch the OMB Plus console, follow the instructions specific to your operating system.

- **UNIX:** At the command prompt, enter:
`<OWB_HOME>/bin/unix/OMBPlus.sh`
- **Windows:** From the Start Menu, navigate to Warehouse Builder, located within Oracle-OUI. Select OWB OMB Plus from the list of menu items.

Connecting to a Repository

From the OMB Plus console, enter:

```
OMBCONNECT <repos>/<password>@<host>:<port>:<service_name>
```

In this expression, `<repos>` is the name of the design-time repository, `<host>` is the machine on which the repository is installed, and `<servicename>` is the name of the database that contains the repository.

Getting Help for OMB Plus Commands

Use the `OMBHELP` command to display help on Warehouse Builder commands. The help describes the purpose of the command, the syntax in BNF format, and each of the keywords or options.

The syntax for `OMBHELP` is:

```
help ::= OMBHELP <command_name> [ <command_specific_options> ] [DETAIL]
```

For details and an example on how to use `OMBHELP`, see [OMBHELP](#) on page 4-7.

Writing OMB Plus Commands

Keep in mind the following points when you execute OMB Plus commands:

Specifying Values

You can set the value of a Boolean configuration property, for example the `IS_DELIMITED` property in the example, using any one of the following values: `TRUE`, `FALSE`, `YES`, `NO`, `1`, or `0`. When you set the value to `TRUE`, `FALSE`, `YES`, or `NO`, enclose the value in single quotes.

When you specify numeric values for a configuration property, do not enclose the values in single quotes.

Special Characters

Do not use a number sign (`#`) in the middle of an OMB Plus command.

Tcl interprets several special characters differently than Warehouse Builder and the Oracle database. Review this and subsequent sections to learn how to properly use the following characters in OMB Plus: dollar sign (`$`), backslash (`\`), bracket (`[]`), and semi-colon (`;`).

Dollar sign (`$`): The dollar sign identifies Tcl variables in Tcl but has no special meaning in Warehouse Builder. Therefore, if you include `$` in the name of a Warehouse Builder object, OMB Plus uses the Tcl convention and displays the `$` inside curly brackets such that `name$` displays as `name{}`. This prevents the name from being misinterpreted as a variable.

Backslash (`\`): Tcl uses the backward slash to indicate the end of a line in a command that spans multiple lines. Therefore, in a multiple line OMB Plus command, use only a backslash (`\`) at the end of each line. This interpretation of the backslash has

implications on how you specify full paths in OMB Plus, as described in ["Specifying Paths"](#) on page 4-3.

Tcl also uses the backslash as the least preferred method for escaping special characters. Use curly braces as the preferred method as described in ["Escaping Special Characters and Writing Complex Arguments"](#) on page 4-3.

Semi-colon(;): The semi-colon separates two commands in Tcl. Using a semi-colon in a quoted string results in an error. As a work around, escape the semi-colon character by putting a backward slash (\) in front of the semi-colon. For example,

```
OMBCREATE FLATFILE 'FF_DSR_RLE' \
SET PROPERTIES(DATA_FILE_NAME, IS_DEMILITED, CHARACTER_SET, RECORD_DELIMITER, \
FIELD_DELIMITER, FIELD_LEFT_ENCLOSURE, FIELD_RIGHT_ENCLOSURE) \
VALUES ('DSR_RLE.dat', 'TRUE', 'WE8MSWIN1252', '\n', '\;', '','', '')
```

Escaping Special Characters and Writing Complex Arguments

Tcl uses curly braces ({} and {}) as preferred method for escaping special characters and writing valid, complex arguments. For the following situations, you can enclose the element in braces and leave the contents alone:

- The element contains embedded spaces.
- The element contains one of the [Special Characters](#) on page 4-2.
- The element starts with a brace or double-quote.
- There are no characters in the element.

To escape single quote marks in elements, consider using the procedure *OMBToSettableString* described in ["Predefined Tcl Procedures"](#) on page 4-4.

You should consider using the backslash for escaping only in the limited situations that using curly braces results in unmatched braces, the last character of the argument is a backslash, or the element contains a backslash followed by another backslash indicating a new line.

Specifying Paths

Do not use a backward slash (\) when you specify the full path for the commands that use the full path, such as, OMBIMPORT, OMBVALIDATE, OMBLOG, and so on. For example, in the following commands are invalid and the log file is not created:

```
set OMBLOG c:\my_project\omb_logfile.log (On Windows)
set OMBLOG \home\my_project\omb_logfile.log (On Unix)
```

On Unix, use a forward slash as the path separator. For example, the following command creates a log file.

```
set OMBLOG /home/my_project/omb_logfile.log
```

On Windows, you can use either a forward slash(/) or two backward slashes (\\) as a path separator. Alternately, you can use a backward slash in the path, but in this case, enclose the entire filename in curly braces. The following are examples of commands that you can use to create a log file.

```
set OMBLOG c:/my_project/omb_logfile.log
set OMBLOG c:\\my_project\\omb_logfile.log
set OMBLOG {c:\my_project\omb_logfile.log}
```

Predefined Tcl Procedures

You can use the predefined Tcl procedures in OMB Plus:

- **OMBToSettableString:** Use this procedure when setting string values that contain single quotes that need to be escaped. The input for this procedure is a Tcl string and the output is a Tcl string with all single-quotes escaped.
- **OMBToTypeObjListString:** This procedure converts an input two-dimensional list to a comma-delimited string. For example, the procedure converts input in the form of

```
{{<object_type> <name>} ... }  
to  
"<object_type> <name>,..."
```

- **OMBPageBreak:** This procedure displays the input string as a sequence of pages, with a pause after each page. When the output of a command is more than the page height, it may be difficult for screen reading software (used for accessibility) to read the whole text. This procedure may be used to break the output of a command into pages.

The two inputs to the OMBPageBreak command are the number of lines to be displayed in a page and the string that is to be split into pages. The string may be the output of an OMB Plus command. For example, the following command displays the output of the OMBHELP OMBCREATE command with 10 lines in a page.

```
OMBPageBreak 10 [OMBHELP OMBCREATE]
```

To display the next 10 lines of the output, press <Enter> on your keyboard.

The OMPageBreak procedure is available for every OMB Plus session.

Running Scripts in OMB Plus

You can write scripts and run them in OMB Plus. For examples of scripts you can write, see "[Sample OMB Plus Scripts](#)" on page 4-11.

Inside the interactive shell, type `source test.tcl` where 'test' is the name of the script you want to run.

At the command line, type `OMBPlus.sh test.tcl` for scripts on UNIX and `OMBPlus.bat test.tcl` for scripts on Windows operating systems.

Locating Errors in Scripts and Multi-line Commands

OMB Plus reports only the first error it encounters while executing a command. As soon as it encounters the first error, it stops processing the command and exits reporting the error.

When an error occurs during the execution a multi-line OMB Plus command, the error message that is displayed does not specify the exact line at which the error occurred. To determine the line at which the error occurred, use the following command immediately after you encounter an error:

```
OMB+> puts $errorInfo
```

OMB Plus Commands

The sections that follow describe the types of commands that comprise the OMB Scripting Language.

- **Metadata Manipulation Language (MML) Commands:** Includes commands for creating, altering, deleting, and retrieving metadata objects.
- **Shell Commands:** Includes help and environment support such as `OMBDCC` and `OMBHELP`. Although these commands enable you to control the scripting environment, you cannot use them to edit the metadata.
- **Administrative Commands:** Fits the MML to the Warehouse Builder back end. For example, the commands `OMBCONNECT`, `OMBDISCONNECT`, `OMBCOMMIT`, or `OMBROLLBACK`.
- **Navigation Commands:** Enable you to navigate the Warehouse Builder repository just as you would navigate a UNIX file system.
- **Service Commands:** Enable you to start Warehouse Builder metadata services such as validation, compilation, deployment, and import or export.

For a list of new command introduced in this release, see "[Commands Introduced in This Release](#)" on page 4-14.

Metadata Manipulation Language (MML) Commands

OMB Plus enables you to create, modify, delete, and retrieve object metadata in Warehouse Builder design and runtime repositories. OMB Plus commands work within the context of a first class object. For a list of first class objects, see "[Warehouse Builder Metadata Objects](#)" on page 4-7.

[Table 4-1](#) lists the standard command names for MML.

Table 4-1 Metadata Manipulation Language Commands

Metadata Manipulation Language (MML)	Description
OMBCREATE	Creates a first class object.
OMBDROP	Deletes a first class object.
OMBALTER	Modifies a first class object.
OMBRETRIEVE	Retrieves information from a first class object.

The `OMBCREATE`, `OMBDROP`, `OMBALTER` and `OMBRETRIEVE` commands accept only the object name as the main argument. Names identified by absolute or relative path are not accepted. To use these commands you must in the parent context of the object to be created, dropped, altered, or retrieved.

OMB Plus executes commands like `OMBCREATE`, `OMBALTER`, and `OMBDROP` within a nested transaction.

OMB Plus interprets clauses within a single command one by one, as illustrated by the following example:

```
OMBCREATE TABLE 'T1' \
  MODIFY COLUMN 'C1' RENAME TO 'C1_NEW' \
  ADD UNIQUE_KEY 'UK1' \
  SET REF COLUMNS ('C1_NEW', 'C2')
```

In the preceding example, OMB Plus renames column C1 to C1_NEW when parsing the `modify_column` clause. In the last line, use the new name for the column, C1_NEW, to specify the referenced columns for the new unique key. For more details about synchronization of cached data, see ["Synchronizing Cached Data with Repository Objects"](#)

The `OMBCREATE` and `OMBRETRIEVE` commands synchronize only the first content object that they are currently working on. The `OMBCREATE` command synchronizes only the parent folder.

Examples

The following example lists the high-level scripting command syntax definitions for the `OMBCREATE` command:

```
OMBCREATE <fco_type> <fco_name> ( [ rename_clause ] [ properties_clause ] [ [ sco_
add_clause_for_alter ] | [ sco_modify_clause ] | [ sco_delete_clause ] ]* )1
rename_clause ::= RENAME TO <new_name>
sco_add_clause_for_alter ::= ADD <sco_type> <sco_name> [ OF parent_sco_clause ] [
AT POSITION <position> ] [ properties_clause ] [ references_clause ]*
sco_modify_clause ::= MODIFY <sco_type> <sco_name> [ OF parent_sco_clause ] ( [
rename_clause ] [ move_to_clause ] [ properties_clause ] [ references_clause ]* )1
move_to_clause ::= MOVE TO POSITION <position>
sco_delete_clause ::= DELETE <sco_type> <sco_name> [ OF parent_sco_clause ]
```

In the preceding example, the number 1 following a group of clauses enclosed by () brackets indicates that you must specify at least one of the clauses.

You can specify a particular Warehouse Builder object by tracing the aggregation relationship from its parent first class object. You can also capture the association relationships by the references clauses. For example, `getSCOClause`, where `sco_type` is the second class object type.

Each action, create, alter, drop, or retrieve works only on the properties and the immediate children of the currently specified object. For example, the retrieve command on a table only enables you to access the properties of the table and the lists of column and constraint names owned by that table. To drill down to the detailed descriptions of the columns and constraints, you can call retrieve on these objects respectively.

The following statement retrieves the data type and length for a column in a view:

```
OMBRETRIEVE VIEW 'V1' COLUMN 'COL1' \
GET PROPERTIES (DATATYPE, LENGTH)
```

When you set and retrieve properties using the `set_properties_clause` and the `get_properties_clause`, you can type the property names in any order.

Physical names are used as object identifiers in scripting. Business names represent an object property. Business names are not used to identify objects. You can identify a cross-component first class object by a path notation.

```
/<project_name>/<module_name>/<fco_name>
or
../<module_name> <fco_name>
```

String values, including object names and string property values, must be enclosed in single quotes.

Warehouse Builder Metadata Objects

Use OMB Plus to access and manipulate the following Warehouse Builder objects, also known as first class objects:

Table 4–2 Warehouse Builder Metadata Objects

Objects (column 1)	Objects (column 2)	Objects (column 3)
▪ Advanced Queues	▪ Functions	▪ Process Flows
▪ Collections	▪ Gateway_Modules	▪ Process FLOW Modules
▪ Connectors	▪ Locations	▪ Process Flow Packages
▪ Cube Tables	▪ Mappings	▪ Projects
▪ Deployment Action Plans	▪ Materialized Views	▪ Runtime Repository Connections
▪ Dimension Tables	▪ Object Types	▪ SAP Modules
▪ External Tables	▪ Oracle Modules	▪ Sequences
▪ Flat File	▪ Packages	▪ Snapshots
▪ Flat File Modules	▪ Procedures	▪ Tables

For Oracle Modules, you can access only those Oracle modules designated as warehouse modules. You cannot access Oracle source modules using OMB Plus.

Shell Commands

Shell commands provide you with an interactive interface to run all Warehouse Builder scripts and standard Tcl commands. OMB Plus shell commands include: OMBHELP, OMBCC, OMBDCC, and OMBENV.

OMBHELP

Use the OMBHELP command to display help on Warehouse Builder commands. The help describes the purpose of the command, the syntax in BNF format, and each of the keywords or options. For complex commands such as OMBCREATE, OMBALTER, and OMBRETRIEVE, you can specify an optional `fc0_type` parameter. OMBHELP then displays the detailed syntax for that particular parameter type. Each command also provides specific options that enable you to display sub-sections of the help page.

The syntax for OMBHELP is:

```
help ::= OMBHELP <command_name> [ <command_specific_options> ] [DETAIL]
```

For example, OMBHELP OMBCONNECT displays the following:

```
OMBCONNECT
Purpose
To connect to OWB repository.
Syntax
OMBCONNECT <user>/<password>@<host:port:SID>
where
  <user> is the OWB repository user name
  <password> is the OWB repository user password
  <host> is the name or IP address of the OWB repository host machine
  <port> is the numeric port for OWB repository database listener
  <SID> is the unique database identifier for OWB repository database
```

Notes:

The connection to OWB repository will be established in single user mode.

If you type `OMBHELP <command_name>` followed by `[DETAIL]`, OMB Plus displays the command purpose, prerequisites, syntax, descriptions for each keyword and parameter, and examples of how to use the command.

The OMBHELP command synchronizes only the only the FCO that you are currently working on.

OMBENV

The syntax for OMBENV is:

```
environment ::= OMBENV
```

This command lists the values for all Warehouse Builder-specific environment variables. [Table 4-3](#) lists the environmental variables. To set an environmental variable, use the Tcl `set` command. Use `unset` to unset an environmental variable.

Table 4-3 Warehouse Builder Environment Variables

Environment Variable	Meaning	Possible Values
OMBTIMER	Enables timing on each Warehouse Builder scripting command. The time is logged to a log file and to the console or shell.	A Tcl boolean value.
OMBLOG	Stores the filename for Warehouse Builder log file.	A valid filename including its path.
OMBPROMPT	Indicates whether OMB Plus will update the command prompt each time the you call OMBCC.	A Tcl boolean value.
OMBCONTINUE_ON_ERROR	Ignores errors that occur in any command that is part of a script and moves to the next command in the script.	A Tcl boolean value.

Administrative Commands

Use these commands to perform administrative jobs on a Warehouse Builder repository. The following commands are available: OMBCONNECT, OMBDISCONNECT, OMBCOMMIT, and OMBROLLBACK.

```
connect ::= OMBCONNECT <username>/<password>@<host>:<port>:<sid>
disconnect ::= OMBDISCONNECT
commit ::= OMBCOMMIT
rollback ::= OMBROLLBACK
```

Navigation Commands

You can use the following commands to navigate the Warehouse Builder repository in the same way you navigate a UNIX file system.

OMBCC

This command enables users to change context Up and Down the Warehouse Builder navigation tree. For example, when you type `...` the current context changes to the parent context. However, if the current context is a modified project, an error message prompts you to commit or rollback your changes.

OMBDCC

This command shows you the current context and the context type. The syntax for OMBDCC is:

```
display_current_context ::= OMBDCC
```

OMBLIST

The child first class objects for folders are listed under OMBLIST. Using this command on folders describes only the folder properties. Note also that the list command allows name matching by regular expression. If you do not include the regular expression, then OMBLIST displays all objects sorted alphabetically.

The generic syntax for OMBLIST in a folder context is:

```
list_folder ::= OMBLIST ( <child_type1_plural> | ... | <child_typeN_plural> ) [ name_
in_regexp ]
name_in_regexp ::= a name in regular expression.
```

For example, under the root context you have:

```
list_root ::= OMBLIST PROJECTS [ name_in_regexp ]
```

The OMBLIST command synchronizes all parent-child relations in the navigation tree.

Service Commands

Service commands perform services like batch operations on Warehouse Builder metadata. [Table 4-4](#) contains a list of service commands and their descriptions.

Table 4-4 Service Commands

Command	Description
OMBCOMPILE	Use this command to compile folders or first class objects such as tables, views, sequences, dimensions, and cubes.
OMBDEPLOY	This command provides deployment service.
OMBIMPORT	This command provides the metadata import service. You can only invoke the OMBIMPORT command from the root context. The four available modes are: CREATE_MODE (CREATE), REPLACE_MODE (REPLACE), UPDATE_MODE (UPDATE), and MERGE_MODE (INCREMENTALUPDATE). The default mode, if not specified in the command, is CREATE_MODE.
OMBVALIDATE	Use this command to validate folders or first class objects such as tables, views, sequences, dimensions, and cubes.

Synchronizing Cached Data with Repository Objects

When you start an OMB Plus session, data about the objects is fetched from the OWB repository and cached in the OMB Plus session. The cached data is synchronized with the data from the repository at certain predefined intervals.

The extent to which the objects are synchronized depends on the OMB Plus command that you execute. For example, some commands synchronize all the parent-child relationships in the navigation tree, whereas some commands synchronize only the first class object that they are currently working on.

- The OMBLIST command synchronizes all the parent-child relationships in the navigation tree.

- The OMBCREATE and OMBRETRIEVE commands synchronize only the first class object that they are currently working on.
- The OMBCREATE command synchronizes only the parent folder.

Consider the following example on synchronization of cached data. You open an OMB Plus session and a Design Center session. In the Design Center, you delete a flat file module called FFM1. You then undelete FFM1 and commit the changes. In the OMB Plus session, you perform the sequence of operations listed. The details of the result of the operation and the logic behind the result is explained:

1. OMBCC FFM1

The context is changed to the module FFM1.

2. OMBCREATE FLATFILE

The flat file is not created because OMBCREATE synchronizes the parent folder. When the synchronization is performed, the parent folder is not found in the cache.

3. OMBLIST FLAT_FILE_MODULE

FFM1 is listed as one of the modules. This is because OMBLIST synchronizes all the parent-child relationships in the navigation tree.

4. OMBCREATE FLATFILE

The flat file is created. This is because the undelete is now reflected in the cache because of the OMBLIST command.

How to Read Syntax Diagrams

Syntax diagrams are drawings that illustrate valid SQL syntax. To read a diagram, trace it from left to right, in the direction shown by the arrows. Commands and other keywords appear in UPPERCASE inside rectangles. Type them exactly as shown in the rectangles. Parameters appear in lowercase inside ovals. Variables are used for the parameters. Punctuation, operators, delimiters, and terminators appear inside circles.

If the syntax diagram has more than one path, you can choose any path to travel. For example, [Figure 4-1](#) shows a syntax diagram that indicates you can specify either ADD, MODIFY, or DELETE:

Figure 4-1 Syntax Diagram with Multiple Paths



If you have the choice of more than one keyword, operator, or parameter, syntax diagrams display the options in a vertical list. For example, in the syntax diagram shown in [Figure 4-2](#), you can specify one or more of the multiple parameters in the stack:

Figure 4-2 Syntax Diagram with Multiple Parameters

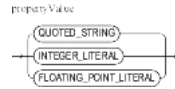


Required Keywords and Parameters

Required keywords and parameters can appear singly or in a vertical list of alternatives. Single required keywords and parameters appear on the *main path* – that is, on the horizontal line you are currently traveling.

If multiple keywords or parameters appear in a vertical list that intersects the main path, one of them is required. You must choose one of the keywords or parameters, but not necessarily the one that appears on the main path. In the example shown in [Figure 4–3](#), select one of the displayed settings:

Figure 4–3 Syntax Diagram with Multiple Choices for a Required Parameter



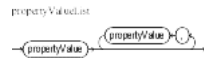
Optional Keywords and Parameters

If keywords and parameters appear in a vertical list preceding the main path, they are optional.

Syntax Loops

Loops enable you to repeat the syntax within them as many times as you like. In the example in [Figure 4–4](#), you can choose one property value and repeatedly to choose another. Separate your selections by commas.

Figure 4–4 Syntax Diagram with a Syntax Loop



Sample OMB Plus Scripts

Subsequent chapters in this guide contain syntax and examples specific to each command. This section contains lengthy examples that are appropriate in the context of a single OMB Plus command statement. These examples provide uninterrupted the series of steps for utilizing particular Warehouse Builder functionality. This section supplements but does not replace the syntax and related diagrams and information for each OMB Plus command.

This section includes sample scripts for the following tasks:

- [Updating a Design Repository](#)
- [Reporting on Repository Objects](#)
- [Finding Invalid Objects](#)
- [Using OMB Plus to Navigate Repositories](#)

Updating a Design Repository

One possible use case is to perform mass update on repository metadata. Users can write the following script to add a primary key with local column ID for each table with name beginning in EDW inside the module MY_MODULE:

```
OMBCC MY_MODULE;
foreach tableName [OMBLIST TABLE EDW*] { \
```

```

OMBCREATE TABLE '$tableName' \
ADD PRIMARY_KEY '$tableName_PK' SET REFERENCE COLUMNS ('ID');}

```

We can build even more powerful and useful script using if-then-else:

```

foreach tableName [OMBLIST TABLE EDW*] { \
set columnList [OMBRETRIEVE TABLE '$tableName' GET COLUMNS]; # Use lsearch to
search for a name in a list
if {[lsearch $columnList 'ID'] == -1} {
    OMBCREATE TABLE '$tableName' \
        ADD COLUMN 'ID' \
        SET PROPERTIES (DATATYPE, LENGTH, NOT_NULL) VALUES \
            ('NUMBER', 10, 'true');
}
}

```

The preceding script checks the list of tables which name begins with EDW whether each of them contains an ID column. If not, it will create an ID column for that table. Hence, executing the preceding script will guarantee that all tables with names beginning in EDW will have the ID column defined.

Reporting on Repository Objects

Another common use is for reporting purpose. The following script displays the properties of the table T1 and its column definitions on standard output:

```

#Displaying metadata of a table
puts -nonewline "Please enter the table name: " gets stdin tableName
puts ""
puts "Report on $tableName"
puts "======"
puts "Physical name = $tableName"
puts "Logical name = [lindex [OMBRETRIEVE TABLE '$tableName' GET \
PROPERTIES(BUSINESS_NAME)] 0]"
puts "Description = [lindex [OMBRETRIEVE TABLE '$tableName' GET \
PROPERTIES(DESCRIPTION)] 0]"
puts "-----"
set columnList [OMBRETRIEVE TABLE '$tableName' GET COLUMNS]
set i 1
foreach colName $columnList {
set dt [lindex [OMBRETRIEVE TABLE '$tableName' COLUMN '$colName' GET \
PROPERTIES(DATATYPE)] 0]
    if { $dt == "VARCHAR2" } {
        set prop [OMBRETRIEVE TABLE '$tableName' COLUMN '$colName' GET
PROPERTIES(LENGTH, NOT_NULL)]
        puts "Column $i: $colName datatype=VARCHAR2 length=[lindex $prop 0] \
not_null=[lindex $prop 1]"
    } elseif { $dt == "NUMBER" } {
        set prop [OMBRETRIEVE TABLE '$tableName' COLUMN '$colName' \
GET PROPERTIES(PRECISION, SCALE, NOT_NULL)]
        puts "Column $i: $colName datatype=NUMBER precision=[lindex $prop 0] \
scale=[lindex $prop 1] not_null=[lindex $prop 2]"
    } elseif { $dt == "DATE" } {
        set prop [OMBRETRIEVE TABLE '$tableName' COLUMN '$colName' GET \
PROPERTIES(NOT_NULL)]
        puts "Column $i: $colName datatype=DATE not_null=[lindex $prop 0]"
    } # end else
    incr i
}
}

```

A sample output is like the following:

```
Physical name = T1
```

```

Logical name = Table 1
Description = This is my first table.
=====
Column: ID datatype=NUMBER precision=0 scale=0 not_null=1
Column: NAME datatype=VARCHAR2 length=30 not_null=1
Column: VALUE datatype=VARCHAR2 length=100 not_null=0

```

Finding Invalid Objects

Users can also take advantage of the validation service provided by scripting, like this:

```

set tableList [OMBLIST TABLES];
foreach tableName $tableList {
  if { [OMBCOMPLETE TABLE '$tableName'] == "Invalid." } {
    set context [OMBDCC];
    puts "Table $context/$tableName is invalid.";
  }
}

```

The preceding script will tell users which table is invalid under the current module.

Using OMB Plus to Navigate Repositories

Another scenario we present is for a disabled user that relies on OMB Plus interactive shell (and also some screen reading software for the disabled) to navigate through a Warehouse Builder repository:

```

OMB+> OMBCONNECT owb/owb@localhost:1521:dev901
Connected.
OMB+> OMBLIST PROJECTS
DIM_TEST_PROJ MY_PROJECT PROJ_ENDTOEND PROJ_RELATIONAL TEST_DIM_PROJ
OMB+> OMBLIST PROJECTS .*RELATION.*
PROJ_RELATIONAL
OMB+> OMBCC 'PROJ_RELATIONAL'
Context changed.
OMB+> OMBDCC
PROJECT /PROJ_RELATIONAL
OMB+> set OMBPROMPT ON
ON
OMB+> OMBDCC
PROJECT /PROJ_RELATIONAL
/PROJ_RELATIONAL>
/PROJ_RELATIONAL> OMBLIST ORACLE_MODULES
WH
/PROJ_RELATIONAL> OMBCC 'WH'
Context changed.
/PROJ_RELATIONAL/WH> OMBLIST TABLES
PRODUCT PO
/PROJ_RELATIONAL/WH> OMBRETRIEVE TABLE 'PO' GET COLUMNS
OID PROD_ID ORDER_DATE CUSTNAME
/PROJ_RELATIONAL/WH> OMBCC '..'
Context changed.
/PROJ_RELATIONAL> OMBCC '..'
Context changed.
/>
/> OMBDISCONNECT
Disconnected.

```

New to OMB Plus in This Release

Refer to this section for information on newly introduced commands and changes to the scripting language.

Commands Introduced in This Release

Table 4–5 lists the commands introduced in this release.

Table 4–5 New OMB Plus Commands

Command Name	Brief Description
OMBCREATE	This command provides methods for connecting to a control center.
CONTROL_CENTER	This command replaces OMBCREATE RUNTIME_REPOSITORY.
.	.
.	.
.	.

Changes to the OMB Plus Syntax

Table 4–6 is a partial listing of changes to the OMB Plus syntax introduced in this release. Consult this section to update any scripts you may have written based on the syntax from a previous release.

Table 4–6 Changes in OMB Plus Scripting Syntax

OMB Plus Command, Keyword, or Clause	Changes Required for Existing Scripts
OMUCREATE MAPPING command	Some return values associated with this command have changed. You may need to modify existing scripts. Refer to the OMB Plus help for a description on the new syntax.
All OMU Commands	For all the OMU commands in general, return values have changed. Refer to the OMB Plus help for a description on the new syntax.
REAL_TIME_MAPPING keyword	Do not use the keyword REAL_TIME_MAPPING as the feature is no longer supported. The keyword was removed from syntax for OMBALTER COLLECTION and OMBCREATE COLLECTION.
PARALLEL_ROW_CODE property	Do not use the PARALLEL_ROW_CODE property as the functionality is no longer supported. The property was removed from OMBALTER DATA_AUDITOR,
DB_LOCATION property	This property was removed from OMBALTER GATEWAY_MODULE
OMBALTER IMPORT_ACTION_PLAN command	For this command, replace the keyword DELETE with UNSET.
CHECK_CONSTRAINT keyword	The CHECK_CONSTRAINT keyword was removed from OMB Plus commands for creating and altering views and materialized views as check constraints do not apply to those objects.
OMBALTER ORACLE_MODULE	Many properties associated with this command were removed. Additional information not available at print date.

Table 4–6 (Cont.) Changes in OMB Plus Scripting Syntax

OMB Plus Command, Keyword, or Clause	Changes Required for Existing Scripts
RUNTIME keyword	Replace the keyword RUNTIME with the keyword CONTROL_CENTER. It applies to OMB Plus commands associated with connecting and disconnecting to a Control Center.
createTimeDimensionComm and clause	The syntax for addSequenceClause has changed. Refer to the OMB Plus help for a description on the new syntax.
OMBEXPORT, OMBEXPORT MDL_FILE, and OMBIMPORT MDL_FILE commands	For all of these commands, do not enclose in brackets the keywords INCLUDE_GRANTS and INCLUDE_USER_DEFINITIONS. For OMBIMPORT MDL_FILE, also do not enclose NO_UPGRADE in brackets.
OMBRECONCILE command	Replace OMBRECONCILE with OMBSYNCHRONIZE.
OMUANALYZEIMPACT command	This command was removed.
OMUANALYZELINEAGE command	This command was removed.
OMUPROAGATECHANGE command	This command was removed.
OMUSHOWLIA command	This command was removed.

OMB Commands

This chapter contains an alphabetical listing of the navigation, service, administrative, and shell commands you can use in OMB Plus. Commands for metadata manipulation are contained in separate chapters.

OMBCAC

Purpose

Change Active Configuration command allows users to set as active another Configuration in a Project.

Prerequisites

Must be in a Project context.

Syntax

```
parseChangeActiveConfigurationCommand = OMBAC "QUOTED_STRING"
```

Examples

```
OMBCAC 'MY_CONFIGURATION_2'
```

This will set the 'MY_CONFIGURATION' configuration as active.

See Also

OMBDAC

OMBCC

Purpose

Change Context command allows users to change the current context to the desired location in OWB tree. The target context can be specified

either as an absolute path starting from the root ('/') or as a relative path starting from the current context. Also, the path can contain '..', which allows to navigate "up" to the parent context.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseChangeContextCommand = OMBCC "QUOTED_STRING"
```

Keywords And Parameters

parseChangeContextCommand

Specify change context command.

QUOTED_STRING

The target context to switch to.

Examples

```
OMBCC '/'
```

changes the context to the root.

```
OMBCC '/MY_PROJECT/ORACLE_1'
```

changes the context to Oracle module 'ORACLE_1', within project 'MY_PROJECT'.

```
OMBCC '..'
```

changes the context to the parent of current context (to the project level, if the current context is an Oracle module, for example).

See Also

OMBDCC

OMBCOMMIT

Purpose

Perform commit action on the repository.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseCommitCommand = OMBCOMMIT
```

Keywords And Parameters

`parseCommitCommand`
Specify commit command.

Examples

```
OMBCOMMIT
```

See Also

```
OMBROLLBACK
```

OMBCOMPARE SNAPSHOT

Purpose

Change management is a key piece of metadata management. This comand provides comparison services of any complex object model in the repository. This command writes the diff between snapshot/component to XML file.

Prerequisites

Snapshots can be compared from any context and either with another snapshot or the current repository objects.

Syntax

```
parseCompareCommand = OMBCOMPARE "compareSnapshotCommand"
compareSnapshotCommand = ( ( SNAPSHOT "QUOTED_STRING" ) WITH
    "getCompareWith" OUTPUT TO "QUOTED_STRING" WRITE ( ALL |
    FOUND_IN_TARGET | FOUND_IN_SOURCE | UPDATED | CHANGED | UNCHANGED ) )
getCompareWith = SNAPSHOT "QUOTED_STRING" [ FOR ( "UNQUOTED_STRING"
    "QUOTED_STRING" ) ] | ( CURRENT FOR ( "UNQUOTED_STRING"
    "QUOTED_STRING" ) )
```

Keywords And Parameters

`parseCompareCommand`

Root production of OMBCOMPARE SNAPSHOT.

`compareSnapshotCommand`

To compare components of snapshots.

`QUOTED_STRING`

Name of source snapshot which needs to be compared with the target snapshot.

`OUTPUT`

Specifies output filename where the XML comparison result will written.

`WRITE`

Specifies filter clause which will make the diff engine only write specified objects of a certain diff state.

`getCompareWith`

Target of the comparison.

SNAPSHOT

Target snapshot which will be compared with the source snapshot.

FOR

Specifies component which exists in the current repository.

CURRENT

Indicates current component's definition as the target of compare action.

Examples

```
OMBCOMPARE SNAPSHOT 'S1' WITH SNAPSHOT 'S2' OUTPUT TO 'd:diff.xml'  
WRITE
```

CHANGED

This command compares snapshot S1 with S2, and writes objects with CHANGED state into diff.xml, Though the diff engine allows to compare any two snapshot, even two unrelate snapshots with totally different components in them. CHANGED filter writes FOUND_IN_SOURCE or FOUND_IN_TARGET or UPDATED

components, CHANGED state represents whole diff,

```
OMBCOMPARE SNAPSHOT 'S1' WITH SNAPSHOT 'S2' OUTPUT TO 'd:diff.xml'  
WRITE
```

FOUND_IN_SOURCE

This command writes objects which are only found in snapshot S1.

```
OMBCOMPARE SNAPSHOT 'S1' WITH CURRENT FOR TABLE '/Project1/WH1/T1'  
OUTPUT
```

```
TO 'd:diff.xml' WRITE ALL
```

This command writes all table objects with any diff state.

See Also

OMBCREATE SNAPSHOT, OMBALTER SNAPSHOT, OMBDROP SNAPSHOT,
OMBRESTORE SNAPSHOT, OMBLIST SNAPSHOT, OMBRETRIEVE SNAPSHOT

OMBCOMPILE

Purpose

This command compiles an repository object. The results are generated in a file in a user defined directory.

Prerequisites

In the context of a Oracle Module.

Syntax

```

parseCompileCommand = OMB_COMPILE ( ( EXPERT | EXPERT_MODULE | TABLE | VIEW
    | SEQUENCE | MATERIALIZED_VIEW | DIMENSION | CUBE | DATA_AUDITOR |
    MAPPING | REAL_TIME_MAPPING | TRANSPORTABLE_MODULE |
    BUSINESS_PRESENTATION_MODULE | BUSINESS_DEFINITION_MODULE |
    EXTERNAL_TABLE | OBJECT_TYPE | NESTED_TABLE | VARYING_ARRAY |
    COLLECTION | CONNECTOR | PRESENTATION_TEMPLATE |
    ALTERNATIVE_SORT_ORDER | LIST_OF_VALUES | DRILL_TO_DETAIL |
    BUSINESS_AREA | DRILL_PATH | ITEM_FOLDER | REGISTERED_FUNCTION |
    PACKAGE | FUNCTION | PROCEDURE | TABLE_FUNCTION | PLSQL_RECORD_TYPE |
    PLSQL_TABLE_TYPE | PLSQL_REF_CURSOR_TYPE | PROCESS_FLOW_PACKAGE |
    ADVANCED_QUEUE | STREAMS_QUEUE | QUEUE_TABLE | QUEUE_PROPAGATION |
    STREAMS_CAPTURE_PROCESS | PROCESS_FLOW_PACKAGE | PROCESS_FLOW_MODULE |
    CALENDAR | CALENDAR_MODULE ) "QUOTED_STRING" [
    "getOutputValidationResults" ] [ "getOutputGeneratedScripts" ] )
getOutputValidationResults = OUTPUT [ VALIDATION_RESULT ] TO (
    "QUOTED_STRING" | ( FILE "QUOTED_STRING" ) ) WRITE ( ( "(" ( ( SUCCESS
    | WARNING | ERROR ) [ "," ] )+ ")" ) | ALL | SUCCESS | WARNING |
    ERROR )
getOutputGeneratedScripts = OUTPUT GENERATION_SCRIPTS TO ( "QUOTED_STRING"
    | ( FILE "QUOTED_STRING" ) )

```

Keywords And Parameters

`parseCompileCommand`

This command compiles a repository object.

`QUOTED_STRING`

The name of the object.

`getOutputValidationResults`

This clause outputs the validation results to one or more files in the specified folder.

`QUOTED_STRING`

A file or directory where validation results are stored.

`getOutputGeneratedScripts`

This clause outputs the generated scripts for an object to one or more files in specified folder.

QUOTED_STRING

A file or directory where generated scripts are stored.

Examples

```
OMB_COMPILE TABLE 'T1' OUTPUT VALIDATION_RESULT TO '/tmp' WRITE  
SUCCESS
```

```
OUTPUT GENERATION_SCRIPTS TO '/tmp'.
```

```
OMB_COMPILE TABLE 'T1' OUTPUT GENERATION_SCRIPTS TO '/tmp'
```

The first example gets the validation results and generated scripts for the table, whereas the second example gets only the generated scripts.

See Also

OMB_VALIDATE

OMBCONN

Purpose

To connect to a OWB repository.

Prerequisites

Must not be connected to another OWB repository. If connected to another repository, use OMBDISCONNECT to disconnect first.

Syntax

```
parseConnectCommand = ( ( OMBCONNECT | OMBCONN ) "UNQUOTED_STRING" [ USE (
    REPOSITORY | REPOS ) "QUOTED_STRING" ] [ USE ( SINGLE_USER_MODE |
    MULTIPLE_USER_MODE ) ] )
```

Keywords And Parameters

`parseConnectCommand`

Specify connect command.

`UNQUOTED_STRING`

Specify the connection string to the database, in the format:

`username/password@host:port:service name`

`QUOTED_STRING`

Optionally, specify the name of a repository to work on. If not provided, the default repository will be used.

`SINGLE_USER_MODE`

If specified, the user will use the repository exclusively.

`MULTIPLE_USER_MODE`

If specified, more than one session can work on the same repository at the same time. This is the default mode.

Examples

```
OMBCONNECT owb_normal_user/welcome@dwsun42:1521:dev817 USE
REPOSITORY
```

```
'owb_repos' USE SINGLE_USER_MODE
```

will connect a normal user `owb_normal_user` to database and work on repository named `'owb_repos'` in single user mode.

See Also

OMBDISCONNECT

OMBCONNECT

Purpose

To connect to a OWB repository.

Prerequisites

Must not be connected to another OWB repository. If connected to another repository, use OMBDISCONNECT to disconnect first.

Syntax

```
parseConnectCommand = ( ( OMBCONNECT | OMBCONN ) "UNQUOTED_STRING" [ USE (
    REPOSITORY | REPOS ) "QUOTED_STRING" ] [ USE ( SINGLE_USER_MODE |
    MULTIPLE_USER_MODE ) ] )
```

Keywords And Parameters

`parseConnectCommand`

Specify connect command.

`UNQUOTED_STRING`

Specify the connection string to the database, in the format:

`username/password@host:port:service name`

`QUOTED_STRING`

Optionally, specify the name of a repository to work on. If not provided, the default repository will be used.

`SINGLE_USER_MODE`

If specified, the user will use the repository exclusively.

`MULTIPLE_USER_MODE`

If specified, more than one session can work on the same repository at the same time. This is the default mode.

Examples

```
OMBCONNECT owb_normal_user/welcome@dwsun42:1521:dev817 USE
REPOSITORY
```

```
'owb_repos' USE SINGLE_USER_MODE
```

will connect a normal user `owb_normal_user` to database and work on repository named `'owb_repos'` in single user mode.

See Also

OMBDISCONNECT

OMBCONNECT CONTROL_CENTER

Purpose

To connect to a Control Center.

This command can either be used when you are already connected to a Design Repository or it can be used independently of a Design Repository.

With Design Repository connection:

This command will connect you to the Control Center that is associated with the current Active Configuration.

If this is the DEFAULT_CONTROL_CENTER and it does not have a user and password specified, then you will be connected to the Control Center as the currently connected Design Repository User. This is the typical out-of-the-box scenario. If the DEFAULT_CONTROL_CENTER has a user specified then you will be connected as that user. Note in this case you may have to provide a password.

If this is not the DEFAULT_CONTROL_CENTER, then you will either connect as the user specified against the Control Center or you will have to provide a user with this command. Note that in this situation you will probably have to supply a password.

Without Design Repository connection:

This command can be used to connect to a Control Center independently of a Design Repository. For example, you may want to OMBDEPLOY a deployment specification that had been previously deployed to the file system or OMBSTART a job. In this case you will have to provide the complete connection details.

Prerequisites

With Design Repository connection:

A Design Repository connection, at least a PROJECT context and the Active Configuration's Selected CONTROL_CENTER must be defined.

Without Design Repository connection:

There must not be a Design Repository connection whenever this method is used.

Syntax

```
parseConnectRuntimeCommand = ( ( OMBCONNECT | OMBCONN ) CONTROL_CENTER [ (
    USE "QUOTED_STRING" ) | ( "UNQUOTED_STRING" USE ( REPOSITORY | REPOS
    ) "QUOTED_STRING" ) ] )
```

Keywords And Parameters

parseConnectRuntimeCommand

Specify Control Center connect command.

QUOTED_STRING

Specific Control Center:

Examples

With Design Repository connection:

```
OMBCONNECT CONTROL_CENTER USE
'RepositoryUserName/RepositoryUserPassword'
```

```
OMBCONNECT CONTROL_CENTER USE 'RepositoryUserPassword'
```

```
OMBCONNECT CONTROL_CENTER
```

Without Design Repository connection:

```
OMBCONNECT CONTROL_CENTER
```

```
RepositoryUserName/RepositoryUserPassword@HostName:PortNumber:OracleServiceName
```

```
USE REPOSITORY 'RepositorySchema'
```

See Also

OMBDISCONNECT CONTROL_CENTER

OMBCONN CONTROL_CENTER

Purpose

To connect to a Control Center.

This command can either be used when you are already connected to a Design Repository or it can be used independently of a Design Repository.

With Design Repository connection:

This command will connect you to the Control Center that is associated with the current Active Configuration.

If this is the DEFAULT_CONTROL_CENTER and it does not have a user and password specified, then you will be connected to the Control Center as the currently connected Design Repository User. This is the typical out-of-the-box scenario. If the DEFAULT_CONTROL_CENTER has a user specified then you will be connected as that user. Note in this case you may have to provide a password.

If this is not the DEFAULT_CONTROL_CENTER, then you will either connect as the user specified against the Control Center or you will have to provide a user with this command. Note that in this situation you will probably have to supply a password.

Without Design Repository connection:

This command can be used to connect to a Control Center independently of a Design Repository. For example, you may want to OMBDEPLOY a deployment specification that had been previously deployed to the file system or OMBSTART a job. In this case you will have to provide the complete connection details.

Prerequisites

With Design Repository connection:

A Design Repository connection, at least a PROJECT context and the Active Configuration's Selected CONTROL_CENTER must be defined.

Without Design Repository connection:

There must not be a Design Repository connection whenever this method is used.

Syntax

```
parseConnectRuntimeCommand = ( ( OMBCONNECT | OMBCONN ) CONTROL_CENTER [ (
    USE "QUOTED_STRING" ) | ( "UNQUOTED_STRING" USE ( REPOSITORY | REPOS
    ) "QUOTED_STRING" ) ] )
```

Keywords And Parameters

parseConnectRuntimeCommand

Specify Control Center connect command.

QUOTED_STRING

Specific Control Center:

Examples

With Design Repository connection:

```
OMBCONNECT CONTROL_CENTER USE
'RepositoryUserName/RepositoryUserPassword'
```

```
OMBCONNECT CONTROL_CENTER USE 'RepositoryUserPassword'
```

```
OMBCONNECT CONTROL_CENTER
```

Without Design Repository connection:

```
OMBCONNECT CONTROL_CENTER
```

```
RepositoryUserName/RepositoryUserPassword@HostName:PortNumber:OracleServiceName
```

```
USE REPOSITORY 'RepositorySchema'
```

See Also

OMBDISCONNECT CONTROL_CENTER

OMBCOPY

Purpose

Copy one or more objects of the same object type. The replace option enables you to overwrite.

Prerequisites

Use of relative path specifications requires awareness of the current context.

Syntax

```
parseCopyCommand = OMBCOPY "copyObjectType" "QUOTED_STRING" TO
                    "QUOTED_STRING" [ USE REPLACE_MODE ]
copyObjectType = ( "UNQUOTED_STRING" )
```

Keywords And Parameters

parseCopyCommand

Specifies the source object type, source path, and target path for the object to copy. Copying objects is subject to the following restrictions:

1. You cannot copy an entire project.
2. When copying objects between projects, you can only copy objects into the current project; you cannot copy objects out to other projects.

QUOTED_STRING

Source and target path specifications can be absolute or relative. To copy multiple objects, include a regular expression as the final step of the source path. If you are copying multiple objects, the final step of the target path must be the folder to which the objects are being copied. If you are only copying one object, you can specify the object's original name or a new name as the final step of the target path.

REPLACE_MODE

Use this option to overwrite existing target objects.

copyObjectType

The type of the object(s) to be copied. Valid types are: PROJECT, ORACLE_MODULE, FLAT_FILE_MODULE, BUSINESS_DEFINITION_MODULE, BUSINESS_PRESENTATION_MODULE, SAP_MODULE, CMI_MODULE, PROCESS_FLOW_MODULE,

PROCESS_FLOW_PACKAGE, PROCESS_FLOW, EXPERT_MODULE, EXPERT,
LOCATION,
CONTROL_CENTER, CONFIGURATION, FLAT_FILE, ADVANCED_QUEUE,
STREAMS_QUEUE,
QUEUE_TABLE, QUEUE_PROPAGATION, STREAMS_CAPTURE_PROCESS,
OBJECT_TYPE,
VARYING_ARRAY, NESTED_TABLE, TABLE, VIEW, MATERIALIZED_VIEW,
SEQUENCE,
DIMENSION, CUBE, DATA_AUDITOR, DATA_PROFILE, DATA_RULE, DATA_
RULE_MODULE,
MAPPING, REAL_TIME_MAPPING, PACKAGE, FUNCTION, PROCEDURE,
BUSINESS_AREA,
COLLECTION, EXTERNAL_TABLE, REGISTERED_FUNCTION, ITEM_FOLDER,
DRILL_PATH,
LIST_OF_VALUES, DRILL_TO_DETAIL, ALTERNATIVE_SORT_ORDER,
PRESENTATION_TEMPLATE and any user defined object types.

Examples

```
OMBCOPY TABLE 'MY_TABLE1' TO 'MY_TABLE2' USE REPLACE_MODE  
OMBCOPY ORACLE_MODULE '/MY_PROJECT/WH1' TO '/MY_PROJECT/WH2'  
OMBCOPY TABLE 'MY_.*' TO '/MY_PROJECT/WH2'
```

See Also

OMBMOVE

OMBDAC

Purpose

Display the name of the active configuration of a project.

Prerequisites

Must be in the context of a project.

Syntax

```
parseDisplayActiveConfigurationCommand = OMBDAC
```

Keywords And Parameters

`parseDisplayActiveConfigurationCommand`
Specify Display Active Configuration command.

Examples

OMBDAC
This will display a name of the active configuration of a project.

See Also

OMBCAC

OMBDC

Purpose

Display Current Context command displays the current context (location) and, for contexts other than the root ('/'), the type of the current folder.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseDisplayCurrentContextCommand = OMBDC
```

Keywords And Parameters

parseDisplayCurrentContextCommand
Specify display current context command.

Examples

```
OMBDC  
will display  
/  
if the current context is the root.  
OMBDC  
will display  
PROJECT /MY_PROJECT  
if the current context is the project 'MY_PROJECT'.
```

See Also

OMBCC

OMBDEBUG MAPPING

Purpose

Debug a mapping by stepping through map execution one operator at a time.

Prerequisites

The current context must be in an Oracle module which contains at least one map.

Syntax

```
debugMappingCommand = OMBDEBUG MAPPING "mappingName" "debugClause"+
mappingName = "QUOTED_STRING"
debugClause = START [ DEBUG ] SESSION | END [ DEBUG ] SESSION | STEP | SKP
              | RESUME | DISPLAY ( [ "INTEGER_LITERAL" ROWS OF ] [
              "dataDisplayGroupDirection" ] DEBUG_DATA [ FOR "groupBottomUpLocator"
              ] | CURRENT_STEP_GROUP ) | SET BREAKPOINT AT "operatorBottomUpLocator"
              | CLEAR BREAKPOINT AT "operatorBottomUpLocator" | REINITIALIZE |
              RESET
dataDisplayGroupDirection = INPUT | OUTPUT
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
operatorBottomUpLocator = OPERATOR "operatorName" [
              "pluggableMapBottomUpLocator" ]
groupName = "QUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
              "pluggableMapBottomUpLocator" ] )
pluggableMapName = "QUOTED_STRING"
```

Keywords And Parameters

mappingName

Name of the mapping.

groupBottomUpLocator

Location of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

groupName

Name of a mapping group.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

pluggableMapName

Name of the pluggable map.

Examples

```
#
# Script to run the mapping debugger for map
/OWB_MAPPING_TS/TEST_MODULE/JOIN7_MAP_INSERT
# Assumes that this map has been created, and has an operator 'JOIN1'
# This map is created by the 'mapping codegen' development acceptance test
#

OMBCONNECT rep_XXXX/rep_XXXX@localhost:1521:ora10.us.oracle.com

OMBCC '/OWB_MAPPING_TS/TEST_MODULE'
OMBCONNECT CONTROL_CENTER 'DEFAULT_CONTROL_CENTER'
# puts "CONNECTED"

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' START SESSION

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' SET BREAKPOINT AT OPERATOR
'JOIN1'

#
# Resume execution until breakpoint at "JOIN1" is hit
#

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' RESUME

#
# Step through one row of data
#

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' STEP

#
# Step through 2 rows of data
#
```

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' STEP STEP

Display the debug data for the current step group. Should be 3 rows,
since we have stepped
3 rows into this operator.
WARNING: will not work if "DATA" is used instead of "DEBUG_DATA",
since "DATA" is already a reserved keyword used for another command
OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' DISPLAY DEBUG_DATA

Display 10 rows of output data for group "EMP.INOUTGRP1"

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' DISPLAY 10 ROWS OF OUTPUT
DEBUG_DATA
FOR GROUP 'INOUTGRP1' OF OPERATOR 'EMP'

Set the column width for debug data to 10 (default is 20)

set COLUMN_WIDTH 10
OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' DISPLAY 10 ROWS OF OUTPUT
DEBUG_DATA
FOR GROUP 'INOUTGRP1' OF OPERATOR 'EMP'

Reset the debug session - reinitialize data and start execution at
beginning of map

#OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' RESET

Reinitialize the debug session - regenerate and deploy debug code. Do
this if the map
is edited.

#OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' REINITIALIZE

#

Skip through the first operator, next step will go to next operator in graph.

#

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' SKIP

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' DISPLAY CURRENT_STEP_GROUP

OMBDEBUG MAPPING 'JOIN7_MAP_INSERT' END SESSION

OMBDEFINE ASSOCIATION_DEFINITION

Purpose

To define an association between two classes (types).

Prerequisites

Association definition to be defined should not already exist. User must have CREATE_EXTENSIONMODEL system privilege and has to be connected in single user mode to run this command.

Syntax

```
parseDefineAssociationCommand = OMBDEFINE ASSOCIATION_DEFINITION
    "QUOTED_STRING" "setAssociationDefinitionPropertiesClause" {
        "addDependencyDefinitionClause" }
setAssociationDefinitionPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
addDependencyDefinitionClause = ADD DEPENDENCY_DEFINITION "QUOTED_STRING"
    [ "setDependencyDefinitionPropertiesClause" ]
propertyNameList = "propertyNameClause" { "," "propertyNameClause" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setDependencyDefinitionPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
propertyNameClause = ( "UNQUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parseDefineAssociationCommand

Define an association between two classes..

setAssociationDefinitionPropertiesClause

Basic properties for ASSOCIATION_DEFINITION:

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: N/A

A descriptive text for this association.

Name: CLASS_1

Type: STRING(200)

Valid Values: N/A

Default: N/A

Class on one side of the association.

NAME: CLASS_2

Type: STRING(200)

Value Values: N/A

Default: N/A

Class on the other side of the association.

Name: ROLE_1

Type: STRING(200)

Valid Values: N/A

Default: N/A

Role on one side of the association.

Name: ROLE_2

Type: STRING(200)

Valid Values: N/A

Default: N/A

Role on the other side of the association.

Name: ROLE_1_MIN_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Minimum cardinality of role 1. Value can be positive integer.

Name: ROLE_1_MAX_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Maximum cardinality of role 1. Value can be positive integer or 'INFINITE'.

Name: ROLE_1_NAVIGABLE

Type: BOOLEAN

Valid Values: true, false

Default: From user defined class side default is true. From OWB class side default is false.

Navigability of role 1.

Name: ROLE_2_MIN_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Minimum cardinality of role 2. Value can be positive integer.

Name: ROLE_2_MAX_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Maximum cardinality of role 2. Value can be positive integer or 'INFINITE'.

Name: ROLE_2_NAVIGABLE

Type: BOOLEAN

Valid Values: true, false

Default: From user defined class side default is true. From OWB class side default is false.

Navigability of role 2.

addDependencyDefinitionClause

Mark this association so that the dependency engine will consider it when computing the lineage and impact dependencies. The only dependency type allowed here for now is 'DATAFLOW'.

propertyNameList

The list of property names.

propertyValueList

The list of property values being set.

setDependencyDefinitionPropertiesClause

Basic dependency-related properties for this association:

Name: SOURCE_ROLE_ID

Type: STRING(200)

Valid Values: ROLE_1, ROLE_2

Default: If one of the ends is a OWB class, then that is the default source. If both ends are user defined classes, then the association is default bidirectional.

Identifies the role (end) of the association which serves as the source for the dependency flow.

Name: BIDIRECTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: If one of the ends is a OWB class, then default is false. If both ends are user defined classes, then default is true.

Specifies whether the association is bi-directional for the dependency flow.

propertyNameClause

Name of a property.

propertyValue

Value of a property.

Examples

```
OMBDEFINE ASSOCIATION_DEFINITION 'UD_ASSOC1'  
SET PROPERTIES (CLASS_1, CLASS_2, ROLE_1, ROLE_2, ROLE_1_MAX_  
CARDINALITY,  
ROLE_1_NAVIGABLE) VALUES ('UD_REPORT', 'TABLE', 'REF_TABLES', 'REF_  
REPORT',  
'INFINITE', 'true') ADD DEPENDENCY_DEFINITION 'DATAFLOW'
```

This will define a new association between class UD_REPORT and TABLE, and define DATAFLOW dependency on this association.

See Also

OMBDESCRIBE ASSOCIATION_DEFINITION

OMBDEFINE CLASS_DEFINITION

Purpose

To define a class (user defined object type).

Prerequisites

Class definition to be defined should not already exist.

Syntax

```

parseDefineClassCommand = OMBDEFINE [ ( FIRST_CLASS_OBJECT |
    SECOND_CLASS_OBJECT | FOLDER | MODULE ) ] CLASS_DEFINITION
    "QUOTED_STRING" [ ( "setClassDefinitionPropertiesClause" |
        "setClassDefinitionIconSetClause" ) ] [ "addSubDefinitionsClause" ]
setClassDefinitionPropertiesClause = SET PROPERTIES "(" "propertyNameList"
    ")" VALUES "(" "propertyValueList" ")"
setClassDefinitionIconSetClause = SET REF ICONSET "QUOTED_STRING"
addSubDefinitionsClause = ( "addPropertyDefinitionClause" |
    "addPropertyGroupDefinitionClause" | "addChildTypeClause" )+
propertyNameList = "propertyNameClause" { "," "propertyNameClause" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addPropertyDefinitionClause = ( ADD [ ( ( CONFIGURATION |
    PHYSICAL_CONFIGURATION ) | LOGICAL | USER_DEFINED ) ]
    PROPERTY_DEFINITION "QUOTED_STRING"
    "setPropertyDefinitionPropertiesClause" )
addPropertyGroupDefinitionClause = ADD PROPERTY_GROUP_DEFINITION
    "QUOTED_STRING" "setPropertyGroupDefinitionPropertiesClause"
addChildTypeClause = ( ADD CHILD_TYPE "QUOTED_STRING" )
propertyNameClause = ( "UNQUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setPropertyDefinitionPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
setPropertyGroupDefinitionPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"

```

Keywords And Parameters

parseDefineClassCommand

Define a class.

FOLDER

Class definition is folder.

MODULE

Class definition is module.

QUOTED_STRING

Name of the class.

setClassDefinitionPropertiesClause

Set the properties on the class definition. Valid properties are BUSINESS_NAME, DESCRIPTION, DISPLAY_NAME, PLURAL_NAME.

setClassDefinitionIconSetClause

Assign an icon to the class definition.

propertyNameList

The list of property names.

propertyValueList

The list of values.

addPropertyDefinitionClause

Add a property definition to the class definition.

addPropertyGroupDefinitionClause

Add a property group definition to the class definition.

addChildTypeClause

Add a aggregate child type to the class definition.

propertyNameClause

The name of the property.

UNQUOTED_STRING

The name of the property for the class definition.

propertyValue

The value of the property.

QUOTED_STRING

The value in string format of the property for the class definition.

INTEGER_LITERAL

The integer value of the property for the class definition.

FLOATING_POINT_LITERAL

The float value of the property for the class definition.

setPropertyDefinitionPropertiesClause

Set the properties for the property definition. Valid properties are TYPE, DEFAULT_VALUE, POSITION, HIDDEN.

Examples

```
OMBDEFINE MODULE CLASS_DEFINITION 'UD_MODULE1'  
SET PROPERTIES (DISPLAY_NAME, PLURAL_NAME) VALUES ('FINANCE',  
'FINANCES')
```

```
ADD PROPERTY_DEFINITION 'P1'
```

```
SET PROPERTIES (TYPE, DEFAULT_VALUE) VALUES ('INTEGER', '2')
```

This will define a new module class, and has one property definition whose type is integer and default value is 2.

```
OMBDEFINE FIRST_CLASS_OBJECT CLASS_DEFINITION 'UD_FCO1'  
SEY PROPERTIES (DISPLAY_NAME, PLURAL_NAME) VALUES ('REPORT',  
'REPORTS')
```

This will define a new first class object class definition.

See Also

OMBDESCRIBE CLASS_DEFINITION

OMBDEFINE COMPONENT_DEFINITION

Purpose

To define a component definition.

Prerequisites

The class definition for the component should already exist. The class must be a first class object.

Syntax

```
parseDefineComponentCommand = OMBDEFINE COMPONENT_DEFINITION
    "QUOTED_STRING" [ "addChildClassesClause" ]
addChildClassesClause = "addChildClassClause"+
addChildClassClause = ( ADD "QUOTED_STRING" )
```

Examples

```
OMBDEFINE COMPONENT_DEFINITION 'UD_FCO1'
ADD 'UD_SCO1'
```

This will define a component for class UD_FCO1, and has one second class definition in the component.

OMBDEFINE DOMAIN_DEFINITION

Purpose

To define a domain.

Prerequisites

Domain is a type with constraints on some other type. For example, you can define a domain which contains a list of valid values of some existing type, or you can define a domain which contains ranges of some existing type.

Syntax

```

parseDefineDomainCommand = OMBDEFINE DOMAIN_DEFINITION "QUOTED_STRING"
    "setDomainPropertiesClause" { "addRangeClause" | "addValueClause" }
setDomainPropertiesClause = SET PROPERTIES "(" "propertyNameList" ")"
    VALUES "(" "propertyValueList" ")"
addRangeClause = ADD RANGE_DEFINITION "QUOTED_STRING"
    "setRangePropertiesClause"
addValueClause = ADD DOMAIN_VALUE "QUOTED_STRING"
propertyNameList = "propertyNameClause" { "," "propertyNameClause" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setRangePropertiesClause = SET PROPERTIES "(" "propertyNameList" ")"
    VALUES "(" "propertyValueList" ")"
propertyNameClause = ( "UNQUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

`parseDefineDomainCommand`

Define a domain.

`QUOTED_STRING`

Name of the domain.

`setDomainPropertiesClause`

Set the properties on the domain. Valid properties are `BASE_TYPE`, `DESCRIPTION`.

`addRangeClause`

Add range to the domain.

`addValueClause`

Add value to the domain.

propertyNameList

The list of property names.

propertyValueList

The list of values.

setRangePropertiesClause

Set the properties on the range. Valid properties are MIN_VALUE_STRING, MIN_VALUE_INCLUSIVE, MAX_VALUE_STRING, MAX_VALUE_INCLUSIVE, DESCRIPTION.

MIN_VALUE_STRING is the minimum value for this range. MIN_VALUE_INCLUSIVE

equal to true means the minimum value is in the range. MAX_VALUE_STRING is the maximum for this range. MAX_VALUE_INCLUSIVE equal to true means the maximum value is in the range.

propertyNameClause

The name of the property.

UNQUOTED_STRING

The name of the property for the class definition.

propertyValue

The value of the property.

QUOTED_STRING

The value in string format of the property for the class definition.

INTEGER_LITERAL

The integer value of the property for the class definition.

FLOATING_POINT_LITERAL

The float value of the property for the class definition.

Examples

```
OMBDEFINE DOMAIN_DEFINITION 'UD_DOMAIN1'  
SET PROPERTIES (BASE_TYPE) VALUES ('INTEGER') ADD RANGE_DEFINITION  
'R1'
```

```
SET PROPERTIES (MIN_VALUE_STRING, MAX_VALUE_STRING) VALUES ('1', '9')
```

This will define a new domain of integer type, and has one range of value from 1 to 9.

```
OMBDEFINE DOMAIN_DEFINITION 'UD_DOMAIN2'
```

```
SET PROPERTIES (BASE_TYPE) VALUES ('STRING') ADD DOMAIN_VALUE 'string  
value
```

```
1'
```

```
ADD DOMAIN_VALUE 'string value 2'
```

This will define a new domain of string type, and has 2 valid values 'string value 1' and 'string value 2'.

See Also

```
OMBDESCRIBE DOMAIN_DEFINITION
```

OMBDEFINE FOLDER_DEFINITION

Purpose

To define a folder definition.

Prerequisites

The class definition for the folder should already exist. The class must be a folder.

Syntax

```
parseDefineFolderCommand = OMBDEFINE FOLDER_DEFINITION "QUOTED_STRING" [  
    "addChildClassesClause" ]  
addChildClassesClause = "addChildClassClause"+  
addChildClassClause = ( ADD "QUOTED_STRING" )
```

Examples

```
OMBDEFINE FOLDER_DEFINITION 'UD_MODULE1'  
ADD 'UD_FCO1'
```

This will define a folder for class UD_MODULE1, and has one class definition in the folder.

OMBDEINSTALL OWB_REPOSITORY

Purpose

Drop or deinstall OWB repository.

Prerequisites

To deinstall OWB repository, within the same OMBPlus session, no other OMB commands should be issued before or after the OMBDEINSTALL OWB_REPOSITORY command.

Syntax

```
parseDeinstallOWBRepositoryCommand = OMBDEINSTALL OWB_REPOSITORY  
    "UNQUOTED_STRING" USING CREDENTIAL "UNQUOTED_STRING" [  
    NET_SERVICE_NAME "QUOTED_STRING" ]
```

Keywords And Parameters

`parseDeinstallOWBRepositoryCommand`

Specify an OMBDEINSTALL OWB_REPOSITORY command.

`UNQUOTED_STRING`

Specify the connection string to the database using this format:

username/password@host:port:service name, OR specify the user name and password pair using this format: username/password.

Examples

1. OMBDEINSTALL OWB_REPOSITORY h_rep5/h USING CREDENTIAL
sys/sys@localhost:1521:orcl92

This drops or de-installs an OWB repository.

OMBDEINSTALL OWB_TARGET_USER

Purpose

Drop or de-install an OWB target user.

Prerequisites

OWB REPOSITORY must exist on the database server.

Syntax

```
parseDeinstallTargetUserCommand = OMBDEINSTALL OWB_TARGET_USER  
    "UNQUOTED_STRING" USING OWB_REPOSITORY "UNQUOTED_STRING" USING  
    CREDENTIAL "UNQUOTED_STRING" [ NET_SERVICE_NAME "QUOTED_STRING" ]
```

Keywords And Parameters

`parseDeinstallTargetUserCommand`

Specify an OMBINSTALL OWB_TARGET_USER command.

UNQUOTED_STRING

Specify the connection string to the database using this format:

username/password@host:port:service name OR specify the user name and password using this format username/password.

Examples

1. OMBDEINSTALL OWB_TARGET_USER h_tu1/h USING OWB_REPOSITORY h_rep1/h

USING CREDENTIAL sys/sys@localhost:1521:orcl92

This drops or de-installs a target user.

OMBDEPLOY

Purpose

To Deploy Action Plans to Control Centers or File Systems.

Prerequisites

If the Deployment is NOT from a Deployment Specification XML file then the following are true. A Control Center connection and a named Deployment Action Plan are required. Also, the current Context must be either an Oracle Module, a Process Flow Module or a Location.

Syntax

```
parseDeployCommand = OMBDEPLOY ( ( DEPLOYMENT_ACTION_PLAN "QUOTED_STRING"
  [ ( AS ( SPECIFICATION | SCRIPT ) TO "QUOTED_STRING" ) |
    CONTROL_CENTER_ONLY ] ) | ( SPECIFICATION FROM "QUOTED_STRING" ) ) [
  ASYNCHRONOUS ]
```

Keywords And Parameters

`parseDeployCommand`

Specify Deploy command.

`DEPLOYMENT_ACTION_PLAN`

Deploy a Deployment Action Plan.

`QUOTED_STRING`

The Deployment Action Plan name. Or, the directory (if AS SCRIPT) or file name (if AS SPECIFICATION) into which the deployment is performed. Or, the source XML filename (if SPECIFICATION FROM).

`AS`

Perform the Deployment to a File System.

`SPECIFICATION`

Deploy as a Deployment Specification XML file.

`SCRIPT`

Deploy as a set of Oracle Script files.

`CONTROL_CENTER_ONLY`

Do not perform a Deployment but only update the Control Center Audit.

FROM

Perform the Deployment from a file on a File System.

ASYNCHRONOUS

Deploy in an asynchronous mode.

Examples

```
OMBDEPLOY DEPLOYMENT_ACTION_PLAN 'MY_DEPLOY_PLAN'
```

OMBDERIVE

Purpose

Derives business definition objects from OWB design objects.

Prerequisites

Should be in an appropriate context for the design object or use the full path for the design object.

The Business Definition Module named in the TO clause should already exist within the current project.

Any Business Areas named should already exist within the Business Definition Module.

Syntax

```

parseDeriveCommand = OMBDERIVE ( ORACLE_MODULE | DIMENSION | CUBE | TABLE
    | VIEW | EXTERNAL_TABLE | FUNCTION | COLLECTION ) "QUOTED_STRING" TO
    BUSINESS_DEFINITION_MODULE "QUOTED_STRING" [ BUSINESS_AREAS "("
        "BANameList" ")" ] [ "parseDeriveCommandParams" ]
BANameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
parseDeriveCommandParams = "parseRuleParameters" { "parseRuleParameters" }
parseRuleParameters = SET [ ( ORACLE_MODULE | DIMENSION | CUBE | TABLE |
    VIEW | EXTERNAL_TABLE | FUNCTION | COLLECTION ) ] "("
    "parameterNameList" ")" VALUES "(" "parameterValueList" ")"
parameterNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
parameterValueList = "parameterValue" { ", " "parameterValue" }
parameterValue = ( "QUOTED_STRING" )

```

Keywords And Parameters

`parseDeriveCommand`

Derives business definition objects from OWB design objects.

The quoted string following TO should be the name of a Business Definition Module in the current project, into which the business definition objects are derived.

Examples

```

OMBDERIVE TABLE 'MOD1/T1' TO BUSINESS_DEFINITION_MODULE 'IM1'
BUSINESS_AREAS ('BA1', 'BA2')

```

This sets up or updates business definition objects in Business Definition Module 'IM1' for table 'T1' within Oracle module 'MOD1', creating shortcuts in business areas BA1 and BA2.

```

OMBDERIVE ORACLE_MODULE 'MOD1' TO BUSINESS_DEFINITION_MODULE
'IM2'

```

This sets up or updates business definition objects in Business Definition

Module 'IM2' for the design objects within Oracle module 'MOD1'.

Note that deriving an Oracle Module does not derive the functions in that module

```
OMBDERIVE FUNCTION 'WB_CUSTOM_TRANS/ARITHMETIC_CUSTOM' TO
BUSINESS_DEFINITION_MODULE 'IM_ORDERENTRY'
```

This derives the custom function ARITHMETIC_CUSTOM. Note that all custom functions reside in module WB_CUSTOM_TRANS

```
OMBDERIVE FUNCTION 'WB_CUSTOM_TRANS/PACK1/CUSTOM_F1' TO
BUSINESS_DEFINITION_MODULE 'IM_ORDERENTRY'
```

This derives from the package PACK1 the custom function CUSTOM_F1.

```
OMBDERIVE COLLECTION 'C1' TO BUSINESS_DEFINITION_MODULE 'IM3'
```

This sets up or updates business definition objects in Business Definition Module 'IM3' for the design objects in collection 'C1'.

To modify the global and rule derivation parameters use the following -

```
OMBDERIVE COLLECTION 'C1' TO BUSINESS_DEFINITION_MODULE 'EUL1'
SET (LOGLEVEL, LOGFILE) VALUES ('INFO', '/tmp/logFile')
SET DIMENSION (BUILDLEVELFOLDERS) VALUES ('TRUE')
```

This sets up or updates business definition objects in Business Definition Module 'EUL1' for the design objects in collection 'C1' passing the value 'INFO' to global parameter LOGLEVEL, the value '/tmp/logFile' to the global parameter LOGFILE and the value 'TRUE' to the Dimension rule parameter BUILDLEVELFOLDERS.

Valid global derivation parameters are -

```
PRESERVEUSERCHANGES ('TRUE', 'FALSE')
LOGLEVEL ('ERROR', 'WARNING', 'INFO', 'TRACE')
LOGFILE - path and name identifying the log file location
VALIDATEBEFOREDERIVE ('TRUE', 'FALSE')
INITIALCAPITALS ('0', '1', '2')
```

Process business names as follows -

'0' leaves name as is,

'1' capitalizes first letter of all words in name,

'2' capitalizes first letter of first word only,
any other values are ignored
REPLACEUNDERScores ('TRUE', 'FALSE')
ABORTONERROR ('TRUE', 'FALSE')

Valid rule parameters for DIMENSION
FORCEBUILDDIMENSIONFOLDER ('TRUE', 'FALSE')
BUILDLEVELFOLDERS ('TRUE', 'FALSE')
DRILLPATHSONLEVELFOLDERS ('TRUE', 'FALSE')
PREFIXITEMS ('TRUE', 'FALSE')
PREFIXSEPARATOR - separator to follow prefix in business names
SORTFOLDERITEMS ('TRUE', 'FALSE')
BUILDDIMROLES ('TRUE', 'FALSE')

Valid rule parameters for CUBE
SORTFOLDERITEMS ('TRUE', 'FALSE')

Valid rule parameters for TABLE, VIEW and EXTERNAL_TABLE
BOUNDTABLESUFFIX - string to append to bound table names (default 'TAB')
DEFAULTAGGREGATE ('Detail', 'AVG', 'COUNT', 'MAX', 'MIN', 'SUM')
REMOVECOLUMNPREFIX ('TRUE', 'FALSE')
SORTFOLDERITEMS ('TRUE', 'FALSE')

OMBDESCRIBE ASSOCIATION_DEFINITION

Purpose

To describe an association definition.

Prerequisites

Association definition must be already exist. This command can be executed for any association definition regardless of current context.

Syntax

```
parseDescribeAssociationCommand = OMBDESCRIBE ASSOCIATION_DEFINITION
    "QUOTED_STRING" ( GET ( "getAssociationPropertiesClause" |
        DEPENDENCY_DEFINITIONS ) | ( DEPENDENCY_DEFINITION "QUOTED_STRING" GET
            "getDependencyPropertiesClause" ) )
getAssociationPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getDependencyPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "propertyNameClause" { "," "propertyNameClause" }
propertyNameClause = ( "UNQUOTED_STRING" )
```

Keywords And Parameters

`parseDescribeAssociationCommand`

Describe an association definition.

`getAssociationPropertiesClause`

Basic properties for ASSOCIATION_DEFINITION:

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: N/A

A descriptive text for this association.

Name: CLASS_1

Type: STRING(200)

Valid Values: N/A

Default: N/A

Class on one side of the association.

NAME: CLASS_2

Type: STRING(200)

Value Values: N/A

Default: N/A

Class on the other side of the association.

Name: ROLE_1

Type: STRING(200)

Valid Values: N/A

Default: N/A

Role on one side of the association.

Name: ROLE_2

Type: STRING(200)

Valid Values: N/A

Default: N/A

Role on the other side of the association.

Name: ROLE_1_MIN_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Minimum cardinality of role 1. Value can be positive integer.

Name: ROLE_1_MAX_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Maximum cardinality of role 1. Value can be positive integer or 'INFINITE'.

Name: ROLE_1_NAVIGABLE

Type: BOOLEAN

Valid Values: true, false

Default: From user defined class side default is true. From OWB class side default is false.

Navigability of role 1.

Name: ROLE_2_MIN_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Minimum cardinality of role 2. Value can be positive integer.

Name: ROLE_2_MAX_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Maximum cardinality of role 2. Value can be positive integer or 'INFINITE'.

Name: ROLE_2_NAVIGABLE

Type: BOOLEAN

Valid Values: true, false

Default: From user defined class side default is true. From OWB class side default is false.

Navigability of role 2.

getDependencyPropertiesClause

Basic dependency-related properties for this association:

Name: SOURCE_ROLE_ID

Type: STRING(200)

Valid Values: ROLE_1, ROLE_2

Default: If one of the ends is a OWB class, then that is the default source. If both ends are user defined classes, then the association is default bidirectional.

Identifies the role (end) of the association which serves as the source for the dependency flow.

Name: BIDIRECTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: If one of the ends is a OWB class, then default is false. If both ends are user defined classes, then default is true.

Specifies whether the association is bi-directional for the dependency flow.

propertyNameList

The list of property names.

propertyNameClause

Name of a property.

Examples

```
OMBDESCRIBE ASSOCIATION_DEFINITION 'UD_ASSOC4' GET PROPERTIES  
(CLASS_1,  
CLASS_2, ROLE_1, ROLE_2)
```

This would retrieve the names of the two classes participating in this association, as well as their role names.

See Also

OMBDEFINE ASSOCIATION_DEFINITION

OMBDESCRIBE CLASS_DEFINITION

Purpose

To describe a class definition or its property definitions.

Prerequisites

Class definition must be already exist. This command can be executed for any class definition regardless of current context.

Syntax

```
parseDescribeClassCommand = OMBDESCRIBE CLASS_DEFINITION "QUOTED_STRING" (  
    ( PROPERTY_DEFINITION "QUOTED_STRING" GET  
      "getPropertyDefinitionPropertiesClause" ) | ( GET ( [ USER_DEFINED |  
        CORE | LOGICAL | ( CONFIGURATION | PHYSICAL_CONFIGURATION ) ]  
        PROPERTY_DEFINITIONS ) ) | GET "getClassDefinitionPropertiesClause" |  
      GET CHILD_TYPES | GET ASSOCIATION_DEFINITIONS )  
    getPropertyDefinitionPropertiesClause = PROPERTIES "(" "propertyNameList"  
      ")"  
    getClassDefinitionPropertiesClause = PROPERTIES "(" "propertyNameList" ")"  
    propertyNameList = "propertyNameClause" { "," "propertyNameClause" }  
    propertyNameClause = ( "UNQUOTED_STRING" )
```

Keywords And Parameters

`parseDescribeClassCommand`
Describe a class definition.

`QUOTED_STRING`
Name of the class definition.

`PROPERTY_DEFINITIONS`
Get the list of property definitions for the class definition.

`propertyNameList`
List of the names of the properties.

`propertyNameClause`
Name of the property.

`UNQUOTED_STRING`
Name of the property.

Examples

```
OMBDESCRIBE CLASS_DEFINITION 'TABLE'
```


GET PROPERTY_DEFINITIONS

This will list all property definitions in this class definition.

OMBDESCRIBE CLASS_DEFINITION 'TABLE'

GET PROPERTIES (STEREOTYPE, IS_ABSTRACT, DESCRIPTION)

This will give the information about this class.

OMBDESCRIBE CLASS_DEFINITION 'TABLE' PROPERTY_DEFINITION 'tbl_udp'

GET PROPERTIES (TYPE, DEFAULT_VALUE, BUSINESS_NAME)

This will give the information about the property definition.

See Also

OMBDEFINE CLASS_DEFINITION, OMBREDEFINE CLASS_DEFINITION

OMBDESCRIBE MODEL

Purpose

To describe a model. A model is usually composed of types and relationships among them. The only model currently supported is

'OWB'. Using this command you can introspect 'OWB' model to list classes and other type definitions, association definitions and domain definitions.

Prerequisites

Must be connected to a repository.

Syntax

```
parseDescribeModelCommand = OMBDESCRIBE MODEL "QUOTED_STRING" GET (
    CLASS_DEFINITIONS | TYPE_DEFINITIONS | DOMAIN_DEFINITIONS |
    ASSOCIATION_DEFINITIONS | PRIMITIVE_TYPES )
```

Keywords And Parameters

parseDescribeModelCommand

Describe a model. The only model currently supported is 'OWB'.

Examples

```
OMBDESCRIBE MODEL 'OWB' GET CLASS_DEFINITIONS
```

This would retrieve the names of all classes in 'OWB' model, including the user defined ones.

```
OMBDESCRIBE MODEL 'OWB' GET ASSOCIATION_DEFINITIONS
```

This would retrieve the names of the user defined associations in 'OWB' model.

OMBDISC

Purpose

To disconnect from a OWB repository or the named Control Center.

Prerequisites

Currently need to be connected to a OWB repository or the named Control Center.

Syntax

```
parseDisconnectCommand = ( ( OMBDISCONNECT | OMBDISC ) [ CONTROL_CENTER ] )
```

Keywords And Parameters

parseDisconnectCommand
Specify disconnect command.

Examples

```
OMBDISCONNECT
```

See Also

```
OMBCONNECT
```

OMBDISCONNECT

Purpose

To disconnect from a OWB repository or the named Control Center.

Prerequisites

Currently need to be connected to a OWB repository or the named Control Center.

Syntax

```
parseDisconnectCommand = ( ( OMBDISCONNECT | OMBDISC ) [ CONTROL_CENTER ] )
```

Keywords And Parameters

`parseDisconnectCommand`
Specify disconnect command.

Examples

```
OMBDISCONNECT
```

See Also

```
OMBCONNECT
```

OMBDISCONNECT CONTROL_CENTER

Purpose

To disconnect from a OWB repository or the named Control Center.

Prerequisites

Currently need to be connected to a OWB repository or the named Control Center.

Syntax

```
parseDisconnectCommand = ( ( OMBDISCONNECT | OMBDISC ) [ CONTROL_CENTER ] )
```

Keywords And Parameters

parseDisconnectCommand
Specify disconnect command.

Examples

```
OMBDISCONNECT
```

See Also

```
OMBCONNECT
```

OMBDISC CONTROL_CENTER

Purpose

To disconnect from a OWB repository or the named Control Center.

Prerequisites

Currently need to be connected to a OWB repository or the named Control Center.

Syntax

```
parseDisconnectCommand = ( ( OMBDISCONNECT | OMBDISC ) [ CONTROL_CENTER ] )
```

Keywords And Parameters

parseDisconnectCommand
Specify disconnect command.

Examples

```
OMBDISCONNECT
```

See Also

```
OMBCONNECT
```

OMBDISPLAYCURRENTMODE

Purpose

Displays the current connection mode to the repository. Returns either SINGLE_USER_MODE (that is exclusive) or MULTIPLE_USER_MODE.

Prerequisites

Must be connected to an OWB repository. If not already connected, use OMBCONNECT first.

Syntax

```
parseDisplayModeCommand = ( OMBDISPLAYCURRENTMODE | OMBDCM )
```

Keywords And Parameters

parseDisplayModeCommand
Specify display mode command.

Examples

```
OMBDISPLAYCURRENTMODE
```

See Also

OMBSWITCHMODE, OMBCONNECT, OMBSAVE, OMBREVERT

OMBENV

Purpose

This command will list the values for all set OMBPlus environment variables. OMBPlus environment variables are regular Tcl variables, so they can be set using standard Tcl "set" command, and unset using standard "unset" command.

Prerequisites

None.

Syntax

```
parseEnvironmentCommand = OMBENV
```

Keywords And Parameters

parseEnvironmentCommand

Specify environment command. The environment variables recognized by OMBPlus are:

OMBPROMPT if set, will display the current context as the prompt

OMBLOG if set with a file name, will log the output to that file

OMBTIMER if set, will display the time taken by the executed command.

Note that the elapsed time will be appended to the command result, therefore the user should not set this variable when the result of the command is expected to be processed

OMBCONTINUE_ON_ERROR if set, will continue the execution of the script, even when some OMB commands fail. Note that if this variable is set, OMB commands no longer return Tcl errors, so any enclosing 'catch' command will not be effective.

Examples

```
OMBENV
```

OMBEXPORT

Purpose

Exports current metadata, metadata definitions, or snapshot metadata to a metadata loader file.

Prerequisites

Connection must be established to repository to be exported from.

Syntax

```
ExportCommand = OMBEXPORT ( ( [ TO ] MDL_FILE "QUOTED_STRING" [ FROM ] [
  "projectClause" ] [ "componentsClause" ] [ ( "classesClause" |
  ALL_CLASS_DEFINITIONS ) ] [ WITH ] [ "dependeeDepthClause" ] [
  "includeGrantsClause" ] [ "includeCMIDefsClause" ] [
  "useFieldSeparatorClause" ] [ "controlFileClause" ] [
  "includeUserDefsClause" ] [ "supportedLangsClause" ] [
  "outputLogClause" ] ) )
projectClause = PROJECT "QUOTED_STRING"
componentsClause = COMPONENTS "(" "componentsList" ")"
classesClause = CLASS_DEFINITIONS "(" "QUOTED_STRING" { ","
  "QUOTED_STRING" } ")"
dependeeDepthClause = DEPENDEE_DEPTH ( MAX | "INTEGER_LITERAL" )
includeGrantsClause = INCLUDE_GRANTS
includeCMIDefsClause = INCLUDE_CMI_DEFINITIONS
useFieldSeparatorClause = [ USE ] FIELD_SEPARATOR "UNQUOTED_STRING"
controlFileClause = CONTROL_FILE "QUOTED_STRING"
includeUserDefsClause = INCLUDE_USER_DEFINITIONS
supportedLangsClause = SUPPORTED_LANGUAGES "(" "supportedLangsList" ")"
outputLogClause = OUTPUT LOG [ TO ] "QUOTED_STRING"
componentsList = "newObjectTypeValue" "QUOTED_STRING" { ","
  "newObjectTypeValue" "QUOTED_STRING" }
supportedLangsList = "QUOTED_STRING" { "," "QUOTED_STRING" }
newObjectTypeValue = "UNQUOTED_STRING"
```

Keywords And Parameters

ExportCommand

Export metadata and optionally metadata definitions to a file.

QUOTED_STRING

Enclose the name of the export metadata file in single quotes.

projectClause

Specify the project to be exported.

QUOTED_STRING

Enclose the name of the project to be exported in single quotes.

componentsClause

List components to be exported.

dependeeDepthClause

Specify how many levels of dependent objects should be exported for components specified for export. Use MAX for all dependents, 0 for no dependents, 1 for one level of dependents. The default is 0.

includeGrantsClause

Use to request that security related metadata be included in the export.

includeCMIDefsClause

Use to request that related CMI Definitions for CMI Modules be included in the export.

useFieldSeparatorClause

Do not use this option. It is an obsolete option that was used to specify character to be used as the field separator in the old metadata file format.

UNQUOTED_STRING

Do not use this option. Obsolete metadata file field separator for old MDL file format. Use BAR or CARAT. The default is BAR.

controlFileClause

Specify a control file with export options not directly supported by OMBEXPORT command.

QUOTED_STRING

Enclose the control file name in single quotes.

includeUserDefsClause

Use to request that metadata definitions be included in the export.

supportedLangsClause

List supported languages to be exported.

outputLogClause

Export log file for export messages and statistics.

QUOTED_STRING

Enclose the log file name in single quotes.

componentsList

Comma separated list of components to be exported.

QUOTED_STRING

Enclose in single quotes the absolute or relative path name of an object (for example 'MODULE_X/TABLE_Y').

supportedLangsList

Comma separated list of supported languages to be exported.

QUOTED_STRING

Language name or ISO id of supported language (for example 'German' or 'de_DE').

Examples

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
OUTPUT
LOG TO 'd:/mdl/exp1.log'
```

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
WITH
DEPENDEE_DEPTH MAX OUTPUT LOG TO 'd:/mdl/exp1.log'
```

```
OMBEXPORT MDL_FILE 'd:/mdl/exp1.mdl' PROJECT 'MY_PROJECT'
COMPONENTS
(ORACLE_MODULE 'DW1', TABLE 'EMP', VIEW 'DEPT_VW') OUTPUT LOG
'd:/mdl/exp1.log'
```

```
OMBEXPORT MDL_FILE 'd:/mdl/snap1.mdl' COMPONENTS (SNAPSHOT
'MY_PROJECT_SNAP1') OUTPUT LOG 'd:/mdl/snap1.log'
```

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
INCLUDE_USER_DEFINITIONS OUTPUT LOG TO 'd:/mdl/exp1.log'
```

See Also

OMBIMPORT

OMBEXPORT MDL_FILE

Purpose

Exports current metadata, metadata definitions, or snapshot metadata to a metadata loader file.

Prerequisites

Connection must be established to repository to be exported from.

Syntax

```
ExportCommand = OMBEXPORT ( ( [ TO ] MDL_FILE "QUOTED_STRING" [ FROM ] [
  "projectClause" ] [ "componentsClause" ] [ ( "classesClause" |
  ALL_CLASS_DEFINITIONS ) ] [ WITH ] [ "dependeeDepthClause" ] [
  "includeGrantsClause" ] [ "includeCMIDefsClause" ] [
  "useFieldSeparatorClause" ] [ "controlFileClause" ] [
  "includeUserDefsClause" ] [ "supportedLangsClause" ] [
  "outputLogClause" ] ) )
projectClause = PROJECT "QUOTED_STRING"
componentsClause = COMPONENTS "(" "componentsList" ")"
classesClause = CLASS_DEFINITIONS "(" "QUOTED_STRING" { ","
  "QUOTED_STRING" } ")"
dependeeDepthClause = DEPENDEE_DEPTH ( MAX | "INTEGER_LITERAL" )
includeGrantsClause = INCLUDE_GRANTS
includeCMIDefsClause = INCLUDE_CMI_DEFINITIONS
useFieldSeparatorClause = [ USE ] FIELD_SEPARATOR "UNQUOTED_STRING"
controlFileClause = CONTROL_FILE "QUOTED_STRING"
includeUserDefsClause = INCLUDE_USER_DEFINITIONS
supportedLangsClause = SUPPORTED_LANGUAGES "(" "supportedLangsList" ")"
outputLogClause = OUTPUT LOG [ TO ] "QUOTED_STRING"
componentsList = "newObjectTypeValue" "QUOTED_STRING" { ","
  "newObjectTypeValue" "QUOTED_STRING" }
supportedLangsList = "QUOTED_STRING" { "," "QUOTED_STRING" }
newObjectTypeValue = "UNQUOTED_STRING"
```

Keywords And Parameters

ExportCommand

Export metadata and optionally metadata definitions to a file.

QUOTED_STRING

Enclose the name of the export metadata file in single quotes.

projectClause

Specify the project to be exported.

QUOTED_STRING

Enclose the name of the project to be exported in single quotes.

componentsClause

List components to be exported.

dependeeDepthClause

Specify how many levels of dependent objects should be exported for components specified for export. Use MAX for all dependents, 0 for no dependents, 1 for one level of dependents. The default is 0.

includeGrantsClause

Use to request that security related metadata be included in the export.

includeCMIDefsClause

Use to request that related CMI Definitions for CMI Modules be included in the export.

useFieldSeparatorClause

Do not use this option. It is an obsolete option that was used to specify character to be used as the field separator in the old metadata file format.

UNQUOTED_STRING

Do not use this option. Obsolete metadata file field separator for old MDL file format. Use BAR or CARAT. The default is BAR.

controlFileClause

Specify a control file with export options not directly supported by OMBEXPORT command.

QUOTED_STRING

Enclose the control file name in single quotes.

includeUserDefsClause

Use to request that metadata definitions be included in the export.

supportedLangsClause

List supported languages to be exported.

outputLogClause

Export log file for export messages and statistics.

QUOTED_STRING

Enclose the log file name in single quotes.

componentsList

Comma separated list of components to be exported.

QUOTED_STRING

Enclose in single quotes the absolute or relative path name of an object (for example 'MODULE_X/TABLE_Y').

supportedLangsList

Comma separated list of supported languages to be exported.

QUOTED_STRING

Language name or ISO id of supported language (for example 'German' or 'de_DE').

Examples

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
OUTPUT
LOG TO 'd:/mdl/exp1.log'
```

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
WITH
DEPENDEE_DEPTH MAX OUTPUT LOG TO 'd:/mdl/exp1.log'
```

```
OMBEXPORT MDL_FILE 'd:/mdl/exp1.mdl' PROJECT 'MY_PROJECT'
COMPONENTS
(ORACLE_MODULE 'DW1', TABLE 'EMP', VIEW 'DEPT_VW') OUTPUT LOG
'd:/mdl/exp1.log'
```

```
OMBEXPORT MDL_FILE 'd:/mdl/snap1.mdl' COMPONENTS (SNAPSHOT
'MY_PROJECT_SNAP1') OUTPUT LOG 'd:/mdl/snap1.log'
```

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
INCLUDE_USER_DEFINITIONS OUTPUT LOG TO 'd:/mdl/exp1.log'
```

See Also

OMBIMPORT MDL_FILE

OMBGRANT DEFAULT_OBJ_PRIV

Purpose

To change the default object privilege setting property of a user, basically it will grant a list of default object privileges to a list of users or roles. These object privileges will really be granted to the specified user or role on the new objects whenever the owner creates them.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseGrantDefObjPrivCommand = OMBGRANT DEFAULT ( OBJ_PRIV |
    OBJECT_PRIVILEGE ) "objPrivNameList" TO ( USER "userOrRoleNameList" |
    ROLE "userOrRoleNameList" ) [ FOR USER "QUOTED_STRING" ]
objPrivNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
userOrRoleNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
```

Keywords And Parameters

parseGrantDefObjPrivCommand

This clause change the default object privilege setting property of a user through granting a list of default object privileges to a list of users or roles.

objPrivNameList

Object privileges. Valid object privileges are: READ, COMPILE, EDIT, FULL_CONTROL.

Examples

```
OMBGRANT DEFAULT OBJ_PRIV EDIT TO ROLE 'DEVELOPMENT_ROLE'
the current login user grants default object privilege EDIT to role
DEVELOPMENT_ROLE, thus whenever the current login user creates an object,
the role DEVELOPMENT_ROLE will be granted EDIT privilege on that object.
```

```
OMBGRANT DEFAULT OBJ_PRIV EDIT TO ROLE 'DEVELOPMENT_ROLE' FOR
USER
'USER1'
```

the current user grants default object privilege EDIT to role DEVELOPMENT_ROLE on behalf of user USER1, thus whenever the user USER1 creates an object, the role DEVELOPMENT_ROLE will be granted EDIT privilege

on that object. Note if the current login user does not change the default object privilege setting for himself, the FOR USER statement is required.

See Also

OMBREVOKE DEFAULT_OBJ_PRIV, OMBLIST DEFAULT_OBJ_PRIVS

OMBGRANT OBJ_PRIV

Purpose

To grant a list of object privileges on the specified objects to a list of users or roles.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseGrantObjPrivCommand = OMBGRANT ( OBJ_PRIV | OBJECT_PRIVILEGE )
    "objPrivNameList" ON "UNQUOTED_STRING" "QUOTED_STRING" [ CASCADE ] TO
    ( USER "userOrRoleNameList" | ROLE "userOrRoleNameList" )
objPrivNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
userOrRoleNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
```

Keywords And Parameters

parseGrantObjPrivCommand

This clause grants a list of object privileges on the specified objects to a list of users or roles.

UNQUOTED_STRING

The type of the objects to be listed. Valid object types are: PROJECT,

ORACLE_MODULE, FLAT_FILE_MODULE, PROCESS_FLOW_MODULE, SAP_MODULE,

BUSINESS_DEFINITION_MODULE, BUSINESS_PRESENTATION_MODULE,

TRANSFORMATION_MODULE, PACKAGE, ADVANCED_QUEUE, BUSINESS_AREA, COLLECTION,

CONNECTOR, CUBE, DIMENSION, EXTERNAL_TABLE, FLAT_FILE, FUNCTION,

REGISTERED_FUNCTION, MAPPING, MATERIALIZED_VIEW, OBJECT_TYPE, PROCEDURE,

ITEM_FOLDER, DRILL_PATH, DRILLS_TO_DETAIL, ALTERNATIVE_SORT_ORDER,

LISTS_OF_VALUE, PRESENTATION_TEMPLATE, SEQUENCE, TABLE, PROCESS_FLOW,

PROCESS_FLOW_PACKAGE, LOCATION, CONTROL_CENTER, CONFIGURATION, DEPLOYMENT,

VIEW.

QUOTED_STRING

an object name or a regular expression refers to a list of objects whose name matches the regular expression.

CASCADE

grant the specified object privileges on a folder object and all its child objects cascade down. Note: if CASCADE is used, the string used to specify the object can only refer to one object and not be a regular expression .

objPrivNameList

Object privileges. Valid object privileges are: READ, COMPILE, EDIT, FULL_CONTROL.

Examples

```
OMBGRANT OBJ_PRIV EDIT ON ORACLE_MODULE '/MY_PROJECT/WH' TO  
USER 'USER1'
```

will grant EDIT on module WH to user USER1

```
OMBGRANT OBJ_PRIV EDIT ON PROJECT 'MY_PROJECT' CASCADE TO ROLE  
'EVERYONE'
```

will grant object privilege EDIT to role EVERYONE on project MY_PROJECT and all its child objects cascade.

```
OMBGRANT OBJ_PRIV READ ON TABLE '/MY_PROJECT/WH/EMP*' TO USER  
'USER1'
```

will grant READ to user USER1 on all tables under module /MY_PROJECT/WH whose names match regular expression EMP*.

See Also

OMBREVOKE OBJ_PRIV, OMBLIST OBJ_PRIVS

OMBGRANT ROLE

Purpose

To grant a list of Warehouse Builder roles to a list of Warehouse Builder users. Note role cannot be granted to another role.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseGrantRoleCommand = OMBGRANT ROLE "userOrRoleNameList" TO USER  
    "userOrRoleNameList"  
userOrRoleNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
```

Keywords And Parameters

parseGrantRoleCommand

This clause grants a list of Warehouse Builder roles to a list of Warehouse Builder users.

Examples

```
OMBGRANT ROLE 'DEV', 'QA' TO USER 'USER1' 'USER2'
```

will grant Warehouse Builder role 'DEV' and 'QA' to user 'USER1' and 'USER2'

See Also

OMBREVOKE ROLE, OMBLIST ROLES

OMBGRANT SYS_PRIV

Purpose

To grant a list of system privileges to a list of users or roles.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseGrantSysPrivCommand = OMBGRANT ( SYS_PRIV | SYSTEM_PRIVILEGE )
    "sysPrivNameList" TO ( USER "userOrRoleNameList" | ROLE
    "userOrRoleNameList" )
sysPrivNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
userOrRoleNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
```

Keywords And Parameters

`parseGrantSysPrivCommand`

This clause grants a list of system privileges to a list of users or roles.

`sysPrivNameList`

List of system privileges. Valid system privileges are:CREATE_PROJECT,
CREATE_MIVDEFINITION,CREATE_EXTENSIONMODEL, CONTROL_CENTER_
ADMIN,
CONTROL_CENTER_DEPLOY, CONTROL_CENTER_EXECUTE, CREATE_
SNAPSHOT

Examples

```
OMBGRANT SYS_PRIV 'CREATE_SNAPSHOT' TO USER 'USER1', 'USER2'
```

will grant system privilege CREATE_SNAPSHOT to user 'USER1' and 'USER2'

```
OMBGRANT SYS_PRIV 'CREATE_PROJECT' TO ROLE 'DEVELOPMENT_ROLE'
```

will grant system privilege 'CREATE_PROJECT' to role 'DEVELOPMENT_ROLE'.

See Also

OMBREVOKE SYS_PRIV, OMBLIST SYS_PRIVS

OMBHELP

Purpose

Displays the manual page for OMB commands.

Prerequisites

None.

Syntax

```
parseHelpCommand = OMBHELP [ HELPID ] [ DETAIL ]
```

Keywords And Parameters

parseHelpCommand

Specifies the command for which to invoke help.

HELPID

An unquoted string that represents either the command name and the optional additional parameter on the command.

DETAIL

Use this keyword to display the manual page in long format.

Examples

The following statement shows the manual page for OMBCREATE TABLE in short format

```
OMBHELP OMBCREATE TABLE
```

To show the manual page in long format (including parameters and example sections), use the DETAIL keyword, like this:

```
OMBHELP OMBCREATE TABLE DETAIL
```

OMBIMPACT

Purpose

To fetch the impact of a change for an object.

Prerequisites

In the context of an object's parent module.

Syntax

```
parseImpactCommand = OMBIMPACT DEPENDENCYTYPE "parseType" "parseFCO" [
    "parseSCO" ] [ "parseDetail" ]
parseType = "QUOTED_STRING"
parseFCO = ( ADVANCED_QUEUE | ALTERNATIVE_SORT_ORDER | CUBE | DATA_RULE |
    DIMENSION | DRILL_PATH | DRILL_TO_DETAIL | EXTERNAL_TABLE | FLAT_FILE
    | FUNCTION | ITEM_FOLDER | LIST_OF_VALUES | MATERIALIZED_VIEW |
    NESTED_TABLE | OBJECT_TYPE | PLUGGABLE_MAPPING | PRESENTATION_TEMPLATE
    | PROCEDURE | QUEUE_TABLE | REGISTERED_FUNCTION | SEQUENCE | TABLE |
    TIME_DIMENSION | VARYING_ARRAY | VIEW ) "QUOTED_STRING"
parseSCO = "UNQUOTED_STRING" "QUOTED_STRING" { "UNQUOTED_STRING"
    "QUOTED_STRING" }
parseDetail = GET MAPPING CONTENTS
```

Keywords And Parameters

parseImpactCommand

Specify OMBIMPACT command.

parseType

The type of impact. Allowable values are: DATAFLOW or DEPLOYMENT.

parseFCO

The object whose impact needs to be fetched.

parseSCO

The second class object whose impact needs to be fetched, specified in the form: SCO_type SCO_name. The second class object has to belong to the first class object specified by the previous clause. If the second class object is not direct child of the first class object, then the whole hierarchy path has to be listed (for example: parent_SCO_type parent_SCO_name child_SCO_type child_SCO_name).

parseDetail

Expands mappings and show operators involved in the impact.

Examples

This command will fetch the impact of a change on table 'EMA01_SRC':

```
OMBIMPACT DEPENDENCYTYPE 'DATAFLOW' TABLE 'EMA01_SRC'
```

The result of this command could be for example:

```
{{TABLE /MY_PROJECT/WH_EMA01/EMA01_SRC} {MAPPING  
/MY_PROJECT/WH_EMA01/EMA01} {TABLE /MY_PROJECT/WH_  
EMA01/EMA01_TGT}}.
```

The following command would fetch the impact of a column change:

```
OMBIMPACT DEPENDENCYTYPE 'DATAFLOW' TABLE 'EMA01_SRC' COLUMN  
'INTEREST'
```

Another example would fetch the impact of the field belonging to a record
in a file:

```
OMBIMPACT DEPENDENCYTYPE 'DATAFLOW' FLAT_FILE 'FILE_MULTI'  
RECORD 'REC1'  
FIELD 'DS'
```

OMBIMPORT

Purpose

Imports metadata from a Metadata Loader file, or from the default metadata location specified in the module.

Please refer to OMBHELP OMBIMPORT MDL_FILE or OMBHELP OMBIMPORT METADATA_LOCATION.

Prerequisites

Must be connected to the repository where the import is to be performed.

Syntax

```

ImportCommand = OMBIMPORT [ FROM ] ( "mdlImportCommand" |
    "oracleDBImportCommand" )
mdlImportCommand = ( MDL_FILE "QUOTED_STRING" [ "noUpgradeClause" ] [
    "includeUserDefsClause" ] [ "useModeClause" | "asSnapshotClause" ] [
    "matchByClause" ] [ "includeGrantsClause" ] [ "includeCMIDefsClause" ] [
    [ "controlFileClause" ] [ "supportedLangsClause" ] [
    "allowDiffBaseLangClause" ] [ "outputLogClause" ] )
oracleDBImportCommand = ( METADATA_LOCATION [ FOR ] IMPORT_ACTION_PLAN
    "QUOTED_STRING" [ "setImportOptions" ] )
noUpgradeClause = NO_UPGRADE
includeUserDefsClause = INCLUDE_USER_DEFINITIONS
useModeClause = USE "modeValue"
asSnapshotClause = AS SNAPSHOT "QUOTED_STRING"
matchByClause = MATCH_BY "matchByValue"
includeGrantsClause = INCLUDE_GRANTS
includeCMIDefsClause = INCLUDE_CMI_DEFINITIONS
controlFileClause = CONTROL_FILE "QUOTED_STRING"
supportedLangsClause = SUPPORTED_LANGUAGES "(" "supportedLangsList" ")"
allowDiffBaseLangClause = ALLOW_DIFFERENT_BASE_LANGUAGE
outputLogClause = OUTPUT LOG [ TO ] "QUOTED_STRING"
setImportOptions = SET OPTIONS "(" "optionNameList" ")" VALUES "("
    "optionValueList" ")"
modeValue = ( CREATE_MODE | REPLACE_MODE | UPDATE_MODE | MERGE_MODE |
    TRANSLATION_DIRECT_LOAD_MODE )
matchByValue = ( UNIVERSAL_IDENTIFIER | NAMES )
supportedLangsList = "QUOTED_STRING" { ",", "QUOTED_STRING" }
optionNameList = "UNQUOTED_STRING" { ",", "UNQUOTED_STRING" }
optionValueList = "optionValue" { ",", "optionValue" }
optionValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

ImportCommand

Import metadata from a Metadata Loader file, or from the default metadata location specified in the module.

mdlImportCommand

Import metadata from a file.

oracleDBImportCommand

To import from the default metadata location based on the import action plan.

QUOTED_STRING

The name of the transient action plan.

noUpgradeClause

Specify that MDL files that are not compatible with current repository version should not be automatically upgraded. The default is to automatically upgrade unless this clause is used.

includeUserDefsClause

Use to request that any metadata definitions in the MDL file be imported.

useModeClause

Specify import mode for importing metadata file.

asSnapshotClause

Specify a new snapshot name into which the contents of the MDL file are to be imported. Use this option to import old archive files as snapshots.

QUOTED_STRING

Enclose the new snapshot name in single quotes.

matchByClause

Specify whether import should search for already existing objects using universal identifiers or physical names.

includeGrantsClause

Use to request that security related metadata be imported.

includeCMIDefsClause

Use to request that related CMI Definitions for CMI Modules be imported.

controlFileClause

Specify a control file with import options not directly supported by the OMBIMPORT command.

QUOTED_STRING

Enclose the control file name in single quotes.

supportedLangsClause

List supported languages to be imported.

allowDiffBaseLangClause

Use to specify that MDL files with a different base language than that of the import target repository can be imported.

outputLogClause

Log file for import messages and statistics.

QUOTED_STRING

Enclose the log file name in single quotes.

setImportOptions

Currently no import options are available to be set.

modeValue

Import mode. Use CREATE_MODE, REPLACE_MODE, UPDATE_MODE, MERGE_MODE, or

TRANSLATION_DIRECT_LOAD_MODE. The default is CREATE_MODE.

CREATE_MODE: create new metadata only

REPLACE_MODE: replace existing objects only

UPDATE_MODE: replace existing objects and create new metadata

MERGE_MODE: merge existing objects and create new metadata

TRANSLATION_DIRECT_LOAD_MODE: import language translation metadata only

matchByValue

Use UNIVERSAL_IDENTIFIER or NAMES. The default is UNIVERSAL_IDENTIFIER.

supportedLangsList

Comma separated list of supported languages to be imported.

QUOTED_STRING

Language name or ISO id of supported language (for example 'German' or 'de_DE').

Examples

```
OMBIMPORT FROM MDL_FILE 'd:/mdl/exp1.mdl' OUTPUT LOG TO  
'd:/mdl/exp1_imp.log'
```

```
OMBIMPORT FROM METADATA_LOCATION FOR IMPORT_ACTION_PLAN  
'PLAN1'
```

See Also

OMBIMPORT MDL_FILE, OMBIMPORT METADATA_LOCATION

OMBIMPORT MDL_FILE

Purpose

Imports metadata and metadata definitions from a Metadata Loader file.

Prerequisites

Must be connected to the repository where the import is to be performed.

Syntax

```
mdlImportCommand = ( MDL_FILE "QUOTED_STRING" [ "noUpgradeClause" ] [
    "includeUserDefsClause" ] [ "useModeClause" | "asSnapshotClause" ] [
    "matchByClause" ] [ "includeGrantsClause" ] [ "includeCMIDefsClause" ]
    [ "controlFileClause" ] [ "supportedLangsClause" ] [
    "allowDiffBaseLangClause" ] [ "outputLogClause" ] )
noUpgradeClause = NO_UPGRADE
includeUserDefsClause = INCLUDE_USER_DEFINITIONS
useModeClause = USE "modeValue"
asSnapshotClause = AS SNAPSHOT "QUOTED_STRING"
matchByClause = MATCH_BY "matchByValue"
includeGrantsClause = INCLUDE_GRANTS
includeCMIDefsClause = INCLUDE_CMI_DEFINITIONS
controlFileClause = CONTROL_FILE "QUOTED_STRING"
supportedLangsClause = SUPPORTED_LANGUAGES "(" "supportedLangsList" ")"
allowDiffBaseLangClause = ALLOW_DIFFERENT_BASE_LANGUAGE
outputLogClause = OUTPUT LOG [ TO ] "QUOTED_STRING"
modeValue = ( CREATE_MODE | REPLACE_MODE | UPDATE_MODE | MERGE_MODE |
    TRANSLATION_DIRECT_LOAD_MODE )
matchByValue = ( UNIVERSAL_IDENTIFIER | NAMES )
supportedLangsList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
```

Keywords And Parameters

mdlImportCommand

Import metadata from a file.

noUpgradeClause

Specify that MDL files that are not compatible with current repository version should not be automatically upgraded. The default is to automatically upgrade unless this clause is used.

includeUserDefsClause

Use to request that any metadata definitions in the MDL file be imported.

useModeClause

Specify import mode for importing metadata file.

asSnapshotClause

Specify a new snapshot name into which the contents of the MDL file are to be imported. Use this option to import old archive files as snapshots.

QUOTED_STRING

Enclose the new snapshot name in single quotes.

matchByClause

Specify whether import should search for already existing objects using universal identifiers or physical names.

includeGrantsClause

Use to request that security related metadata be imported.

includeCMIDefsClause

Use to request that related CMI Definitions for CMI Modules be imported.

controlFileClause

Specify a control file with import options not directly supported by the OMBIMPORT command.

QUOTED_STRING

Enclose the control file name in single quotes.

supportedLangsClause

List supported languages to be imported.

allowDiffBaseLangClause

Use to specify that MDL files with a different base language than that of the import target repository can be imported.

outputLogClause

Log file for import messages and statistics.

QUOTED_STRING

Enclose the log file name in single quotes.

modeValue

Import mode. Use CREATE_MODE, REPLACE_MODE, UPDATE_MODE, MERGE_MODE, or

TRANSLATION_DIRECT_LOAD_MODE. The default is CREATE_MODE.

CREATE_MODE: create new metadata only

REPLACE_MODE: replace existing objects only

UPDATE_MODE: replace existing objects and create new metadata

MERGE_MODE: merge existing objects and create new metadata

TRANSLATION_DIRECT_LOAD_MODE: import language translation metadata only

matchByValue

Use UNIVERSAL_IDENTIFIER or NAMES. The default is UNIVERSAL_IDENTIFIER.

supportedLangsList

Comma separated list of supported languages to be imported.

QUOTED_STRING

Language name or ISO id of supported language (for example 'German' or 'de_DE').

Examples

```
OMBIMPORT FROM MDL_FILE 'd:/mdl/exp1.mdl' OUTPUT LOG TO  
'd:/mdl/exp1_imp.log'
```

```
OMBIMPORT MDL_FILE 'd:/mdl/exp1.mdl' USE UPDATE_MODE OUTPUT LOG  
TO  
'd:/mdl/exp1_imp.log'
```

See Also

OMBEXPORT MDL_FILE

OMBIMPORT METADATA_LOCATION

Purpose

Import metadata from the default metadata location specified in the module. Currently we only support

tables/views/sequences for Oracle/Gateway/SAP/MIV modules.

Prerequisites

Must be connected to the repository where the import is to be performed.

Syntax

```
oracleDBImportCommand = ( METADATA_LOCATION [ FOR ] IMPORT_ACTION_PLAN
    "QUOTED_STRING" [ "setImportOptions" ] )
setImportOptions = SET OPTIONS "(" "optionNameList" ")" VALUES "("
    "optionValueList" ")"
optionNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
optionValueList = "optionValue" { "," "optionValue" }
optionValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

oracleDBImportCommand

To import from the default metadata location based on the import action plan.

QUOTED_STRING

The name of the transient action plan.

setImportOptions

Currently no import options are available to be set.

Examples

```
OMBIMPORT FROM METADATA_LOCATION FOR IMPORT_ACTION_PLAN
'PLAN1'
```

This command will carry out the import actions specified in import action plan PLAN1.

See Also

OMBCREATE TRANSIENT IMPORT_ACTION_PLAN

OMBINSTALL OWB_RAC

Purpose

Register RAC instance (only for RAC 's second instance onwards).

Prerequisites

This command should be run locally on the database server.

Syntax

```
parseRegisterRACCommand = OMBINSTALL OWB_RAC [ OWBRT_SYS_PASSWORD  
    "QUOTED_STRING" ] USING CREDENTIAL "UNQUOTED_STRING" [  
    NET_SERVICE_NAME "QUOTED_STRING" ]
```

Keywords And Parameters

parseRegisterRACCommand

Specify an OMBINSTALL OWB_RAC command.

UNQUOTED_STRING

Specify the connection string to the database using this format:

username/password@host:port:service name OR specify the user name and password using this format username/password.

Examples

1. OMBINSTALL OWB_RAC USING CREDENTIAL sys/sys@localhost:1521:orcl92

This registers a RAC instance.

OMBINSTALL OWB_REPOSITORY

Purpose

Install OWB repository.

Prerequisites

To install OWB repository, within the same OMBPlus session, no other OMB commands should be issued before or after the OMBINSTALL OWB_REPOSITORY command. The value of OWB_SERVER_HOME_CHECK_FLAG can be set to either TRUE

or FALSE. If OWB_SERVER_HOME_CHECK_FLAG is set to TRUE,

OWB_SERVER_ORACLE_HOME must be provided as a valid path which OWB software

is installed.

Syntax

```
parseInstallOWBRepositoryCommand = OMBINSTALL OWB_REPOSITORY
  "UNQUOTED_STRING" [ NAS_PORT "QUOTED_STRING" ] [ DATA_TABLESPACE
  "QUOTED_STRING" ] [ INDEX_TABLESPACE "QUOTED_STRING" ] [
  TEMPORARY_TABLESPACE "QUOTED_STRING" ] [ SNAPSHOT_TABLESPACE
  "QUOTED_STRING" ] [ BASE_LANGUAGE "QUOTED_STRING" ] [
  SUPPORTED_LANGUAGES "(" "QUOTED_STRING" { "," "QUOTED_STRING" } ")" ]
  [ OWB_SERVER_ORACLE_HOME "QUOTED_STRING" ] [
  OWB_SERVER_HOME_CHECK_FLAG "UNQUOTED_STRING" ] [ OWBRT_SYS_PASSWORD
  "QUOTED_STRING" ] USING CREDENTIAL "UNQUOTED_STRING" [
  NET_SERVICE_NAME "QUOTED_STRING" ]
```

Keywords And Parameters

parseInstallOWBRepositoryCommand

Specify an OMBINSTALL OWB_REPOSITORY command.

UNQUOTED_STRING

Specify the connection string to the database using this format:

username/password@host:port:service name, OR specify the user name and password pair using this format: username/password.

QUOTED_STRING

Specify the tablespace name, NAS port number or language name.

Examples

```
1. OMBINSTALL OWB_REPOSITORY h_rep1/h NAS_PORT '4040' DATA_
TABLESPACE
'USERS' INDEX_TABLESPACE 'INDX' TEMPORARY_TABLESPACE 'TEMP'
```

```
SNAPSHOT_TABLESPACE 'USERS' BASE_LANGUAGE 'AMERICAN ENGLISH'  
SUPPORTED_LANGUAGES ('ARABIC', 'ASSAMESE', 'CANADIAN FRENCH',  
'SIMPLIFIED  
CHINESE') USING CREDENTIAL sys/sys@localhost:1521:orcl92
```

This creates an OWB repository using the nondefault NAS port, nondefault tablespaces and nondefault supporting languages.

```
2. OMBINSTALL OWB_REPOSITORY h_rep2/h NAS_PORT '4040' DATA_  
TABLESPACE  
'USERS' INDEX_TABLESPACE 'INDX' TEMPORARY_TABLESPACE 'TEMP'  
SNAPSHOT_TABLESPACE 'USERS' USING CREDENTIAL  
sys/sys@localhost:1521:orcl92
```

This creates an OWB repository using the nondefault NAS port, nondefault tablespaces and default supporting languages.

```
3. OMBINSTALL OWB_REPOSITORY h_rep3/h NAS_PORT '4040' USING  
CREDENTIAL  
sys/sys@localhost:1521:orcl92
```

This creates an OWB repository using the nondefault NAS port, default tablespaces and default supporting languages.

```
4. OMBINSTALL OWB_REPOSITORY h_rep4/h USING CREDENTIAL  
sys/sys@localhost:1521:orcl92
```

This creates an OWB repository using the default NAS port, default tablespaces and default supporting languages.

OMBINSTALL OWB_TARGET_USER

Purpose

Create or install an OWB target user.

Prerequisites

OWB REPOSITORY must exist on the database server.

Syntax

```
parseInstallTargetUserCommand = OMBINSTALL OWB_TARGET_USER
    "UNQUOTED_STRING" [ DATA_TABLESPACE "QUOTED_STRING" ] [
    INDEX_TABLESPACE "QUOTED_STRING" ] [ TEMPORARY_TABLESPACE
    "QUOTED_STRING" ] USING OWB_REPOSITORY "UNQUOTED_STRING" USING
    CREDENTIAL "UNQUOTED_STRING" [ NET_SERVICE_NAME "QUOTED_STRING" ]
```

Keywords And Parameters

`parseInstallTargetUserCommand`

Specify an OMBINSTALL OWB_TARGET_USER command.

`UNQUOTED_STRING`

Specify the connection string to the database using this format:

username/password@host:port:service name OR specify the user name and password using this format username/password.

`QUOTED_STRING`

Specify the tablespace name.

Examples

1. OMBINSTALL OWB_TARGET_USER h_tu2/h USING OWB_REPOSITORY h_rep1/h USING

CREDENTIAL sys/sys@localhost:1521:orcl92

This creates a target user using the default tablespaces.

2. OMBINSTALL OWB_TARGET_USER h_tu1/h DATA_TABLESPACE 'USERS'

INDEX_TABLESPACE 'INDX' TEMPORARY_TABLESPACE 'TEMP' USING OWB_REPOSITORY

h_rep1/h USING CREDENTIAL sys/sys@localhost:1521:orcl92

This creates a target user using the nondefault tablespaces.

OMBLINEAGE

Purpose

To fetch the data lineage.

Prerequisites

In the context of an object's parent module.

Syntax

```
parseLineageCommand = OMBLINEAGE DEPENDENCYTYPE "parseType" "parseFCO" [  
    "parseSCO" ] [ "parseDetail" ]  
parseType = "QUOTED_STRING"  
parseFCO = ( ADVANCED_QUEUE | ALTERNATIVE_SORT_ORDER | CUBE | DATA_RULE |  
    DIMENSION | DRILL_PATH | DRILL_TO_DETAIL | EXTERNAL_TABLE | FLAT_FILE  
    | FUNCTION | ITEM_FOLDER | LIST_OF_VALUES | MATERIALIZED_VIEW |  
    NESTED_TABLE | OBJECT_TYPE | PLUGGABLE_MAPPING | PRESENTATION_TEMPLATE  
    | PROCEDURE | QUEUE_TABLE | REGISTERED_FUNCTION | SEQUENCE | TABLE |  
    TIME_DIMENSION | VARYING_ARRAY | VIEW ) "QUOTED_STRING"  
parseSCO = "UNQUOTED_STRING" "QUOTED_STRING" { "UNQUOTED_STRING"  
    "QUOTED_STRING" }  
parseDetail = GET MAPPING CONTENTS
```

Keywords And Parameters

parseLineageCommand

Specify OMBLINEAGE command.

parseType

The type of lineage. Allowable values are: DATAFLOW or DEPLOYMENT.

parseFCO

The object whose lineage needs to be fetched.

parseSCO

The second class object whose lineage needs to be fetched, specified in the form: SCO_type SCO_name. The second class object has to belong to the first class object specified by the previous clause. If the second class object is not direct child of the first class object, then the whole hierarchy path has to be listed (for example: parent_SCO_type parent_SCO_name child_SCO_type child_SCO_name).

parseDetail

Expands mappings and show operators involved in the lineage.

Examples

This command will fetch the lineage of table 'EMA01_TGT':

```
OMBLINEAGE DEPENDENCYTYPE 'DATAFLOW' TABLE 'EMA01_TGT'
```

The result of this command could be for example:

```
{{TABLE /MY_PROJECT/WH_EMA01/EMA01_SRC} {MAPPING  
/MY_PROJECT/WH_EMA01/EMA01} {TABLE /MY_PROJECT/WH_  
EMA01/EMA01_TGT}}  
  
{{CONFIGURATION /MY_PROJECT/MY_PROJECT/DEFAULT_CONFIGURATION}  
{TABLE  
/MY_PROJECT/WH_EMA01/EMA01_TGT}}
```

The following command would fetch the lineage of a column:

```
OMBLINEAGE DEPENDENCYTYPE 'DATAFLOW' TABLE 'EMA01_TGT' COLUMN  
'TOTAL'
```

Another example would fetch the lineage of a level attribute belonging to a level in a dimension:

```
OMBLINEAGE DEPENDENCYTYPE 'DATAFLOW' DIMENSION 'GEOGRAPHY'  
LEVEL 'CITY'  
LEVEL_ATTRIBUTE 'CTY_KEY'
```

OMBLIST

Purpose

This command lists the specified Warehouse Builder objects under the current folder. You can also provide a regular expression to list objects in a different folder, or to list objects with names matching a pattern, or both. Before displaying the objects, this command first refreshes the object list from the repository. This means that the list of objects returned also reflects any additions or deletions by other users.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseListCommand = "listActionPlans" | "listGeneral"  
listActionPlans = OMBLIST DEPLOYMENT_ACTION_PLANS [ "QUOTED_STRING" ]  
listGeneral = OMBLIST "UNQUOTED_STRING" [ "QUOTED_STRING" ]
```

Keywords And Parameters

`parseListCommand`

Specify the list command.

`listActionPlans`

Deployment action plans are listed.

`listGeneral`

OWB objects of the specified type are listed.

`UNQUOTED_STRING`

The type (plural) of the objects to be listed. Valid object types are:

ACTIVITY_TEMPLATES, ACTIVITY_TEMPLATE_FOLDERS, ADVANCED_QUEUES,

ALTERNATIVE_SORT_ORDERS, BUSINESS_AREAS, BUSINESS_DEFINITION_MODULES,

BUSINESS_PRESENTATION_MODULES, CALENDARS, CALENDAR_MODULES,

CMI_DEFINITIONS, CMI_MODULES, COLLECTIONS, CONFIGURATIONS, CONNECTORS,

CONTROL_CENTERS, CUBES, DATA_AUDITORS, DATA_PROFILES, DATA_RULES,

DATA_RULE_MODULES, DEPLOYMENTS, DEPLOYMENT_ACTION_PLANS, DIMENSIONS,

DRILL_PATHS, DRILLS_TO_DETAIL, EXPERTS, EXPERT_MODULES, EXTERNAL_
 TABLES,
 FLAT_FILES, FLAT_FILE_MODULES, FUNCTIONS, GATEWAY_MODULES,
 ICONSETS,
 ITEM_FOLDERS, LISTS_OF_VALUES, LOCATIONS, MAPPINGS, MATERIALIZED_
 VIEWS,
 NESTED_TABLES, OBJECT_TYPES, ORACLE_MODULES, PACKAGES, PLSQL_
 RECORD_TYPES,
 PLSQL_REF_CURSOR_TYPES, PLSQL_TABLE_TYPES, PLUGGABLE_MAPPINGS,
 PLUGGABLE_MAPPING_FOLDERS, PRESENTATION_TEMPLATES,
 PROCEDURES,
 PROCESS_FLOWS, PROCESS_FLOW_MODULES, PROCESS_FLOW_PACKAGES,
 PROJECTS,
 QUEUE_PROPAGATIONS, QUEUE_TABLES, REAL_TIME_MAPPINGS,
 REGISTERED_FUNCTIONS,
 ROLES, SAP_MODULES, SEQUENCES, SNAPSHOTS, STREAMS_CAPTURE_
 PROCESSES,
 STREAMS_QUEUES, TABLES, TABLE_FUNCTIONS, TIME_DIMENSIONS,
 TRANSFORMATION_MODULES, TRANSPORTABLE_MODULES, USERS,
 VARYING_ARRAYS,
 VIEWS.

QUOTED_STRING

Mandatory for CONNECTORS, specify a location name; optionally for all other objects, specify a regular expression, which is used to filter the results.

The syntax of the regular expressions follow the syntax from TCL.

Here are some of the quantifiers:

* -- represents a sequence of 0 or more matches of the atom.

+ -- represents a sequence of 1 or more matches of the atom.

? -- represents a sequence of 0 or 1 matches of the atom.

Here are some of the atoms:

. -- matches any single character

k -- (where k is a non-alphanumeric character) matches that character taken as an ordinary character,

for example `\\` matches a backslash character.

c -- where c is alphanumeric(possibly followed by other characters), an escape.

(re) -- (where re is any regular expression) matches a match for re, with the match noted for possible reporting.

[char] -- a bracket expression, matching any one of the chars.

Further documentation on the syntax can be found at the TCL manual page under the command 'regexp'.

Examples

OMBLIST TABLES

will list the tables under the current Oracle module folder context (if the current context is not an Oracle module, an error will be produced).

OMBLIST ORACLE_MODULES '/MY_PROJECT/O.*'

will list Oracle modules starting with letter 'O', within project 'MY_PROJECT'.

OMBLIST ORACLE_MODULES '/MY_PROJECT/[ABC\].*'

will list Oracle modules starting with either letter A or B or C, within project 'MY_PROJECT'.

OMBLIST LOCATIONS

will list all created locations within the repository.

OMBLIST CONNECTORS 'MY_LOCATION'

will list all connectors of the 'MY_LOCATION' location.

OMBLIST DEFAULT_OBJ_PRIVS

Purpose

To list the default object privileges granted to a specified role or user from the owner. These object privileges will really be granted to the specified user or role on the new object whenever the owner creates it.

Prerequisites

The default object privileges can be listed from any context.

Syntax

```
parseListDefObjPrivs = OMBLIST DEFAULT ( OBJ_PRIVS | OBJECT_PRIVILEGES ) (
    OF ( USER | ROLE ) "QUOTED_STRING" ) [ FOR USER "QUOTED_STRING" ]
```

Keywords And Parameters

`parseListDefObjPrivs`

This clause will show the default object privilege setting property of a user, basically will list the default object privileges granted to a specified role or user from the owner. The default object privileges will be really ganted to the specified user or role on the newly created object by the owner.

Examples

```
OMBLIST DEFAULT OBJ_PRIVS OF USER 'USER1'
```

will list the default object privileges granted to user 'USER1' from the owner, by default who is current login user.

```
OMBLIST DEFAULT OBJ_PRIVS OF ROLE 'EVERYONE' FOR USER 'USER2'
```

will list the default object privileges granted to role 'EVERYONE' from owner 'USER2'. Note if the current user want to show other user's default object privilege setting property, the FOR USER statement is required.

OMBLIST ICONSETS

Purpose

To list the icon sets in the current repository.

Prerequisites

Must be connected to a repository, in any context.

Syntax

```
listIconSetCommand = OMBLIST ICONSETS [ "QUOTED_STRING" ] [ OF  
"QUOTED_STRING" ]
```

Keywords And Parameters

`listIconSetCommand`

This command lists the iconsets in a repository.

`QUOTED_STRING`

A regular expression that describes the name of the icon set. Or the name of the group to which iconsets belong.

Examples

```
OMBLIST ICONSETS 'ICON.*'
```

See Also

OMBCREATE ICONSET

OMBLIST OBJ_PRIVS

Purpose

To list the object privileges granted to a specified user on a specified object.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseListObjPrivs = OMBLIST ( OBJ_PRIVS | OBJECT_PRIVILEGES ) ON
    "UNQUOTED_STRING" "QUOTED_STRING" ( OF ( USER | ROLE ) "QUOTED_STRING"
    )
```

Keywords And Parameters

parseListObjPrivs

This clause lists the object privileges granted to a specified user on a specified object.

UNQUOTED_STRING

The type of the objects to be listed. Valid object types are: PROJECT,

ORACLE_MODULE, FLAT_FILE_MODULE, PROCESS_FLOW_MODULE, SAP_MODULE,

BUSINESS_DEFINITION_MODULE, BUSINESS_PRESENTATION_MODULE,

TRANSFORMATION_MODULE, PACKAGE, ADVANCED_QUEUE, BUSINESS_AREA, COLLECTION,

CONNECTOR, CUBE, DIMENSION, EXTERNAL_TABLE, FLAT_FILE, FUNCTION,

REGISTERED_FUNCTION, MAPPING, MATERIALIZED_VIEW, OBJECT_TYPE, PROCEDURE,

ITEM_FOLDER, DRILL_PATH, DRILLS_TO_DETAIL, ALTERNATIVE_SORT_ORDER,

LISTS_OF_VALUE, PRESENTATION_TEMPLATE, SEQUENCE, TABLE, PROCESS_FLOW,

PROCESS_FLOW_PACKAGE, LOCATION, CONTROL_CENTER, CONFIGURATION, DEPLOYMENT,

VIEW.

Examples

```
OMBLIST OBJ_PRIVS ON TABLE 'EMP' OF USER 'USER1'
```

will list object privileges granted to user 'USER1' on table object 'EMP'.

You must be under an oracle module context to succeed.

OMBLIST OBJECT_PRIVILEGES ON TABLE '/MY_PROJECT/MY_MODULE/EMP'
OF ROLE

'EVERYONE' /n will list object privileges granted to role 'EVERYONE' on
table 'EMP'. Note that you can reference an object by using a full path
name.

OMBLIST SNAPSHOT

Purpose

This command lists all the snapshots existing or all snapshots for a specific component.

Prerequisites

Snapshots can be listed from any context.

Syntax

```
parseListCommand = OMBLIST "listSnapshotCommand"  
listSnapshotCommand = SNAPSHOTS [ FOR ( "UNQUOTED_STRING" "QUOTED_STRING"  
    ) ]
```

Keywords And Parameters

parseListCommand

Root production of OMBLIST SNAPSHOT.

listSnapshotCommand

To list existing snapshots.

SNAPSHOTS

Lists all the snapshots existing in the repository.

FOR

Lists all the snapshots existing for a specific component or folder.

Examples

OMBLIST SNAPSHOTS

This command lists all the snapshots ever taken in the lifetime of the repository.

OMBLIST SNAPSHOTS FOR TABLE '/Project/WH/T1'

This command lists all the snapshots for table T1. It can be seen as a version tree of table T1.

See Also

OMBCREATE SNAPSHOT, OMBALTER SNAPSHOT, OMBDROP SNAPSHOT, OMBRESTORE SNAPSHOT, OMBCOMPARE SNAPSHOT, OMBRETRIEVE SNAPSHOT

OMBLIST SYS_PRIVS

Purpose

To list the system privileges granted to a user or a role.

Prerequisites

System privileges granted to a user or a role can be listed from any context.

Syntax

```
parseListSysPrivs = OMBLIST ( SYS_PRIVS | SYSTEM_PRIVILEGES ) ( OF ( USER  
| ROLE ) "QUOTED_STRING" )
```

Keywords And Parameters

parseListSysPrivs

This clause lists the system privileges granted to a user or a role.

Examples

```
OMBLIST SYS_PRIVS OF USER 'USER1'
```

will list system privileges granted to user USER1.

```
OMBLIST SYSTEM_PRIVILEGES OF ROLE 'DEVELOPMENT_ROLE'
```

will list system privileges granted to role 'DEVELOPMENT_ROLE'.

OMBLIST USERS

Purpose

To list existing roles or list existing roles whose names match the given regular expression or list the granted roles of a given user.

Prerequisites

Warehouse Builder users can be list from any context.

Syntax

```
parseListUser = OMBLIST USERS ( [ "QUOTED_STRING" | ( OF ROLE  
"QUOTED_STRING" ) ] )
```

Keywords And Parameters

parseListUser

This clause lists existing Warehouse Builder users or list existing users matches the given regular expression or list granted users of a given role.

Examples

```
OMBLIST USERS
```

will list all existing warehouse builder users.

```
OMBLIST USERS 'USER*'
```

will list all users whose names match regular express 'USER*'.

```
OMBLIST USERS OF ROLE 'DEVELOPMENT_ROLE'
```

will list the users who have been granted the role 'DEVELOPMENT_ROLE'.

OMBLOCK

Purpose

Lock one or more objects. If a list of objects is specified, the objects will be locked one by one in the given order. The command will behave atomically, that is, if it fails to lock any of the objects in the list, then it will lock none of them.

Prerequisites

No other user should have a lock on any of the objects.

Syntax

```
parseLockCommand = OMBLOCK "parseTypeNameList"
parseTypeNameList = "objectType" "QUOTED_STRING" { "," "objectType"
"QUOTED_STRING" }
objectType = ( ADVANCED_QUEUE | STREAMS_QUEUE | BUSINESS_AREA | COLLECTION
| CONNECTOR | CONFIGURATION | DEPLOYMENT | CUBE | DIMENSION | EXPERT
| EXPERT_MODULE | EXTERNAL_TABLE | FLAT_FILE_MODULE | FLAT_FILE |
FUNCTION | TABLE_FUNCTION | GATEWAY_MODULE |
BUSINESS_DEFINITION_MODULE | REGISTERED_FUNCTION | LOCATION | MAPPING
| MATERIALIZED_VIEW | OBJECT_TYPE | VARYING_ARRAY | NESTED_TABLE |
ORACLE_MODULE | PACKAGE | PROCEDURE | PROCESS_FLOW |
PROCESS_FLOW_MODULE | PROCESS_FLOW_PACKAGE | PROJECT | ITEM_FOLDER |
DRILL_PATH | LIST_OF_VALUES | DRILL_TO_DETAIL | ALTERNATIVE_SORT_ORDER
| PRESENTATION_TEMPLATE | BUSINESS_PRESENTATION_MODULE |
CONTROL_CENTER | SAP_MODULE | CMI_MODULE | SEQUENCE | TABLE | VIEW |
CMI_DEFINITION | PLSQL_RECORD_TYPE | "UNQUOTED_STRING" )
```

Keywords And Parameters

`parseLockCommand`

Specify lock command.

`parseTypeNameList`

Specify the object or the list of objects to be locked.

`QUOTED_STRING`

Name of the object to be locked. Can be specified as an absolute path or as a path relative to the current context. However, there is the restriction that all objects to be locked must be in the current project.

`objectType`

Type of the object to be locked.

Examples

```
OMBLOCK TABLE 'T1', VIEW '/MY_PROJECT/ORACLE_1/V1'
```

will lock table 'T1' in the current module, and view 'V1' in Oracle module 'ORACLE_1' from project 'MY_PROJECT'.

See Also

OMBUNLOCK

OMBMLSUPDATE OWB_REPOSITORY

Purpose

Add more languages into an existing OWB repository to support the object translation.

Prerequisites

OWB repository should already exist in the database.

Syntax

```
parseUpdateMlsOWBRepositoryCommand = OMBMLSUPDATE OWB_REPOSITORY
    "UNQUOTED_STRING" [ SUPPORTED_LANGUAGES "(" "QUOTED_STRING" { ","
    "QUOTED_STRING" } ")" ] USING CREDENTIAL "UNQUOTED_STRING" [
    NET_SERVICE_NAME "QUOTED_STRING" ]
```

Keywords And Parameters

`parseUpdateMlsOWBRepositoryCommand`

Specify an OMBMLSUPDATE OWB_REPOSITORY command.

`UNQUOTED_STRING`

Specify the connection string to the database using this format:

username/password@host:port:service name, OR specify the user name and password pair using this format: username/password.

`QUOTED_STRING`

Specify the language name.

Examples

```
1. OMBMLSUPDATE OWB_REPOSITORY h_rep4/h SUPPORTED_LANGUAGES (
    'FRENCH',
```

```
'Mexican Spanish' ) USING CREDENTIAL sys/sys@localhost:1521:orcl92
```

This adds two languages 'FRENCH' and 'Mexican Spanish' into an OWB repository.

OMBMOVE

Purpose

Move one or more objects of the same object type. The replace option enables you to overwrite.

Prerequisites

Use of relative path specifications requires awareness of the current context.

Syntax

```
parseMoveCommand = OMBMOVE "moveObjectType" "QUOTED_STRING" TO  
                    "QUOTED_STRING" [ USE REPLACE_MODE ]  
moveObjectType = ( "UNQUOTED_STRING" )
```

Keywords And Parameters

parseMoveCommand

Specifies the source object type, source path, and target path for the object to move. Moving objects is subject to the following restrictions: 1. You can only move objects within the current project; you cannot move objects between projects. 2. You cannot move an object within the same parent folder; if your purpose is to rename the object, use the 'OMBALTER...RENAME TO...' command.

QUOTED_STRING

Source and target path specifications can be absolute or relative. To move multiple objects, include a regular expression as the final step of the source path. If you are moving multiple objects, the final step of the target path must be the folder to which the objects are being copied. If you are only moving one object, you can specify the object's original name or a new name as the final step of the target path.

REPLACE_MODE

Use this option to overwrite existing target objects.

moveObjectType

The type of the object(s) to be copied. Valid types are: PROJECT, ORACLE_MODULE, FLAT_FILE_MODULE, BUSINESS_DEFINITION_MODULE, BUSINESS_PRESENTATION_MODULE, SAP_MODULE, CMI_MODULE, PROCESS_FLOW_MODULE,

PROCESS_FLOW_PACKAGE, PROCESS_FLOW, EXPERT_MODULE, EXPERT,
FLAT_FILE,
ADVANCED_QUEUE, STREAMS_QUEUE, QUEUE_TABLE, QUEUE_
PROPAGATION,
STREAMS_CAPTURE_PROCESS, OBJECT_TYPE, VARYING_ARRAY, NESTED_
TABLE, TABLE,
VIEW, MATERIALIZED_VIEW, SEQUENCE, DIMENSION, CUBE, DATA_AUDITOR,
DATA_PROFILE, DATA_RULE, DATA_RULE_MODULE, MAPPING, REAL_TIME_
MAPPING,
PACKAGE, FUNCTION, PROCEDURE, BUSINESS_AREA, COLLECTION,
EXTERNAL_TABLE,
REGISTERED_FUNCTION, ITEM_FOLDER, DRILL_PATH, LIST_OF_VALUES,
DRILL_TO_DETAIL, ALTERNATIVE_SORT_ORDER, PRESENTATION_TEMPLATE
and any user
defined object types.

Examples

```
OMBMOVE TABLE 'MY_TABLE1' TO '../WH2/MY_TABLE1' USE REPLACE_MODE
OMBMOVE VIEW 'MY_VIEW1' TO '../WH2/MY_VIEW1'
OMBMOVE TABLE 'MY_.*' TO '/MY_PROJECT1/WH2'
```

See Also

OMBCOPY

OMBPROFILE

Purpose

To execute the action plans associated with profiling. You can do profiling, correction schema generation and correction map generation with this command.

Prerequisites

This command must be done in the context of a Data Profile. Furthermore, the corresponding transient action plans need to have been created before invoking this command.

Syntax

```
parseProfileCommand = OMBPROFILE ( ( ANALYZE_ACTION_PLAN "QUOTED_STRING" )  
    | ( ( CORRECTION_SCHEMA_ACTION_PLAN "QUOTED_STRING" IN ORACLE_MODULE  
        "QUOTED_STRING" ) [ ( DROP PREVIOUS CORRECTIONS ) ] ) | ( (  
        CORRECTION_MAPS_ACTION_PLAN "QUOTED_STRING" IN ORACLE_MODULE  
        "QUOTED_STRING" ) [ ( DROP PREVIOUS CORRECTIONS ) ] ) )
```

Keywords And Parameters

`parseProfileCommand`
Executes the analyze action plan.

Examples

```
OMBPROFILE ANALYZE_ACTION_PLAN 'ANALYZE_PLAN'
```


OMBRECONCILE

Purpose

Synchronize the target metadata definition with the source metadata definition.

Prerequisites

1. The current context of scripting must be a project atleast.
2. The target, or source objects, or both should exist in the current project.
3. No concurrent user is operating on the target.

Syntax

```

ReconcileCommand = ( OMBRECONCILE | OMBSYNCHRONIZE ) ( "parseIOObject" |
    "parseSourceFCOSCO" TO "parseTargetFCOSCO" USE "(" "setStrategyClause"
    "," "setStrategyClause" ")" )
parseIOObject = ITEM_FOLDER "QUOTED_STRING"
parseSourceFCOSCO = ( ( DATA_AUDITOR "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( MAPPING "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( REAL_TIME_MAPPING "QUOTED_STRING" [
    OPERATOR "QUOTED_STRING" ] ) ) | ( ( FLAT_FILE "QUOTED_STRING" [
    RECORD "QUOTED_STRING" ] ) ) | ( PROCESS_FLOW "QUOTED_STRING" [
    ACTIVITY "QUOTED_STRING" ] ) | ( TABLE | EXTERNAL_TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | CUBE | DIMENSION | ADVANCED_QUEUE |
    OBJECT_TYPE | VARYING_ARRAY | NESTED_TABLE | ACTIVITY_TEMPLATE |
    PACKAGE | FUNCTION | TABLE_FUNCTION | PROCEDURE ) "QUOTED_STRING"
parseTargetFCOSCO = ( ( DATA_AUDITOR "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( MAPPING "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( REAL_TIME_MAPPING "QUOTED_STRING" [
    OPERATOR "QUOTED_STRING" ] ) ) | ( ( FLAT_FILE "QUOTED_STRING" [
    RECORD "QUOTED_STRING" ] ) ) | ( ( PROCESS_FLOW "QUOTED_STRING" [
    ACTIVITY "QUOTED_STRING" ] ) ) | ( TABLE | EXTERNAL_TABLE | VIEW |
    MATERIALIZED_VIEW | ADVANCED_QUEUE | VARYING_ARRAY | NESTED_TABLE |
    OBJECT_TYPE | ACTIVITY_TEMPLATE | FUNCTION ) "QUOTED_STRING"
setStrategyClause = ( RECONCILE_STRATEGY | SYNCHRONIZE_STRATEGY )
    "retrieveReconcileStrategyClause" | MATCHING_STRATEGY
    "retrieveMatchingStrategyClause"
retrieveReconcileStrategyClause = "QUOTED_STRING"
retrieveMatchingStrategyClause = "QUOTED_STRING"

```

Keywords And Parameters

ReconcileCommand

Synchronizes the target metadata definition with the source metadata definition.

setStrategyClause

The strategy to be used for synchronization. Must be one of
RECONCILE_STRATEGY OR **MATCHING_STRATEGY**

RECONCILE_STRATEGY

MERGE or REPLACE.

MERGE : Updates the matching objects in the target with the metadata definition in the source object and creates new objects in the target for source objects that do not match.

REPLACE : Updates the matching objects in the target with the metadata definition in the source object, creates a new object in the target for source objects that do not match, and deletes objects in the target that have no match in the source.

MATCHING_STRATEGY

Indicates the matching strategy to be used between the object types in source and target. Currently, the list of available matching strategies for a source and target combination are listed subsequently. For synchronization between RECORD and EXTERNAL_TABLE, the valid matching strategies are MATCH_BY_OBJECT_ID, MATCH_BY_OBJECT_NAME, MATCH_BY_OBJECT_POSITION. For

synchronization involving OPERATOR of MAP, the available matching strategies are MATCH_BY_OBJECT_ID, MATCH_BY_OBJECT_NAME, MATCH_BY_OBJECT_POSITION, MATCH_BY_OBJECT_ID_AND_NAME, MATCH_BY_OBJECT_ID_AND_POSITION, MATCH_BY_OBJECT_NAME_AND_POSITION,

MATCH_BY_OBJECT_ID_AND_NAME_AND_POSITION. Note that the source map might be

modified during outbound synchronization of maps, which would require locking both the source and the target before synchronization invocation.

retrieveReconcileStrategyClause

Must of one of 'MERGE' or 'REPLACE'

retrieveMatchingStrategyClause

String representing the matching paradigm between the source and the target object. The list varies for every source and target object type.

Examples

```
OMBRECONCILE EXTERNAL_TABLE 'et1' TO EXTERNAL_TABLE 'et2'  
USE (RECONCILE_STRATEGY 'MERGE', MATCHING_STRATEGY 'MATCH_BY_  
OBJECT_NAME')
```

```
OMBRECONCILE MAPPING 'm1' OPERATOR 'o1' TO TABLE 't1'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY  
'MATCH_BY_OBJECT_NAME')
```

```
OMBRECONCILE MAPPING 'm1' OPERATOR 'o1' TO FLAT_FILE 'f1'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY  
'MATCH_BY_OBJECT_NAME')
```

```
OMBRECONCILE MAPPING 'm1' OPERATOR 'o1' TO MAPPING 'm1' OPERATOR  
'o2'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY 'MATCH_BY_  
OBJECT_ID')
```

```
OMBRECONCILE ITEM_FOLDER 'F1'
```

```
OMBRECONCILE TABLE 't1' TO DATA_AUDITOR 'a1' OPERATOR 'o2'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY 'MATCH_BY_  
OBJECT_ID')
```

For Item Folder synchronization the source is obtained from the repository.

For simple item folders the source

is the table that the item folder is based on. For complex item folders

the source is the dependent folders.

The user does not have to specify the source.

OMBREDEFINE ASSOCIATION_DEFINITION

Purpose

To redefine an association between two classes (types).

Prerequisites

Association definition to be redefined should already exist. This command can be executed for any association definition regardless of current context. User must have CREATE_EXTENSIONMODEL system privilege and has to be connected in single user mode to run this command.

Syntax

```

parseRedefineAssociationCommand = OMBREDEFINE ASSOCIATION_DEFINITION
    "QUOTED_STRING" ( RENAME TO "QUOTED_STRING" |
    "setAssociationDefinitionPropertiesClause" | (
    "modifyDependencyDefinitionClause" |
    "deleteDependencyDefinitionClause" | "addDependencyDefinitionClause"
    )+ )
setAssociationDefinitionPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
modifyDependencyDefinitionClause = MODIFY DEPENDENCY_DEFINITION
    "QUOTED_STRING" "setDependencyDefinitionPropertiesClause"
deleteDependencyDefinitionClause = DELETE DEPENDENCY_DEFINITION
    "QUOTED_STRING"
addDependencyDefinitionClause = ADD DEPENDENCY_DEFINITION "QUOTED_STRING"
    [ "setDependencyDefinitionPropertiesClause" ]
propertyNameList = "propertyNameClause" { "," "propertyNameClause" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setDependencyDefinitionPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
propertyNameClause = ( "UNQUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

parseRedefineAssociationCommand

Redefine an association between two classes..

setAssociationDefinitionPropertiesClause

Basic properties for re-defining ASSOCIATION_DEFINITION:

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: N/A

A descriptive text for this association.

Name: ROLE_1_MIN_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Minimum cardinality of role 1. Value can be positive integer.

Name: ROLE_1_MAX_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Maximum cardinality of role 1. Value can be positive integer or 'INFINITE'.

Name: ROLE_1_NAVIGABLE

Type: BOOLEAN

Valid Values: true, false

Default: From user defined class side default is true. From OWB class side default is false.

Navigability of role 1.

Name: ROLE_2_MIN_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Minimum cardinality of role 2. Value can be positive integer.

Name: ROLE_2_MAX_CARDINALITY

Type: STRING(200)

Valid Values: N/A

Default: N/A

Maximum cardinality of role 2. Value can be positive integer or 'INFINITE'.

Name: ROLE_2_NAVIGABLE

Type: BOOLEAN

Valid Values: true, false

Default: From user defined class side default is true. From OWB class side default is false.

Navigability of role 2.

modifyDependencyDefinitionClause

Modifies the properties of the specified dependency for this association.

deleteDependencyDefinitionClause

Un-mark this association so that the dependency engine will ignore it when computing the lineage and impact dependencies of that type.

addDependencyDefinitionClause

Mark this association so that the dependency engine will consider it when computing the lineage and impact dependencies of the specified type. The only dependency type allowed here for now is 'DATAFLOW'.

propertyNameList

The list of property names.

propertyValueList

The list of property values being set.

setDependencyDefinitionPropertiesClause

Basic dependency-related properties for this association:

Name: SOURCE_ROLE_ID

Type: STRING(200)

Valid Values: ROLE_1, ROLE_2

Default: If one of the ends is a OWB class, then that is the default source. If both ends are user defined classes, then the association is default bidirectional.

Identifies the role (end) of the association which serves as the source for the dependency flow.

Name: BIDIRECTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: If one of the ends is a OWB class, then default is false. If both ends are user defined classes, then default is true.

Specifies whether the association is bi-directional for the dependency flow.

propertyNameClause

Name of a property.

propertyValue

Value of a property.

Examples

```
OMBREDEFINE ASSOCIATION_DEFINITION 'UD_ASSOC2'  
SET PROPERTIES (DESCRIPTION, ROLE_1_MIN_CARDINALITY,  
ROLE_1_MAX_CARDINALITY, ROLE_1_NAVIGABLE) VALUES ('Some association  
description text', '1', 'INFINITE', 'true') DELETE DEPENDENCY_DEFINITION  
'DATAFLOW'
```

This will redefine association 'UD_ASSOC2' to change the description, as well as the navigability, minimum and maximum cardinality of role 1. Also it removes DATAFLOW dependency marking on this association.

See Also

OMBDEFINE ASSOCIATION_DEFINITION, OMBDESCRIBE ASSOCIATION_DEFINITION

OMBREDEFINE CLASS_DEFINITION

Purpose

To redefine a class.

Prerequisites

Class definition to be redefined should already exist. This command can be executed for any class definition regardless of current context. User must have CREATE_EXTENSIONMODEL system privilege, and user has to connect in single user mode.

The valid types to define user defined properties are: INTEGER, STRING, FLOAT, DOUBLE, DATE, TIMESTAMP, BOOLEAN, LONG, FILE, URL

Syntax

```

parseRedefineClassCommand = OMBREDEFINE CLASS_DEFINITION "QUOTED_STRING" (
    ( "setClassDefinitionPropertiesClause" |
      "setClassDefinitionIconSetClause" |
      "unsetClassDefinitionIconSetClause" ) [ "modifySubDefinitionsClause" ]
    | "modifySubDefinitionsClause" )
setClassDefinitionPropertiesClause = SET PROPERTIES "(" "propertyNameList"
    ")" VALUES "(" "propertyValueList" ")"
setClassDefinitionIconSetClause = SET REF ICONSET "QUOTED_STRING"
unsetClassDefinitionIconSetClause = UNSET REF ICONSET
modifySubDefinitionsClause = ( "addPropertyDefinitionClause" |
    "modifyPropertyDefinitionClause" | "deletePropertyDefinitionClause" |
    "addPropertyGroupDefinitionClause" |
    "modifyPropertyGroupDefinitionClause" |
    "deletePropertyGroupDefinitionClause" | "addChildTypeClause" |
    "deleteChildTypeClause" )+
propertyNameList = "propertyNameClause" { "," "propertyNameClause" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addPropertyDefinitionClause = ( ADD [ ( ( CONFIGURATION |
    PHYSICAL_CONFIGURATION ) | LOGICAL | USER_DEFINED ) ]
    PROPERTY_DEFINITION "QUOTED_STRING"
    "setPropertyDefinitionPropertiesClause" )
modifyPropertyDefinitionClause = ( MODIFY PROPERTY_DEFINITION
    "QUOTED_STRING" ( RENAME TO "QUOTED_STRING" |
    "setPropertyDefinitionPropertiesClause" ) )
deletePropertyDefinitionClause = ( DELETE PROPERTY_DEFINITION
    "QUOTED_STRING" )
addPropertyGroupDefinitionClause = ADD PROPERTY_GROUP_DEFINITION
    "QUOTED_STRING" "setPropertyGroupDefinitionPropertiesClause"
modifyPropertyGroupDefinitionClause = ( MODIFY PROPERTY_GROUP_DEFINITION
    "QUOTED_STRING" ( RENAME TO "QUOTED_STRING" |
    "setPropertyGroupDefinitionPropertiesClause" ) )
deletePropertyGroupDefinitionClause = ( DELETE PROPERTY_GROUP_DEFINITION
    "QUOTED_STRING" )
addChildTypeClause = ( ADD CHILD_TYPE "QUOTED_STRING" )
deleteChildTypeClause = ( DELETE CHILD_TYPE "QUOTED_STRING" )
propertyNameClause = ( "UNQUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |

```



```
"FLOATING_POINT_LITERAL" )
setPropertyDefinitionPropertiesClause = SET PROPERTIES "("
  "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
setPropertyGroupDefinitionPropertiesClause = SET PROPERTIES "("
  "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
```

Keywords And Parameters

parseRedefineClassCommand

Redefine a class definition.

CLASS_DEFINITION

Redefine a class definition.

QUOTED_STRING

Name of the class definition.

propertyNameList

The list of properties for the class definition.

propertyValueList

The list of values provided for the class definition.

propertyNameClause

The name of the property.

UNQUOTED_STRING

The name of the property for the class definition.

propertyValue

The value of the property.

QUOTED_STRING

The value in string format of the property for the class definition.

INTEGER_LITERAL

The integer value of the property for the class definition.

FLOATING_POINT_LITERAL

The float value of the property for the class definition.

Examples

```
OMBREDEFINE CLASS_DEFINITION 'TABLE'
```

```
ADD PROPERTY_DEFINITION 'UDP_TBL_1' SET PROPERTIES (TYPE, DEFAULT_
VALUE)
```

```
VALUES ('INTEGER', '100')
```

This will add an User-defined property definition to class definition
'TABLE'.

```
OMBREDEFINE CLASS_DEFINITION 'TABLE' DELETE PROPERTY_DEFINITION
'UDP_TBL_1'
```

This will delete property definition 'UDP_TBL_1' from class definition
'TABLE'. Property definition must exist before deleting it.

```
OMBREDEFINE CLASS_DEFINITION 'TABLE' MODIFY PROPERTY_DEFINITION
'UDP_TBL_1'
```

```
SET PROPERTIES (DEFAULT_VALUE, BUSINESS_NAME)
```

```
VALUES ('99', 'UDP_TBL_2')
```

This will change the name of property definition to 'UDP_TBL_2' and default
value to 99. Property definition must exist before modifying it. TYPE can
not be changed for property definition.

See Also

```
OMBDESCRIBE CLASS_DEFINITION
```

OMBREGISTER LOCATION

Purpose

Register a location with a Control Center.

Prerequisites

Must be in the context of a project and connected to a Control Center.

Syntax

```
registerLocationCommand = OMBREGISTER LOCATION "QUOTED_STRING" [ REUSE ]
```

Keywords And Parameters

registerLocationCommand

Register a location.

QUOTED_STRING

The name of the location to register.

Examples

```
OMBREGISTER LOCATION 'MY_ORACLE_LOCATION' REUSE
```

OMBREGISTER USER

Purpose

To register a database user as a Warehouse Builder user.

Prerequisites

Commit or Rollback is needed if there is any change.

Syntax

```
parseRegisterUserCommand = ( OMBREGISTER USER "QUOTED_STRING" [ SET
    PROPERTIES "(" "propertyNameList" ")" VALUES "(" "propertyValueList"
    ")" ] [ WITH CREATE DBUSER OPTION IDENTIFIED BY "QUOTED_STRING"
    THROUGH SYSDBA CONNECTION "UNQUOTED_STRING" ] )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parseRegisterUserCommand

This command registers a Warehouse Builder user.

propertyNameList

A list of valid properties are as shown.

Basic properties for USER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the User

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the User

Name: ISTARGETSCHEMA

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the user will be set up as target schema for deployment; and also the property TARGETSCHEMAPWD must be provided when you are setting the ISTARGETSCHEMA as true.

Name: TARGETSCHEMAPWD

Type: STRING(30)

Valid Values: N/A

Default: N/A

This properties will be provided only when you are setting ISTARGETSCHEMA as true, so that the necessary target schema objects can be installed into the potential target schema. And this property cannot be retrieved due to security consideration.

User preferences:

Name: LOCALE

Type: STRING

Valid Values: Albanian, Arabic, Bulgarian, Byelorussian, Catalan, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, French, German, Greek, Hebrew, Hungarian, Icelandic, Italian, Japanese, Korean, Lithuanian, Macedonian, Norwegian, Polish, Portuguese, Romanian, Russian, Serbian, Serbo_Croatian, Slovak, Slovenian, Spanish, Swedish, Thai, Turkish, Ukranian

Default: "

Name: SHOW_PROJECT

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_MODULE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_LOCATION
Type: BOOLEAN
Valid Values: true, false
Default: true

Name: SHOW_ACTION
Type: BOOLEAN
Valid Values: true, false
Default: true

Name: SHOW_TYPE
Type: BOOLEAN
Valid Values: true, false
Default: true

Name: DEFAULT_PROFILE_LOCATION
Type: STRING
Valid Values: N/A
Default: "

Name: ALLOW_UNDO_REDO
Type: BOOLEAN
Valid Values: true, false
Default: false

Name: PAUSE_AFTER_COMPILE
Type: BOOLEAN
Valid Values: true, false
Default: false

Name: PROMPT_FOR_COMMIT
Type: BOOLEAN
Valid Values: true, false
Default: false

Name: PROMPT_FOR_JOB_NAME

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_EXECUTION_PARAMS

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_MONITOR

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DEPLOYMENT_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DEPLOYMENT_DEPENDENCIES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_MONITOR_RESULTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_MONITOR_LOGFILE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: PERSONALITY

Type: STRING
Valid Values: N/A
Default: Default

Name: SHOW_GUIDED_ASSISTANCE
Type: BOOLEAN
Valid Values: true, false
Default: false

Name: HIDE_WIZARD_WELCOME_PAGES
Type: BOOLEAN
Valid Values: true, false
Default: false

Name: SHOW_DELETE_CONFIRMATION
Type: BOOLEAN
Valid Values: true, false
Default: true

Name: RECYCLE_DELETED_OBJECTS
Type: BOOLEAN
Valid Values: true, false
Default: true

Name: EMPTY_RECYCLE_BIN
Type: BOOLEAN
Valid Values: true, false
Default: false

Name: CLEAR_CLIPBOARD
Type: BOOLEAN
Valid Values: true, false
Default: true

Name: SHOW_GENERATION_PROJECT
Type: BOOLEAN
Valid Values: true, false

Default: true

Name: SHOW_GENERATION_MODULE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_LOCATION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_GENERATION_ACTION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_GENERATION_TYPE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_FILE_PATH

Type: STRING(1000)

Valid Values: N/A

Default: "

Name: LOG_FILE_NAME

Type: STRING(1000)

Valid Values: N/A

Default: log

Name: LOG_FILE_MAX_SIZE

Type: STRING

Valid Values: 1-10000000

Default: 100

Name: LOG_ERROR_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_WARNING_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_INFORMATION_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: NAMING_MODE

Type: STRING

Valid Values: PHYSICAL_NAMING_MODE, BUSINESS_NAMING_MODE

Default: PHYSICAL_NAMING_MODE

Name: PROPAGATE_NAME_CHANGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: DESIGNREPOS_PWD_PERSIST

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: RUNTIMEREPOS_PWD_SHARE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: DEFAULT_SEC_POLICY

Type: STRING

Valid Values: MINIMUM_SECURITY, MAXIMUM_SECURITY

Default: MINIMUM_SECURITY

Examples

```
OMBREGISTER USER 'USER1' SET PROPERTIES(BUSINESS_NAME, DESCRIPTION,  
ISTARGETSCHEMA,TARGETSCHEMAPWD) VALUES('developer user1', 'one user  
from  
developer group', 'true','passwordForUser1')  
WITH CREATE DBUSER OPTION IDENTIFIED BY 'passwordForUser1' THROUGH  
SYSDBA  
CONNECTION sys/change_on_install
```

This will first use database DBA connection credential to create a database user USER1 identified by password passwordForUser1, and then register the user USER1 with current working repository.

Note: if the database user USER1 already exists, WITH CREATE DBUSER OPTION should be omitted, for example:

```
OMBREGISTER USER 'USER1' SET PROPERTIES(BUSINESS_NAME, DESCRIPTION,  
ISTARGETSCHEMA) VALUES('developer user1', 'one user from developer  
group','false')
```

See Also

OMBUNREGISTER USER, OMBALTER USER, OMBRETRIEVE USER

OMBRESTORE SNAPSHOT

Purpose

A snapshot is a history point of individual or group of components. The user can recover the components' previously captured states by using the snapshot restore functionality.

Prerequisites

A component can be restored from a snapshot into the current repository.

Syntax

```
parseRestoreCommand = OMBRESTORE "restoreSnapshotCommand"  
restoreSnapshotCommand = ( SNAPSHOT "QUOTED_STRING" [ CASCADE UP ] [  
    IGNORE CLASS_DEFINITION ( DIFFERENCES | DIFF ) ] [ FOR (  
    "UNQUOTED_STRING" "QUOTED_STRING" ) ] )
```

Keywords And Parameters

parseRestoreCommand

Root production for OMBRESTORE SNAPSHOT.

restoreSnapshotCommand

To restore snapshot components into the repository.

QUOTED_STRING

Name of the snapshot from which components are to be restored.

CASCADE

CASCADE UP - Optional clause for letting the user restore a component even if its parent does not exist in the current repository.

CLASS_DEFINITION

IGNORE CLASS_DEFINITION DIFF - Optional clause for letting the user restore a snapshot whether or not the meta-model of the snapshot is different from that of the current repository.

FOR

Optional component clause for partial restore. This can be used to specify which components of a snapshot are to be restored.

Examples

OMBRESTORE SNAPSHOT 'S1'

This command restores all components from the snapshot into the repository. If the corresponding components are not found in the repository, then they appear as newly recovered components from history.

OMBRESTORE SNAPSHOT 'S1' FOR TABLE '/Project/WH1/T1'

This command replaces the current definition of the component in the repository with the snapshot component.

OMBRESTORE SNAPSHOT 'S1' CASCADE UP

This command restores all objects of the snapshot into the repository whether or not their parents exist in the current repository. If not, components along with their corresponding parents are restored.

OMBRESTORE SNAPSHOT 'S1' IGNORE CLASS_DEFINITION DIFF

This command restores the snapshot regardless of whether or not the meta-model of the snapshot is different to that of the current repository.

See Also

OMBCREATE SNAPSHOT, OMBALTER SNAPSHOT, OMBDROP SNAPSHOT, OMBCOMPARE SNAPSHOT, OMBLIST SNAPSHOT, OMBRETRIEVE SNAPSHOT

OMBREVERT

Purpose

Perform revert action on the repository (all objects are reverted to the last saved state).

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseRevertCommand = OMBREVERT
```

Keywords And Parameters

parseRevertCommand
Specify revert command.

Examples

```
OMBREVERT
```

See Also

OMBSAVE

OMBREVOKE DEFAULT_OBJ_PRIV

Purpose

To change the default object privilege setting property of a user, basically it will revoke a list of default object privileges from a list of users or roles. These object privileges will not be granted to the specified user or role on the new objects whenever the owner creates them.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseRevokeDefObjPrivCommand = OMBREVOKE DEFAULT ( OBJ_PRIV |
    OBJECT_PRIVILEGE ) "objPrivNameList" FROM ( USER "userOrRoleNameList"
    | ROLE "userOrRoleNameList" ) [ FOR USER "QUOTED_STRING" ]
objPrivNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
userOrRoleNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
```

Keywords And Parameters

`parseRevokeDefObjPrivCommand`

This clause changes the default object privilege setting property of a user through revoking a list of default object privileges from a list of users or roles.

`objPrivNameList`

Object privileges. Valid object privileges are: READ, COMPILE, EDIT, FULL_CONTROL.

Examples

```
OMBREVOKE DEFAULT OBJ_PRIV EDIT FROM ROLE 'DEVELOPMENT_ROLE'
```

the current login user revokes default object privilege EDIT from role DEVELOPMENT_ROLE, thus whenever the current login user creates an object, the role DEVELOPMENT_ROLE will not be granted EDIT privilege on that object any more

```
OMBREVOKE DEFAULT OBJ_PRIV EDIT FROM ROLE 'DEVELOPMENT_ROLE'
FOR USER
```

```
'USER1'
```

the current user revokes default object privilege EDIT from role DEVELOPMENT_ROLE on behalf of user USER1, thus whenever the user USER1

creates an object, the role DEVELOPMENT_ROLE will not be granted EDIT privilege on that object. Note if the current login user does not change the default object privilege setting for himself, the FOR USER statement is required.

See Also

OMBGRANT DEFAULT_OBJ_PRIV, OMBLIST DEFAULT_OBJ_PRIVS

OMBREVOKE OBJ_PRIV

Purpose

To revoke a list of object privileges on the specified objects from a list of users or roles.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseRevokeObjPrivCommand = OMBREVOKE ( OBJ_PRIV | OBJECT_PRIVILEGE )
    "objPrivNameList" ON "UNQUOTED_STRING" "QUOTED_STRING" [ CASCADE ]
    FROM ( USER "userOrRoleNameList" | ROLE "userOrRoleNameList" )
objPrivNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
userOrRoleNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
```

Keywords And Parameters

parseRevokeObjPrivCommand

This clause revokes a list of object privileges on the specified objects from a list of users or roles.

UNQUOTED_STRING

The type of the objects to be listed. Valid object types are: PROJECT,

ORACLE_MODULE, FLAT_FILE_MODULE, PROCESS_FLOW_MODULE, SAP_MODULE,

BUSINESS_DEFINITION_MODULE, BUSINESS_PRESENTATION_MODULE,

TRANSFORMATION_MODULE, PACKAGE, ADVANCED_QUEUE, BUSINESS_AREA, COLLECTION,

CONNECTOR, CUBE, DIMENSION, EXTERNAL_TABLE, FLAT_FILE, FUNCTION,

REGISTERED_FUNCTION, MAPPING, MATERIALIZED_VIEW, OBJECT_TYPE, PROCEDURE,

ITEM_FOLDER, DRILL_PATH, DRILLS_TO_DETAIL, ALTERNATIVE_SORT_ORDER,

LISTS_OF_VALUE, PRESENTATION_TEMPLATE, SEQUENCE, TABLE, PROCESS_FLOW,

PROCESS_FLOW_PACKAGE, LOCATION, CONTROL_CENTER, CONFIGURATION, DEPLOYMENT,

VIEW.

objPrivNameList

Object privileges. Valid object privileges are: READ, COMPILE, EDIT, FULL_CONTROL.

Examples

```
OMBREVOKE OBJ_PRIV EDIT ON ORACLE_MODULE '/MY_PROJECT/WH'  
FROM USER 'USER1'
```

will revoke EDIT on module WH from user USER1

```
OMBREVOKE OBJ_PRIV EDIT ON PROJECT 'MY_PROJECT' CASCADE FROM  
ROLE
```

```
'EVERYONE'
```

will revoke object privilege EDIT from role EVERYONE on project MY_PROJECT and all its child objects cascade.

```
OMBREVOKE OBJ_PRIV READ ON TABLE '/MY_PROJECT/WH/EMP*' FROM  
USER 'USER1'
```

will revoke READ from user USER1 on all tables under module /MY_PROJECT/WH whose names match regular expression EMP*.

See Also

OMBGRANT OBJ_PRIV, OMBLIST OBJ_PRIVS

OMBREVOKE ROLE

Purpose

To revoke a list of Warehouse Builder roles from a list of Warehouse Builder users.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseRevokeRoleCommand = OMBREVOKE ROLE "userOrRoleNameList" FROM USER  
    "userOrRoleNameList"  
userOrRoleNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
```

Keywords And Parameters

`parseRevokeRoleCommand`

This clause revokes a list of Warehouse Builder roles from a list of Warehouse Builder users.

Examples

```
OMBREVOKE ROLE 'DEV' , 'QA' FROM USER 'USER1', 'USER2'
```

will revoke Warehouse Builder role 'DEV' and 'QA' from user 'USER1' and 'USER2'

See Also

OMBGRANT ROLE, OMBLIST ROLES

OMBREVOKE SYS_PRIV

Purpose

To revoke a list of system privileges from a list of users or roles.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseRevokeSysPrivCommand = OMBREVOKE ( SYS_PRIV | SYSTEM_PRIVILEGE )  
    "sysPrivNameList" FROM ( USER "userOrRoleNameList" | ROLE  
    "userOrRoleNameList" )  
sysPrivNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }  
userOrRoleNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
```

Keywords And Parameters

parseRevokeSysPrivCommand

This clause revokes a list of system privileges from a list of users or roles.

sysPrivNameList

List of system privileges. Valid system privileges are:CREATE_PROJECT,
CREATE_MIVDEFINITION,CREATE_EXTENSIONMODEL, CONTROL_CENTER_
ADMIN,
CONTROL_CENTER_DEPLOY, CONTROL_CENTER_EXECUTE, CREATE_
SNAPSHOT

Examples

OMBREVOKE SYS_PRIV 'CREATE_SNAPSHOT' FROM USER 'USER1', 'USER2'
will revoke system privilege CREATE_SNAPSHOT from user USER1 and USER2.

See Also

OMBGRANT SYS_PRIV, OMBLIST SYS_PRIVS

OMBROLLBACK

Purpose

Perform rollback action on the repository.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseRollbackCommand = OMBROLLBACK
```

Keywords And Parameters

parseRollbackCommand
Specify rollback command.

Examples

```
OMBROLLBACK
```

See Also

```
OMBCOMMIT
```

OMBSAVE

Purpose

Perform save action on the repository (all objects are saved).

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseSaveCommand = OMBSAVE
```

Keywords And Parameters

parseSaveCommand
Specify save command.

Examples

```
OMBSAVE
```

See Also

```
OMBREVERT
```

OMBSHOW

Purpose

To show the session or system parameter information.

Prerequisites

Need to connect to a repository.

Syntax

```
parseShowCommand = ( OMBSHOW "UNQUOTED_STRING" )
```

Keywords And Parameters

`parseShowCommand`

Show the session or system parameters. The Valid session or system parameter names are: USER, REPOSITORY, CONNECTION, REPOSITORY_VERSION, CLIENT_VERSION.

`UNQUOTED_STRING`

Specify the session or system parameter name.

The Valid session or system parameter names are: USER, REPOSITORY, CONNECTION, REPOSITORY_VERSION, CLIENT_VERSION.

Examples

```
OMBSHOW USER
```

will show the session user name.

OMBSTART

Purpose

To execute objects from the Control Center. If the ASYNCHRONOUS keyword is specified then this command will return the Execution Job

Identifier. Note, this can then be passed to the OMBSTOP command.

The 'Best Practice' usage of this command is within the context of a Design Repository connection. However, advanced users may use this command with a 'Control Center Only' usage. Note, the MAPPING keyword cannot be used in this mode and you must specify the deployed Mapping type.

If it is required to fully specify the Deployed Mapping type then the possible values are ABAPFILE, DATAAUDITOR, DATAPROFILE, PLSQLMAP, PROCESSFLOW, SQLLOADERCONTROLFILE and SCHEDULEDJOB.

The Deployment Location of the object is optional if the name of the executable object is unique in the Control Center. However, please specify the 'CONTROL_CENTER' location for types that are deployed to the Control Center (SQL Loader and SAP Mappings).

The System Parameters that can be overridden are

PLSQL Mapping - BULK_SIZE COMMIT_FREQUENCY DEFAULT_AUDIT_LEVEL
DEFAULT_OPERATING_MODE DEFAULT_PURGE_GROUP MAXIMUM_
NUMBER_OF_ERRORS

SQL Loader Mapping - AUDIT BIND_SIZE CONTROL_FILE_LOCATION
CONTROL_FILE_NAME DEFAULT_PURGE_GROUP DIRECT_MODE DISCARD_
MAX
ERRORS_ALLOWED LOG_FILE_LOCATION LOG_FILE_NAME PERFORM_
PARALLEL_LOAD
READ_SIZE RECORDS_TO_LOAD RECORDS_TO_SKIP ROWS_PER_COMMIT
SKIP_INDEX_MAINTENANCE SKIP_UNUSABLE_INDEXES

SAP Mapping - BACKGROUND CONTROL_FILE_NAME DATA_FILE_NAME
FILE_DELIMITER_FOR_STAGING_FILE LOG_FILE_NAME NESTED_LOOP
PRIMARY_FOREIGN_KEY_FOR_JOIN SAP_LOCATION SQL_JOIN_COLLAPSING

STAGING_FILE_DIRECTORY USE_SELECT_SINGLE

Data Auditor - BULK_SIZE COMMIT_FREQUENCY DEFAULT_AUDIT_LEVEL
 DEFAULT_OPERATING_MODE DEFAULT_PURGE_GROUP MAXIMUM_
 NUMBER_OF_ERRORS

Data Profile - There are no System Parameters defined.

Prerequisites

'Best Practice' usage requires a Design Repository connection and the context to be set to that of executable object. The 'Control Center Only' usage does not require a Design Repository connection and therefore no context.

Syntax

```
parseExecuteCommand = ( OMBSTART | OMBEXECUTE ) ( ( MAPPING | DATA_AUDITOR
  | PROCESS_FLOW | SCHEDULABLE | REAL_TIME_MAPPING | "UNQUOTED_STRING"
  ) "QUOTED_STRING" [ AS "QUOTED_STRING" ] [ IN "QUOTED_STRING" ] ) [
  OVERRIDE SYSTEM_PARAMETER "overrideParameters" ] [ OVERRIDE
  CUSTOM_PARAMETER "overrideParameters" ] [ ASYNCHRONOUS ]
overrideParameters = "(" "stringList" ")" VALUES "(" "stringList" ")"
stringList = "stringListValue" { ",", "stringListValue" }
stringListValue = ( "QUOTED_STRING" )
```

Keywords And Parameters

parseExecuteCommand

Specify Execute command.

QUOTED_STRING

The name of the object to be executed or the execution's Audit Name.

AS

The execution's Audit Name.

IN

The deployed object's location.

OVERRIDE

Override an Execution Parameter.

SYSTEM_PARAMETER

Specify to override a System Parameter.

CUSTOM_PARAMETER

Specify to override a Custom Parameter.

ASYNCHRONOUS

Execute in an asynchronous mode.

Examples

Best Practice usage:

```
OMBSTART MAPPING 'MY_TABLE_MAPPING' IN 'MY_DEPLOYMENT_
LOCATION'
```

```
OMBSTART MAPPING 'MY_SQLLOADER_MAP' IN 'CONTROL_CENTER'
```

```
OMBSTART MAPPING 'MY_SAP_MAP' IN 'CONTROL_CENTER'
```

Control Center Only:

```
OMBSTART PLSQLMAP 'MY_TABLE_MAPPING' IN 'MY_DEPLOYMENT_
LOCATION'
```

```
OMBSTART SQLLOADERCONTROLFILE 'MY_SQLLOADER_MAP' IN 'CONTROL_
CENTER'
```

```
OMBSTART ABAPFILE 'MY_SAP_MAP' IN 'CONTROL_CENTER'
```

OMBSTATS

Purpose

This command will display statistical information about the scripting run, such as memory usage, and elapsed time.

Prerequisites

None.

Syntax

```
parseStatisticsCommand = OMBSTATS
```

Examples

```
OMBSTATS
```

OMBSTOP

Purpose

To stop jobs running in the Control Center.

Prerequisites

A Control Center connection. Note, a Design Repository connection is not required.

Syntax

```
parseStopCommand = OMBSTOP ( "INTEGER_LITERAL" )
```

Keywords And Parameters

parseStopCommand

Specify Stop command.

INTEGER_LITERAL

The job id.

Examples

```
OMBSTOP 123456789
```

OMBSWITCHBACKMODE

Purpose

To switch the current repository connection back to the previous mode. Valid modes are: SINGLE_USER_MODE (that is, exclusive) and MULTIPLE_USER_MODE.

Prerequisites

Must be connected to an OWB repository and not have any outstanding/unsaved work. If not already connected, use OMBCONNECT first. If there exists work that is not saved, use either OMBSAVE or OMBREVERT first. If the current mode is the connection mode (that is, there is no previous mode) then the mode is not switched.

Syntax

```
parseSwitchBackModeCommand = OMBSWITCHBACKMODE
```

Keywords And Parameters

parseSwitchBackModeCommand
Specify switch back command.

Examples

```
OMBSWITCHBACKMODE
```

See Also

OMBDISPLAYCURRENTMODE, OMBSWITCHMODE, OMBCONNECT, OMBSAVE, OMBREVERT

OMBSWITCHMODE

Purpose

To switch the current repository connection between SINGLE_USER_MODE (that is, exclusive) and MULTIPLE_USER_MODE.

Prerequisites

Must be connected to an OWB repository and not have any outstanding/unsaved work. If not already connected, use OMBCONNECT first. If there exists work that is not saved, use either OMBSAVE or OMBREVERT first.

Syntax

```
parseSwitchModeCommand = OMBSWITCHMODE ( SINGLE_USER_MODE |  
MULTIPLE_USER_MODE )
```

Keywords And Parameters

parseSwitchModeCommand
Specify switch command.

SINGLE_USER_MODE

If specified, the user's current connection will be changed to permit them to use the repository exclusively.

MULTIPLE_USER_MODE

If specified, the user's current connection will be changed to permit more than one session to work on the repository at the same time.

Examples

```
OMBSWITCHMODE SINGLE_USER_MODE
```

```
OMBSWITCHMODE MULTIPLE_USER_MODE
```

See Also

OMBDISPLAYCURRENTMODE, OMBCONNECT, OMBSAVE, OMBREVERT

OMBSYNCHRONIZE

Purpose

Synchronize the target metadata definition with the source metadata definition.

Prerequisites

1. The current context of scripting must be a project atleast.
2. The target, or source objects, or both should exist in the current project.
3. No concurrent user is operating on the target.

Syntax

```

ReconcileCommand = ( OMBRECONCILE | OMBSYNCHRONIZE ) ( "parseIOObject" |
    "parseSourceFCOSCO" TO "parseTargetFCOSCO" USE "(" "setStrategyClause"
    "," "setStrategyClause" ")" )
parseIOObject = ITEM_FOLDER "QUOTED_STRING"
parseSourceFCOSCO = ( ( DATA_AUDITOR "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( MAPPING "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( PLUGGABLE_MAPPING "QUOTED_STRING" [
    OPERATOR "QUOTED_STRING" ] ) ) | ( ( REAL_TIME_MAPPING "QUOTED_STRING" [
    [ OPERATOR "QUOTED_STRING" ] ) ) | ( ( FLAT_FILE "QUOTED_STRING" [
    RECORD "QUOTED_STRING" ] ) ) | ( ( PROCESS_FLOW "QUOTED_STRING" [
    ACTIVITY "QUOTED_STRING" ] ) ) | ( TABLE | EXTERNAL_TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | CUBE | DIMENSION | ADVANCED_QUEUE |
    OBJECT_TYPE | VARYING_ARRAY | NESTED_TABLE | ACTIVITY_TEMPLATE |
    PACKAGE | FUNCTION | TABLE_FUNCTION | PROCEDURE ) "QUOTED_STRING"
parseTargetFCOSCO = ( ( DATA_AUDITOR "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( MAPPING "QUOTED_STRING" [ OPERATOR
    "QUOTED_STRING" ] ) ) | ( ( PLUGGABLE_MAPPING "QUOTED_STRING" [
    OPERATOR "QUOTED_STRING" ] ) ) | ( ( REAL_TIME_MAPPING "QUOTED_STRING" [
    [ OPERATOR "QUOTED_STRING" ] ) ) | ( ( FLAT_FILE "QUOTED_STRING" [
    RECORD "QUOTED_STRING" ] ) ) | ( ( PROCESS_FLOW "QUOTED_STRING" [
    ACTIVITY "QUOTED_STRING" ] ) ) | ( TABLE | EXTERNAL_TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | ADVANCED_QUEUE | VARYING_ARRAY |
    NESTED_TABLE | OBJECT_TYPE | ACTIVITY_TEMPLATE | FUNCTION )
    "QUOTED_STRING"
setStrategyClause = ( RECONCILE_STRATEGY | SYNCHRONIZE_STRATEGY )
    "retrieveReconcileStrategyClause" | MATCHING_STRATEGY
    "retrieveMatchingStrategyClause"
retrieveReconcileStrategyClause = "QUOTED_STRING"
retrieveMatchingStrategyClause = "QUOTED_STRING"

```

Examples

```

OMBSYNCHRONIZE EXTERNAL_TABLE 'et1' TO EXTERNAL_TABLE 'et2'
USE (RECONCILE_STRATEGY 'MERGE', MATCHING_STRATEGY 'MATCH_BY_
OBJECT_NAME')

```

```

OMBSYNCHRONIZE MAPPING 'm1' OPERATOR 'o1' TO TABLE 't1'
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY
'MATCH_BY_OBJECT_NAME')

```

```
OMBSYNCHRONIZE MAPPING 'm1' OPERATOR 'o1' TO FLAT_FILE 'f1'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY  
'MATCH_BY_OBJECT_NAME')
```

```
OMBSYNCHRONIZE MAPPING 'm1' OPERATOR 'o1' TO MAPPING 'm1'  
OPERATOR 'o2'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY 'MATCH_BY_  
OBJECT_ID')
```

```
OMBSYNCHRONIZE ITEM_FOLDER 'F1'
```

```
OMBSYNCHRONIZE TABLE 't1' TO DATA_AUDITOR 'a1' OPERATOR 'o2'  
USE (RECONCILE_STRATEGY 'REPLACE', MATCHING_STRATEGY 'MATCH_BY_  
OBJECT_ID')
```

For Item Folder synchronization the source is obtained from the repository.

For simple item folders the source

is the table that the item folder is based on. For complex item folders

the source is the dependent folders.

The user does not have to specify the source.

OMBTRANSLATE EXTRACT

Purpose

Extracts business names and descriptions from a metadata loader file to an XLIFF translation file.

Prerequisites

None.

Syntax

```
mdlExtractCommand = ( EXTRACT [ FROM ] MDL_FILE "QUOTED_STRING" [ TO ]
    "extractTranslateFileClause" [ "setExtractOptions" ] [
    "componentsClause" ] [ "extractControlFileClause" ] [
    "extractOutputLogClause" ] )
extractTranslateFileClause = TRANSLATE_FILE "QUOTED_STRING"
setExtractOptions = SET OPTIONS "(" "extractOptionNameList" ")" VALUES "("
    "extractOptionValueList" ")"
componentsClause = COMPONENTS "(" "componentsList" ")"
extractControlFileClause = CONTROL_FILE "QUOTED_STRING"
extractOutputLogClause = OUTPUT LOG [ TO ] "QUOTED_STRING"
extractOptionNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
extractOptionValueList = "extractOptionValue" { ", " "extractOptionValue" }
componentsList = "objectTypeValue" "QUOTED_STRING" { ", " "objectTypeValue"
    "QUOTED_STRING" }
extractOptionValue = ( "QUOTED_STRING" )
objectTypeValue = ( PROJECT | ORACLE_MODULE | TABLE | VIEW | SEQUENCE |
    MATERIALIZED_VIEW | FUNCTION | PROCEDURE | PACKAGE | DIMENSION | CUBE
    | ADVANCED_QUEUE | STREAMS_QUEUE | MAPPING | REAL_TIME_MAPPING |
    PROCESS_FLOW_MODULE | PROCESS_FLOW_PACKAGE | PROCESS_FLOW | SAP_MODULE
    | CMI_MODULE | GATEWAY_MODULE | EXTERNAL_TABLE | FLAT_FILE_MODULE |
    FLAT_FILE | BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE
    | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL |
    ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
    PRESENTATION_TEMPLATE | LOCATION | CONNECTOR | CONTROL_CENTER |
    CONFIGURATION | COLLECTION | SNAPSHOT | ROLE | USER | ICONSET |
    TRANSFORMATION_MODULE | CALENDAR_MODULE | CALENDAR_FOLDER | CALENDAR |
    EXPERT_MODULE | EXPERT | DATA_RULE_MODULE | DATA_RULE | DATA_AUDITOR
    | STREAMS_CAPTURE_PROCESS | QUEUE_TABLE | QUEUE_PROPAGATION |
    OBJECT_TYPE | NESTED_TABLE | VARYING_ARRAY | DEPLOYMENT | DATA_PROFILE
    | PROFILE_REFERENCE | PLSQL_TABLE_TYPE | PLSQL_RECORD_TYPE |
    PLSQL_REF_CURSOR_TYPE | PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER |
    CMI_DEFINITION | ACTIVITY_TEMPLATE | ACTIVITY_TEMPLATE_FOLDER |
    TRANSPORTABLE_MODULE )
```

Keywords And Parameters

mdlExtractCommand

Extract business names and descriptions from a metadata loader file to an XLIFF translation file.

QUOTED_STRING

Enclose the name of the source metadata loader file in single quotes.

extractTranslateFileClause

Specify the name of the translation file created by extract.

QUOTED_STRING

Enclose the name of the translation file to be created in single quotes.

setExtractOptions

Set options to be used for extract.

componentsClause

List components to be merged.

extractControlFileClause

Specify a control file with options not directly supported by OMBTRANSLATE EXTRACT command.

QUOTED_STRING

Enclose the control file name in single quotes.

extractOutputLogClause

Extract log file for messages and statistics.

QUOTED_STRING

Enclose the log file name in single quotes.

extractOptionNameList

Comma separated list of extract option names (for example TARGETEQUALSSOURCE, EXTRACTNULLS).

UNQUOTED_STRING

Valid option names are TARGETEQUALSSOURCE, EXTRACTNULLS, MAPPINGDETAILS, DTD.

extractOptionValueList

Comma separated list of extract option values (for example 'Y', 'Y').

componentsList

Comma separated list of components to be merged.

QUOTED_STRING

Name of an object (for example 'TABLE_Y').

extractOptionValue

Value for an option in list (for example 'Y').

QUOTED_STRING

Enclose the option value in single quotes.

objectTypeValue

Object type to be merged (for example TABLE, VIEW and so on).

Examples

```
OMBTRANSLATE EXTRACT FROM MDL_FILE 'd:/mdl/exp1.mdl' TO  
TRANSLATE_FILE
```

```
'd:/mdl/exp1.xlf' OUTPUT LOG TO 'd:/mdl/exp1_extract.log'
```

```
OMBTRANSLATE EXTRACT FROM MDL_FILE 'd:/mdl/exp1.mdl' TO  
TRANSLATE_FILE
```

```
'd:/mdl/exp1.xlf' SET OPTIONS(TARGETEQUALSSOURCE, EXTRACTNULLS)  
VALUES
```

```
("Y", "Y")
```

See Also

OMBTRANSLATE MERGE

OMBTRANSLATE MERGE

Purpose

Merges translated business names and descriptions from an XLIFF translation file into a metadata loader file.

Prerequisites

None.

Syntax

```
mdlMergeCommand = ( MERGE [ FROM ] MDL_FILE "QUOTED_STRING"
    "mergeTranslateFileClause" [ TO ] "mergedMDLFileClause" [
    "setMergeOptions" ] [ "componentsClause" ] [ "mergeControlFileClause"
    ] [ "mergeOutputLogClause" ] )
mergeTranslateFileClause = TRANSLATE_FILE "QUOTED_STRING"
mergedMDLFileClause = MERGED_MDL_FILE "QUOTED_STRING"
setMergeOptions = SET OPTIONS "(" "extractOptionNameList" ")" VALUES "("
    "extractOptionValueList" ")"
componentsClause = COMPONENTS "(" "componentsList" ")"
mergeControlFileClause = CONTROL_FILE "QUOTED_STRING"
mergeOutputLogClause = OUTPUT LOG [ TO ] "QUOTED_STRING"
extractOptionNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
extractOptionValueList = "extractOptionValue" { ", " "extractOptionValue" }
componentsList = "objectTypeValue" "QUOTED_STRING" { ", " "objectTypeValue"
    "QUOTED_STRING" }
extractOptionValue = ( "QUOTED_STRING" )
objectTypeValue = ( PROJECT | ORACLE_MODULE | TABLE | VIEW | SEQUENCE |
    MATERIALIZED_VIEW | FUNCTION | PROCEDURE | PACKAGE | DIMENSION | CUBE
    | ADVANCED_QUEUE | STREAMS_QUEUE | MAPPING | REAL_TIME_MAPPING |
    PROCESS_FLOW_MODULE | PROCESS_FLOW_PACKAGE | PROCESS_FLOW | SAP_MODULE
    | CMI_MODULE | GATEWAY_MODULE | EXTERNAL_TABLE | FLAT_FILE_MODULE |
    FLAT_FILE | BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE
    | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL |
    ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
    PRESENTATION_TEMPLATE | LOCATION | CONNECTOR | CONTROL_CENTER |
    CONFIGURATION | COLLECTION | SNAPSHOT | ROLE | USER | ICONSET |
    TRANSFORMATION_MODULE | CALENDAR_MODULE | CALENDAR_FOLDER | CALENDAR |
    EXPERT_MODULE | EXPERT | DATA_RULE_MODULE | DATA_RULE | DATA_AUDITOR
    | STREAMS_CAPTURE_PROCESS | QUEUE_TABLE | QUEUE_PROPAGATION |
    OBJECT_TYPE | NESTED_TABLE | VARYING_ARRAY | DEPLOYMENT | DATA_PROFILE
    | PROFILE_REFERENCE | PLSQL_TABLE_TYPE | PLSQL_RECORD_TYPE |
    PLSQL_REF_CURSOR_TYPE | PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER |
    CMI_DEFINITION | ACTIVITY_TEMPLATE | ACTIVITY_TEMPLATE_FOLDER |
    TRANSPORTABLE_MODULE )
```

Keywords And Parameters

mdlMergeCommand

Merge business names and descriptions from an XLIFF translation file into a metadata loader file.

QUOTED_STRING

Enclose the name of the source metadata loader file in single quotes.

mergeTranslateFileClause

Specify the name of the source translation file.

QUOTED_STRING

Enclose the name of the source translation file in single quotes.

mergedMDLFileClause

Specify the name of the merged metadata loader file which will contain the translation information.

QUOTED_STRING

Enclose the name of the merged metadata loader file in single quotes.

setMergeOptions

Set options to be used for merge.

componentsClause

List components to be merged.

mergeControlFileClause

Specify a control file with merge options not directly supported by OMBTRANSLATE EXTRACT command.

QUOTED_STRING

Enclose the control file name in single quotes.

mergeOutputLogClause

Merge log file for merge messages and statistics.

QUOTED_STRING

Enclose the log file name in single quotes.

extractOptionNameList

Comma separated list of extract option names (for example TARGETEQUALSSOURCE, EXTRACTNULLS).

UNQUOTED_STRING

Valid option names are TARGETEQUALSSOURCE, EXTRACTNULLS, MAPPINGDETAILS, DTD.

extractOptionValueList

Comma separated list of extract option values (for example 'Y', 'Y').

componentsList

Comma separated list of components to be merged.

QUOTED_STRING

Name of an object (for example 'TABLE_Y').

extractOptionValue

Value for an option in list (for example 'Y').

QUOTED_STRING

Enclose the option value in single quotes.

objectTypeValue

Object type to be merged (for example TABLE, VIEW and so on).

Examples

```
OMBTRANSLATE MERGE FROM MDL_FILE 'd:/mdl/exp1.mdl' TRANSLATE_FILE  
'd:/mdl/exp1_de.xlf' TO MERGED_MDL_FILE 'd:/mdl/exp1_de.mdl' OUTPUT LOG  
TO  
'd:/mdl/exp1_de_merge.log'
```

```
OMBTRANSLATE MERGE FROM MDL_FILE 'd:/mdl/exp1.mdl' TRANSLATE_FILE  
'd:/mdl/exp1_de.xlf' TO MERGED_MDL_FILE 'd:/mdl/exp1_de.mdl'
```

See Also

OMBTRANSLATE EXTRACT

OMBUNDEFINE ASSOCIATION_DEFINITION

Purpose

To undefine an association between two classes (types).

Prerequisites

Association definition to be undefined should already exist. This command can be executed for any association definition regardless of current context. Only user defined associations can be undefined. User must have CREATE_EXTENSIONMODEL system privilege and has to be connected in single user mode to run this command.

Syntax

```
parseUndefineAssociationCommand = OMBUNDEFINE ASSOCIATION_DEFINITION  
    "QUOTED_STRING"
```

Keywords And Parameters

parseUndefineAssociationCommand
Undefine an association.

Examples

```
OMBUNDEFINE ASSOCIATION_DEFINITION 'UD_ASSOC3'
```

This will undefine association 'UD_ASSOC3'.

See Also

OMBDEFINE ASSOCIATION_DEFINITION, OMBDESCRIBE ASSOCIATION_DEFINITION

OMBUNDEFINE CLASS_DEFINITION

Purpose

To undefine a class.

Prerequisites

Class definition to be undefined should already exist. This command can be executed for any class definition regardless of current context. Only user defined classes can be undefined. User must have CREATE_EXTENSIONMODEL system privilege and has to be connected in single user mode to run this command.

Syntax

```
parseUndefineClassCommand = OMBUNDEFINE CLASS_DEFINITION "QUOTED_STRING" [  
    CASCADE ]
```

Keywords And Parameters

parseUndefineClassCommand

Undefine a class.

Examples

```
OMBUNDEFINE CLASS_DEFINITION 'UD_MY_CLASS'
```

This will undefine class 'UD_MY_CLASS'.

See Also

OMBDEFINE CLASS_DEFINITION, OMBDESCRIBE CLASS_DEFINITION

OMBUNLOCK

Purpose

Unlock one or more objects, previously locked by OMBLOCK command. Note that if the object(s) have been modified, save or revert is also required in order for the lock(s) to be released.

Prerequisites

The object(s) should have been locked previously using OMBLOCK command.

Syntax

```

parseUnLockCommand = OMBUNLOCK "parseTypeNameList"
parseTypeNameList = "objectType" "QUOTED_STRING" { "," "objectType"
"QUOTED_STRING" }
objectType = ( ADVANCED_QUEUE | STREAMS_QUEUE | BUSINESS_AREA | COLLECTION
| CONNECTOR | CONFIGURATION | DEPLOYMENT | CUBE | DIMENSION | EXPERT
| EXPERT_MODULE | EXTERNAL_TABLE | FLAT_FILE_MODULE | FLAT_FILE |
FUNCTION | TABLE_FUNCTION | GATEWAY_MODULE |
BUSINESS_DEFINITION_MODULE | REGISTERED_FUNCTION | LOCATION | MAPPING
| MATERIALIZED_VIEW | OBJECT_TYPE | VARYING_ARRAY | NESTED_TABLE |
ORACLE_MODULE | PACKAGE | PROCEDURE | PROCESS_FLOW |
PROCESS_FLOW_MODULE | PROCESS_FLOW_PACKAGE | PROJECT | ITEM_FOLDER |
DRILL_PATH | LIST_OF_VALUES | DRILL_TO_DETAIL | ALTERNATIVE_SORT_ORDER
| PRESENTATION_TEMPLATE | BUSINESS_PRESENTATION_MODULE |
CONTROL_CENTER | SAP_MODULE | CMI_MODULE | SEQUENCE | TABLE | VIEW |
CMI_DEFINITION | PLSQL_RECORD_TYPE | "UNQUOTED_STRING" )

```

Keywords And Parameters

`parseUnLockCommand`

Specify unlock command.

`parseTypeNameList`

Specify the object or the list of objects to be unlocked.

`QUOTED_STRING`

Name of the object to be unlocked. Can be specified as an absolute path or as a path relative to the current context. However, there is the restriction that all objects to be unlocked must be in the current project.

`objectType`

Type of the object to be unlocked.

Examples

OMBUNLOCK TABLE 'T1', VIEW '/MY_PROJECT/ORACLE_1/V1'
will unlock table 'T1' in the current module, and view 'V1' in Oracle

module 'ORACLE_1' from project 'MY_PROJECT'.

See Also

OMBLOCK

OMBUNREGISTER CONTROL_CENTER

Purpose

Un-register a control center.

Prerequisites

Must be logged on as the owner of the repository and connected in a single user mode.

Syntax

```
unRegisterControlCenterCommand = OMBUNREGISTER CONTROL_CENTER  
    "QUOTED_STRING"
```

Examples

```
OMBUNREGISTER CONTROL_CENTER 'MY_CONTROL_CENTER'
```

OMBUNREGISTER LOCATION

Purpose

Un-register a location with a Control Center.

Prerequisites

Must be in the context of a project and connected to a Control Center.

Syntax

```
unRegisterLocationCommand = OMBUNREGISTER LOCATION "QUOTED_STRING"
```

Examples

```
OMBUNREGISTER LOCATION 'MY_ORACLE_LOCATION'
```

OMBUNREGISTER USER

Purpose

To unregister a specified Warehouse Builder user. This will not drop the user from database.

Prerequisites

A Warehouse Builder user can be unregistered from any context.

Syntax

```
parseUnregisterUserCommand = ( OMBUNREGISTER USER "QUOTED_STRING" [
    IDENTIFIED BY "QUOTED_STRING" ] )
```

Keywords And Parameters

parseUnregisterUserCommand

This clause unregister a specified Warehouse user.

Examples

```
OMBUNREGISTER USER 'USER1'
```

will unregister user 'USER1'.

```
OMBUNREGISTER USER 'USER1' IDENTIFIED BY 'passwdOfUser1'
```

will unregistered user 'USER1'. The password provided from IDENTIFIED BY clause will be used to clean up the target schema when unregister it if USER1 is a target schema. If IDENTIFIED BY clause is ommited, Oracle Warehouse Builder will try to retrieve this information from related location if available, otherwise there will have some Oracle Warehouse Builder objects left in the target schema. The IDENTIFIED BY clause is not necessary if the user USER1 is not used as a target schema.

See Also

OMBREGISTER USER, OMBALTER USER, OMBRETRIEVE USER

OMBVALIDATE

Purpose

This command validates an repository object. The results are generated in a file in a user defined directory.

Prerequisites

In the context of a Oracle Module except when validating Project and Oracle Module. To validate a Project the user needs to be in the Root context. To validate Oracle Module the user needs to be in Project context.

Syntax

```
parseValidateCommand = OMBVALIDATE ( ( EXPERT | EXPERT_MODULE | TABLE |
VIEW | SEQUENCE | ORACLE_MODULE | COLLECTION | MATERIALIZED_VIEW |
TRANSPORTABLE_MODULE | BUSINESS_PRESENTATION_MODULE |
BUSINESS_DEFINITION_MODULE | DIMENSION | CUBE | DATA_AUDITOR | MAPPING
| REAL_TIME_MAPPING | PROJECT | OBJECT_TYPE | VARYING_ARRAY |
NESTED_TABLE | PACKAGE | FUNCTION | PROCEDURE | TABLE_FUNCTION |
EXTERNAL_TABLE | FLAT_FILE | FLAT_FILE_MODULE | PRESENTATION_TEMPLATE
| ALTERNATIVE_SORT_ORDER | LIST_OF_VALUES | DRILL_TO_DETAIL |
BUSINESS_AREA | DRILL_PATH | ITEM_FOLDER | REGISTERED_FUNCTION |
LOCATION | CONNECTOR | CONTROL_CENTER | CONFIGURATION | DEPLOYMENT |
PROCESS_FLOW_MODULE | PROCESS_FLOW | PROCESS_FLOW_PACKAGE |
ADVANCED_QUEUE | STREAMS_QUEUE | QUEUE_TABLE | QUEUE_PROPAGATION |
STREAMS_CAPTURE_PROCESS | PLSQL_RECORD_TYPE | PLSQL_TABLE_TYPE |
PLSQL_REF_CURSOR_TYPE | CALENDAR | CALENDAR_MODULE ) "QUOTED_STRING" [
"getOutputValidationResults" ] )
getOutputValidationResults = OUTPUT [ VALIDATION_RESULT ] TO (
"QUOTED_STRING" | ( FILE "QUOTED_STRING" ) ) WRITE ( ( "(" ( ( SUCCESS
| WARNING | ERROR ) [ "," ] )+ ")" ) | ALL | SUCCESS | WARNING |
ERROR )
```

Keywords And Parameters

`parseValidateCommand`

This command validates a repository object.

`QUOTED_STRING`

The name of the object.

`getOutputValidationResults`

This clause outputs the validation results to one or more files in the specified folder.

`QUOTED_STRING`

A directory where validation results are stored.

Examples

```
OMBVALIDATE TABLE 'T1' OUTPUT VALIDATION_RESULT TO '/tmp' WRITE  
(SUCCESS,  
ERROR)
```

See Also

OMBCOMPILE

OMBALTER to OMBALTER EXTERNAL_ TABLE

This chapter lists commands associated with OMBALTER in alphabetical order, concluding with the command OMBALTER EXTERNAL_TABLE. Subsequent commands associated with OMBALTER are contained in the next chapter.

OMBALTER

Purpose

Alter metadata for a component.

Prerequisites

Should be in the parent context of the component to alter.

Syntax

```
alterCommand = OMBALTER "fco_type" "fco_name" "alterCommandSubClauses"
alterCommandSubClauses = ( "renameClause" [ "setPropertiesClause" ] {
    "setReferenceClause" | "unsetReferenceClause" } {
    "addSCOClausesForAlter" | "modifySCOClauses" | "deleteSCOClauses" } ) | (
    "setPropertiesClause" { "setReferenceClause" | "unsetReferenceClause"
    } { "addSCOClausesForAlter" | "modifySCOClauses" | "deleteSCOClauses" }
    ) | ( ( "setReferenceClause" | "unsetReferenceClause" )+ {
    "addSCOClausesForAlter" | "modifySCOClauses" | "deleteSCOClauses" } ) | (
    "addSCOClausesForAlter" | "modifySCOClauses" | "deleteSCOClauses" )+
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = SET PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = SET ( REF | REFERENCE ) [ "qualifier" ] "type"
    "quotedNameList" [ { "parentSCOClauses" } OF "fco_type" "fco_name" ]
unsetReferenceClause = UNSET ( REF | REFERENCE ) [ "qualifier" ] "type"
addSCOClausesForAlter = ADD "sco_type" "sco_name" { "parentSCOClauses" } [
    AT POSITION "pos" ] [ "setPropertiesClause" ] { "setReferenceClause" }
modifySCOClauses = MODIFY "sco_type" "sco_name" { "parentSCOClauses" }
    "modifySCOClauses"
deleteSCOClauses = DELETE "sco_type" "sco_name" { "parentSCOClauses" }
propertyNameList = "propertyName" { ", " "propertyName" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
quotedNameList = "QUOTED_STRING" | "(" "QUOTED_STRING" { ", "
    "QUOTED_STRING" } ")"
parentSCOClauses = OF "sco_type" "sco_name"
modifySCOClauses = ( "renameClause" [ "moveToClause" ] [
    "setPropertiesClause" ] { "setReferenceClause" } ) | ( "moveToClause"
    [ "setPropertiesClause" ] { "setReferenceClause" } ) | (
    "setPropertiesClause" { "setReferenceClause" } ) |
    "setReferenceClause"+
propertyName = "UNQUOTED_STRING"
propertyValue = "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL"
moveToClause = MOVE TO POSITION "pos"
```

Keywords And Parameters

alterCommand

Specify the component to alter.

fco_type

The type of the component.

fco_name

The physical name of the component in single quotes.

alterCommandSubClauses

Use this clause to rename the component, reset its properties, or modify the child objects of the component.

renameClause

Rename the component.

setPropertyClause

Set object properties.

setReferenceClause

Specify reference objects.

qualifier

Specify which reference to set, if there are more than one pointing to the same type.

unsetReferenceClause

Removes an existing reference.

qualifier

Specify which reference to set, if there are more than one pointing to the same type.

addSCOClausesForAlter

Add child objects under the component.

modifySCOClauses

Change definition of a child object.

deleteSCOClauses

Delete a child object.

propertyNameList

A list of property names.

propertyValueList

A list of property values.

quotedNameList

A list of single-quoted physical names.

parentSCOClause

Used to specify the path from a child object to the component

modifySCOSubClauses

Use this clause to rename a child object, reset its properties or references, or reorder it.

propertyName

An unquoted string representing the name of a property.

propertyValue

The value of a property.

moveToClause

Used to reorder child objects.

Examples

This is an example for altering a table:

```
OMBALTER TABLE 'T1' SET PROPERTIES (DESCRIPTION) VALUES ('My First  
Table')
```

The following statement alters the column of a view:

```
OMBALTER VIEW 'V1'  
MODIFY COLUMN 'COL1'  
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2')
```

See Also

OMBCREATE, OMBDROP

OMBALTER ACTIVITY_TEMPLATE

Purpose

To alter activity template folder.

Prerequisites

Should be in the context of an Activity Template Folder.

Syntax

```

parseAlterTemplate = "QUOTED_STRING" ( ( "renameClause" [
    "alterTemplatePropertiesOrIconSetClause" ] [ "modifyParametersClause"
] ) | ( "alterTemplatePropertiesOrIconSetClause" [
    "modifyParametersClause" ] ) | "modifyParametersClause" )
renameClause = RENAME TO "QUOTED_STRING"
alterTemplatePropertiesOrIconSetClause = ( ( SET "setPropertiesClause" ) |
    "setReferenceIconSetClause" | "unsetReferenceIconSetClause" )+
modifyParametersClause = ( "addParameterClause" | "modifyParameterClause"
    | "deleteParameterClause" )+
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = SET ( REFERENCE | REF ) ICONSET
    "QUOTED_STRING"
unsetReferenceIconSetClause = UNSET ( REFERENCE | REF ) ICONSET
addParameterClause = ( ADD PARAMETER "QUOTED_STRING" ) [ SET
    "setPropertiesClause" ]
modifyParameterClause = MODIFY PARAMETER "QUOTED_STRING" ( (
    "renameClause" [ SET "setPropertiesClause" ] ) | ( SET
    "setPropertiesClause" ) )
deleteParameterClause = DELETE PARAMETER "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

setPropertiesClause

Basic properties for ACTIVITY_TEMPLATE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Activity Template

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Activity Template

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
alterActivityTemplateCommandExampleTag??
```

See Also

OMBALTER, OMBCREATE ACTIVITY_TEMPLATE, OMBDROP ACTIVITY_TEMPLATE

OMBALTER ACTIVITY_TEMPLATE_FOLDER

Purpose

To alter activity template folder.

Prerequisites

Should be in the context of a Project.

Syntax

```

parseAlterTemplateFolder = "QUOTED_STRING" ( ( RENAME TO "QUOTED_STRING" [
    "alterTemplateFolderPropertiesOrIconSetClause" ] ) |
    "alterTemplateFolderPropertiesOrIconSetClause" )
alterTemplateFolderPropertiesOrIconSetClause = ( ( SET
    "setPropertyClause" ) | "setReferenceIconSetClause" |
    "unsetReferenceIconSetClause" )+
setPropertyClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = SET ( REFERENCE | REF ) ICONSET
    "QUOTED_STRING"
unsetReferenceIconSetClause = UNSET ( REFERENCE | REF ) ICONSET
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

setPropertyClause

Basic properties for ACTIVITY_TEMPLATE_FOLDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Activity Template Folder

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Activity Template Folder

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

Examples

```
OMBALTER ACTIVITY_TEMPLATE_FOLDER 'FOLDER1' SET PROPERTIES (NAME)
VALUES
('FOLDER_RENAME')
```

See Also

OMBALTER, OMBCREATE ACTIVITY_TEMPLATE_FOLDER, OMBDROP
ACTIVITY_TEMPLATE_FOLDER

OMBALTER ADVANCED_QUEUE

Purpose

Alter the Advanced Queue by resetting its properties.

Prerequisites

Should be in the context of an Oracle Module. The Queue Table should exist in the same Oracle Module.

Syntax

```
alterAQCommand = OMBALTER ( ADVANCED_QUEUE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterAQCommand

Alters the Advanced Queue with the given name by either renaming it or by setting it's properties or both.

renameClause

Renames the Advanced Queue to the given name.

alterPropertiesOrIconSetClause

Set properties or change the Icon Set.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Advanced Queue.

Valid properties are as shown:

Basic properties for ADVANCED_QUEUE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Advanced Queue

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Advanced Queue

Name: QTABLE

Type: STRING(4000)

Valid Values: N/A

Default: "

Queue Table for the Advanced Queue. This has to be the name of a Queue Table(Queue_Table) existing in the same Oracle Module.

Properties for ADVANCED_QUEUE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DEQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Dequeue Enabled for AQ

Name: ENQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enqueue enabled for AQ

Name: GENERATE_ADVANCED_QUEUE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the Advanced Queue.

Name: GENERATE_TEMPORARY_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the temporary table.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MAX_RETRIES

Type: NUMBER

Valid Values: N/A

Default: 5

Max. number of Retries

Name: RETENTION_TIME

Type: NUMBER

Valid Values: N/A

Default: 0

Message retention time

Name: RETRY_DELAY

Type: NUMBER

Valid Values: N/A

Default: 0

Delay period before retry

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Set the specified Icon Set.

unsetReferenceIconSetClause

Unset the Icon Set.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER ADVANCED_QUEUE 'SOME_ADVANCED_QUEUE' SET PROPERTIES  
(MAX_RETRIES,  
RETRY_DELAY, RETENTION_TIME, ENQUEUE_ENABLED, DEQUEUE_ENABLED)  
VALUES  
(10,20,60,'true','false')
```

This will set its properties as specified.

See Also

OMBALTER, OMBCREATE ADVANCED_QUEUE, OMBRETRIEVE ADVANCED_QUEUE, OMBDROP ADVANCED_QUEUE

OMBALTER ALTERNATIVE_SORT_ORDER

Purpose

Alters an alternative sort order.

Prerequisites

Should be in the context of a business definition module or use the full path.

Syntax

```
alterAlternativeSortOrderCommand = ( OMBALTER ALTERNATIVE_SORT_ORDER
  "QUOTED_STRING" ( ( "renameClause" [ SET
    "setPropertyClauseforLOVandD2D" ] [ SET "setReferenceIconSetClause"
  ] [ UNSET "unsetReferenceIconSetClause" ] {
    "alterAlternativeSortOrderClauses" } ) | ( SET
    "setPropertyClauseforLOVandD2D" [ SET "setReferenceIconSetClause" ]
  [ UNSET "unsetReferenceIconSetClause" ] {
    "alterAlternativeSortOrderClauses" } ) | ( SET
    "setReferenceIconSetClause" [ UNSET "unsetReferenceIconSetClause" ] {
    "alterAlternativeSortOrderClauses" } ) | ( UNSET
    "unsetReferenceIconSetClause" { "alterAlternativeSortOrderClauses" } )
  | ( "alterAlternativeSortOrderClauses" {
    "alterAlternativeSortOrderClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertyClauseforLOVandD2D = PROPERTIES "("
  "propertyNameListforLOVandD2D" ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterAlternativeSortOrderClauses = SET ( REF | REFERENCE )
  "addAlternativeSortOrderClauseforAlter" | UNSET ( REF | REFERENCE )
  "deleteAlternativeSortOrderClauses" |
  "addAlternativeSortOrderReferenceClause"
propertyNameListforLOVandD2D = ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) {
  ", " ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) }
propertyValueList = "propertyValue" { ", " "propertyValue" }
addAlternativeSortOrderClauseforAlter = ( DEFINING ITEM "QUOTED_STRING" OF
  ITEM_FOLDER "QUOTED_STRING" ) | ( ORDERED ITEM "QUOTED_STRING" OF
  ITEM_FOLDER "QUOTED_STRING" )
deleteAlternativeSortOrderClauses = ( DEFINING ITEM ) | ( ORDERED ITEM ) |
  ( ITEM "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING" )
addAlternativeSortOrderReferenceClause = SET ( REF | REFERENCE ) ITEM
  "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`alterAlternativeSortOrderCommand`

This clause alters an alternative sort order.

`QUOTED_STRING`

name of the alternative sort order.

renameClause

Renames an alternative sort order with a different name.

setPropertyClauseforLOVandD2D

This clause sets the properties of the object.

Basic properties for ALTERNATIVE_SORT_ORDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the alternative sort order

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the alternative sort order

Name: DRILL_TO_DETAIL

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the alternative sort order enables drilling between the item folders containing the items that use the alternative sort order

Name: RETRIEVE_VALUES_GROUP_SIZE

Type: Number

Valid Values: N/A

Default: '100'

The number of rows to be fetched from the database at a time

Name: CACHE_VALUES

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the list of values should be cached in memory

Name: REQUIRE_SEARCH

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether search criteria should be requested

Name: SHOW_IN_NAVIGATOR

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be shown in the item navigator

Name: SORTED_DISTINCT

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be displayed sorted with duplicates hidden

Properties for ALTERNATIVE_SORT_ORDER:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterAlternativeSortOrderClauses

This clause modifies an alternative sort order.

propertyNameListforLOVandD2D

This is the list of property names.

propertyValueList

This is the list of property values.

addAlternativeSortOrderClauseforAlter

This clause sets the defining item or ordering item for the alternative sort order.

DEFINING

This sets the defining item for the alternative sort order.

ORDERED

This sets the ordered item for the alternative sort order.

deleteAlternativeSortOrderClauses

This deletes specific item references from an alternative sort order.

DEFINING

This deletes the defining item from an alternative sort order.

ITEM

This deletes a reference to an item from an alternative sort order.

ORDERED

This deletes the ordered item from an alternative sort order.

addAlternativeSortOrderReferenceClause

This adds a reference to an item to an alternative sort order.

propertyValue

This is a property value.

Examples

```
OMBALTER ALTERNATIVE_SORT_ORDER 'AS' SET PROPERTIES (DESCRIPTION)
VALUES
('AS')
```

See Also

```
OMBCREATE ALTERNATIVE_SORT_ORDER, OMBRETRIEVE ALTERNATIVE_
SORT_ORDER
```

OMBALTER ANALYZE_ACTION_PLAN

Purpose

To alter an existing profile action plan.

Prerequisites

alterAnalyzeActionPlanPreTag??

Syntax

```
AlterActionPlanCommand = ( OMBALTER ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ( (
    "renameActionPlanClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "addActionClause" { "addActionClause" |
    "modifyActionClause" | "deleteActionClause" } ) | (
    "modifyActionClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "deleteActionClause" { "addActionClause"
    | "modifyActionClause" | "deleteActionClause" } ) ) )
renameActionPlanClause = RENAME TO "QUOTED_STRING"
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
modifyActionClause = MODIFY ACTION "QUOTED_STRING" ( (
    "renameActionClause" [ "setUnsetClause" ] ) | "setUnsetClause" )
deleteActionClause = DELETE ACTION "QUOTED_STRING"
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
    "setReferenceClause"
renameActionClause = RENAME TO "QUOTED_STRING"
setUnsetClause = ( ( SET "setClauseForAlter" ) | ( UNSET
    "unsetReferenceClause" ) )
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
    "useClause" }
setClauseForAlter = ( "propertiesClause" [ SET "setReferenceClause" |
    UNSET "unsetReferenceClause" ] ) | "setReferenceClause"
unsetReferenceClause = ( REF | REFERENCE )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" ) )
```

Keywords And Parameters

AlterActionPlanCommand

Alter an action plan.

QUOTED_STRING

Action plan name.

renameActionPlanClause

Rename an action plan.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Action name.

modifyActionClause

Modify an action of an action plan.

QUOTED_STRING

Action name.

deleteActionClause

Delete an action of an action plan.

QUOTED_STRING

Action name.

setClause

Set the properties of an action, or associate an object with an action, or both.

renameActionClause

Rename an action of the action plan.

setUnsetClause

Set the properties, associate/disassociate an object with an action, or both.

propertiesClause

Set the properties, or associate/disassociate an object with an action, or both.

setReferenceClause

Associate an object with an action.

ObjType

Object type. The only valid value is DATA_PROFILE_TABLE.

setClauseForAlter

Set the properties, or associate/disassociate an object with an action, or both.

unsetReferenceClause

Disassociate a previously associated object from an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Currently used only in CORRECTION_SCHEMA_ACTION_PLAN.

propertyValue

Value of a property.

Examples

```
OMBALTER ANALYZE_ACTION_PLAN 'ANALYZE_PLAN'  
ADD ACTION 'A1' SET REF PROFILE_REFERENCE 'EMP'
```

See Also

OMBCREATE ANALYZE_ACTION_PLAN, OMBPROFILE

OMBALTER BUSINESS_AREA

Purpose

Alters a Business Area.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
alterBusinessAreaCommand = ( OMBALTER BUSINESS_AREA "QUOTED_STRING" ( (
    "renameClause" [ SET "setPropertiesClause" ] [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterBusinessAreaClauses" } ) ) | ( SET "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterBusinessAreaClauses" } ) ) | ( SET "setReferenceIconSetClause" [
    UNSET "unsetReferenceIconSetClause" ] { "alterBusinessAreaClauses" }
    ) | ( UNSET "unsetReferenceIconSetClause" { "alterBusinessAreaClauses"
    } ) ) | ( "alterBusinessAreaClauses" { "alterBusinessAreaClauses" } ) )
)
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterBusinessAreaClauses = UNSET ( REF | REFERENCE ) "("
    "unsetfolderNameList" ")" | SET ( REF | REFERENCE ) "("
    "setfolderNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
unsetfolderNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
setfolderNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterBusinessAreaCommand

This clause alters a Business Area.

QUOTED_STRING

name of the Business Area.

renameClause

Renames a Business Area with a different name.

setPropertiesClause

This clause sets the properties of the object.

Basic properties for BUSINESS_AREA:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the business area

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the business area

Properties for BUSINESS_AREA:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterBusinessAreaClauses

This clause adds, modifies or deletes business area shortcuts.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

unsetfolderNameList

Used to remove folder references for this business area.

setfolderNameList

Used to set item folder references for this business area.

propertyValue

This is a property value.

Examples

```
OMBALTER BUSINESS_AREA 'SALES' SET PROPERTIES (DESCRIPTION) VALUES  
(Sales  
area')
```

See Also

OMBCREATE BUSINESS_AREA, OMBRETRIEVE BUSINESS_AREA

OMBALTER BUSINESS_DEFINITION_MODULE

Purpose

Alter a business definition module by renaming it, or reset its properties, or both.

Prerequisites

Should be in the context of project.

Syntax

```

alterEULModuleCommand = OMBALTER ( BUSINESS_DEFINITION_MODULE
    "QUOTED_STRING" ( "renameClause" [
        "alterPropertiesOrReferenceClauseForDataOnlyModule" ] |
        "alterPropertiesOrReferenceClauseForDataOnlyModule" |
        "addOrRemoveOrModifyModuleReferenceLocationClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataOnlyModule = ( ( SET ( (
    "setPropertyClause" [ ( SET "setReferenceClauseForDataOnlyModule" [
    UNSET "unsetReferenceClauseForDataOnlyModule" ] ) | ( UNSET
    "unsetReferenceClauseForDataOnlyModule" [ SET
    "setReferenceClauseForDataOnlyModule" ] ) ] ) | (
    "setReferenceClauseForDataOnlyModule" [ UNSET
    "unsetReferenceClauseForDataOnlyModule" ] ) ) ) | ( UNSET
    "unsetReferenceClauseForDataOnlyModule" [ SET
    "setReferenceClauseForDataOnlyModule" ] ) ) [
    "addOrRemoveOrModifyModuleReferenceLocationClause" ]
addOrRemoveOrModifyModuleReferenceLocationClause = (
    "addReferenceLocationClause" | "removeReferenceLocationClause" |
    "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
    "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataOnlyModule = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
unsetReferenceClauseForDataOnlyModule = ( "unsetReferenceLocationClause" [
    UNSET "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause"
    )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" SET AS DEFAULT
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

alterEULModuleCommand

This command modifies an existing business definition module.

QUOTED_STRING

Name of the existing business definition module in single quotes.

renameClause

Rename a business definition module.

alterPropertiesOrReferenceClauseForDataOnlyModule

Alter existing business definition module's properties, or locations, or icon sets, or all.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the business definition module.

setPropertiesClause

Associate a set of properties with the existing business definition module.

Basic properties for BUSINESS_DEFINITION_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a business definition module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a business definition module

Properties for BUSINESS_DEFINITION_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Discoverer Location for Business Definition Module

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Name: MLS_DEPLOYMENT_LANGUAGE

Type: STRING

Valid Values: N/A

Default: MLS_BASE_LANGUAGE

MLS Language to be used for deployment

Name: OBJECT_MATCHING

Type: STRING

Valid Values: BY_IDENTIFIER, BY_NAME

Default: BY_IDENTIFIER

Whether import should match up objects by identifier or by name

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

setReferenceClauseForDataOnlyModule

Set location, or icon set, or both for the new business definition module.

unsetReferenceClauseForDataOnlyModule

Unset location, or icon set, or both for the business definition module.

addReferenceLocationClause

Add a runtime location to the new business definition module.

removeReferenceLocationClause

Remove a runtime location from the business definition module.

modifyReferenceLocationClause

Modify a runtime location of the business definition module.

propertyNameList

Comma-delimited list of property names. Property names are not in quotes.

propertyValueList

Comma-delimited list of property values.

setReferenceLocationClause

Set a location for the existing business definition module.

setReferenceIconSetClause

Set icon set for the new business definition module.

unsetReferenceLocationClause

Unset a location for the existing business definition module.

unsetReferenceIconSetClause

Unset icon set for the business definition module.

propertyValue

Value for a specified property.

Examples

```
OMBALTER BUSINESS_DEFINITION_MODULE 'src_module' RENAME TO 'Sales_
module'
```

```
SET PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a
Sales
```

```
module.', 'Sales module')
```

This will rename a business definition module "src_module" to

"Sales_module", and set its description to "This becomes a Sales module.",

set its business name to "Sales module".

See Also

OMBALTER, OMBCREATE BUSINESS_DEFINITION_MODULE, OMBDROP
BUSINESS_DEFINITION_MODULE

OMBALTER BUSINESS_PRESENTATION_MODULE

Purpose

Alter the presentation module by renaming it, or reset its properties, or both.

Prerequisites

Should be in the context of project.

Syntax

```
alterReportModuleCommand = OMBALTER ( BUSINESS_PRESENTATION_MODULE
    "QUOTED_STRING" ( "renameClause" [
        "alterPropertiesOrReferenceClauseForDataOnlyModule" ] |
        "alterPropertiesOrReferenceClauseForDataOnlyModule" |
        "addOrRemoveOrModifyModuleReferenceLocationClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataOnlyModule = ( ( SET ( (
    "setPropertiesClause" [ ( SET "setReferenceClauseForDataOnlyModule" [
        UNSET "unsetReferenceClauseForDataOnlyModule" ] ) | ( UNSET
        "unsetReferenceClauseForDataOnlyModule" [ SET
        "setReferenceClauseForDataOnlyModule" ] ) ) ] | (
        "setReferenceClauseForDataOnlyModule" [ UNSET
        "unsetReferenceClauseForDataOnlyModule" ] ) ) ) | ( UNSET
        "unsetReferenceClauseForDataOnlyModule" [ SET
        "setReferenceClauseForDataOnlyModule" ] ) ) [
        "addOrRemoveOrModifyModuleReferenceLocationClause" ]
addOrRemoveOrModifyModuleReferenceLocationClause = (
    "addReferenceLocationClause" | "removeReferenceLocationClause" |
    "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
    "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataOnlyModule = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
unsetReferenceClauseForDataOnlyModule = ( "unsetReferenceLocationClause" [
    UNSET "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause"
)
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" SET AS DEFAULT
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterReportModuleCommand

This command modifies an existing presentation module.

QUOTED_STRING

Name of the existing presentation module in single quotes.

renameClause

Rename a presentation module.

alterPropertiesOrReferenceClauseForDataOnlyModule

Alter existing business presentation module's properties, or locations or icon sets, or all the three.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the business presentation module.

setPropertyClause

Associate a set of properties with the existing presentation module.

Basic properties for BUSINESS_PRESENTATION_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a presentation module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a presentation module

Properties for BUSINESS_PRESENTATION_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

BI Beans Location for Business Presentation Module

Name: DEFAULT_CATALOG_FOLDER

Type: STRING

Valid Values: N/A

Default: "

Default Catalog Folder for deployed BI Beans presentations

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceClauseForDataOnlyModule

Set location and/or icon set for the new business presentation module.

unsetReferenceClauseForDataOnlyModule

Unset location and/or icon set for the business presentation module.

addReferenceLocationClause

Add a runtime location to the new business presentation module.

removeReferenceLocationClause

Remove a runtime location from the business presentation module.

modifyReferenceLocationClause

Modify a runtime location of the business presentation module.

propertyNameList

Comma-delimited list of property names. Property names are not in quotes.

propertyValueList

Comma-delimited list of property values.

setReferenceLocationClause

Set a location for the existing presentation module.

setReferenceIconSetClause

Set icon set for the new business presentation module.

unsetReferenceLocationClause

Unset a location for the existing presentation module.

unsetReferenceIconSetClause

Unset icon set for the business presentation module.

propertyValue

Value for a specified property.

Examples

```
OMBALTER BUSINESS_PRESENTATION_MODULE 'salesrep_module' RENAME TO  
'newsalesrep_module' SET PROPERTIES (DESCRIPTION, BUSINESS_NAME)  
VALUES
```

```
('This becomes a new sales module.', 'new sales module')
```

This will rename the presentation module "salesrep_module" to "newsalesrep_module", and set its description to "This becomes a new sales module.", set its business name to "new sales module".

See Also

OMBALTER, OMBCREATE BUSINESS_PRESENTATION_MODULE, OMBDROP BUSINESS_PRESENTATION_MODULE

OMBALTER CALENDAR

Purpose

To alter calendar.

Prerequisites

Should be in the context of a CalendarFolder.

Syntax

```

parseAlterCalendar = OMBALTER CALENDAR "QUOTED_STRING" ( ( "renameClause"
  [ "alterCalendarPropertiesOrIconSetClause" ] [ "modifySchedulesClause"
  ] ) | ( "alterCalendarPropertiesOrIconSetClause" [
  "modifySchedulesClause" ] ) | "modifySchedulesClause" )
renameClause = RENAME TO "QUOTED_STRING"
alterCalendarPropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [
  SET "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause"
  ] ) | "setReferenceIconSetClause" ) ) | UNSET
  "unsetReferenceIconSetClause"
modifySchedulesClause = ( "addScheduleClause" | "modifyScheduleClause" ) +
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
  "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addScheduleClause = ADD SCHEDULE
modifyScheduleClause = MODIFY SCHEDULE "QUOTED_STRING" ( SET
  "setPropertiesClause" )
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

setPropertiesClause

Basic properties for CALENDAR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Calendar

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Calendar

Each calendar contains a single schedule with the name 'LOCALWINDOW' which has the following properties which define the various aspects of the schedule

Basic properties for SCHEDULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Schedule

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Schedule

Basic properties for the owned SCHEDULE OBJECT :

Name: START_TIME

Type: STRING(4000)

Valid Values: Start time in the format specified by property DATE_FORMAT.

The value NULL can also be passed if a schedule that takes on a start time of ASAP, that is upon deployment.

Default: "

Start time for the schedule

Name: END_TIME

Type: STRING(4000)

Valid Values: End time in the format specified by property DATE_FORMAT. The value NULL can also be passed if a schedule that repeats forever is to be created..

Default: "

End time for the schedule

Name: TIMEZONE

Type: STRING(4000)

Valid Values: N/A

Default: "

Time zone which times refer to.

Name: REPEAT_EXPRESSION

Type: STRING(4000)

Valid Values: N/A

Default: "

iCal format of a repeat expression. If no REPEAT_EXPRESSION is supplied, the scheduled activity is only performed once. The REPEAT_EXPRESSION includes the following:

FREQ

This specifies the type of recurrence. It must be specified. The possible predefined frequency values are YEARLY, MONTHLY, WEEKLY, DAILY, HOURLY, MINUTELY, and SECONDLY. Alternatively, specifies an existing schedule to use as a user-defined frequency.

INTERVAL

This specifies a positive integer representing how often the recurrence repeats. The default is 1, which means every second for secondly, every day for daily, and so on. The maximum value is 999.

BYMONTH

This specifies which month or months you want the job to execute in. You can use numbers such as 1 for January and 3 for March, as well as three-letter abbreviations such as FEB for February and JUL for July.

BYWEEKNO

This specifies the week of the year as a number. byweekno is only valid for YEARLY.

BYYEARDAY

This specifies the day of the year as a number. Valid values are 1 to 366. An example is 69, which is March 10 (31 for January, 28 for February, and 10 for March). 69 evaluates to March 10 for non-leap years and March 9 in leap years. -2 will always evaluate to December 30th independent of whether it is a leap year.

BYMONTHDAY

This specifies the day of the month as a number. Valid values are 1 to 31.

An example is 10, which means the 10th day of the selected month. You can use the minus sign (-) to count backward from the last day, so, for example, `BYMONTHDAY=-1` means the last day of the month and `BYMONTHDAY=-2`

means the next to last day of the month.

BYDAY

This specifies the day of the week from Monday to Sunday in the form `MON`, `TUE`, and so on. Using numbers, you can specify the 26th Friday of the year, if using a `YEARLY` frequency, or the 4th `THU` of the month, using a `MONTHLY` frequency. Using the minus sign, you can say the second to last Friday of the month. For example, `-1 FRI` is the last Friday of the month.

BYHOUR

This specifies the hour on which the job is to run. Valid values are 0 to 23. As an example, 10 means 10 a.m.

BYMINUTE

This specifies the minute on which the job is to run. Valid values are 0 to 59. As an example, 45 means 45 minutes past the chosen hour.

BYSECOND

This specifies the second on which the job is to run. Valid values are 0 to 59. As an example, 30 means 30 seconds past the chosen minute.

BYSETPOS (10gR2 only)

This selects one or more items by position in the list of timestamps that result after the whole calendaring expression is evaluated. It is useful for requirements such as running a job on the last workday of the month. Rather than attempting to express this with the other `BY` clauses, you can code the calendaring expression to evaluate to a list of every workday of the month, and then add the `BYSETPOS` clause to select only the last item of that list.

All of the preceding properties can be used in the `SET PROPERTIES` clause as well.

The following properties are supported for the GET PROPERTIES clause only

Name: DATE_FORMAT

Type: STRING

Valid Values: N/A

Default: N//A

Region specific time format string, eg MMM/dd/yy:HH:mm:ss.

Name: AVAILABLE_TIMEZONES

Type: STRING ARRAY

Valid Values: N/A

Default: N//A

List of available time zone ids that can be used to set the TIMEZONE property. This property is for information purposes only.

Name: PREVIEW_DATES

Type: STRING(4000)

Valid Values: N/A

Default: "

Preview of dates that this schedule includes. This property is only valid for OMBRETRIEVE.

Properties for CALENDAR:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBALTER CALENDAR 'CALENDAR' SET PROPERTIES (NAME) VALUES ('MOD_
RENAME')
```

The alter command can also alter the owned schedule, following is an example:

OMBALTER CALENDAR 'CAL1' MODIFY SCHEDULE 'LOCALWINDOW' SET
PROPERTIES
(TIMEZONE) VALUES ('Pacific')

See Also

OMBALTER, OMBCREATE CALENDAR, OMBDROP CALENDAR

OMBALTER CALENDAR_MODULE

Purpose

To alter a calendar module.

Prerequisites

Should be in the context of a Project.

Syntax

```

parseAlterModule = OMBALTER CALENDAR_MODULE "QUOTED_STRING" ( ( RENAME TO
    "QUOTED_STRING" [ "alterModulePropertiesOrIconSetClause" ] ) |
    "alterModulePropertiesOrIconSetClause" )
alterModulePropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ (
    ( SET ( "setReferenceLocationClause" [ ( SET
    "setReferenceIconSetClause" ) | ( UNSET "unsetReferenceIconSetClause"
    ) ] ) ) | ( UNSET "unsetReferenceLocationClause" [ ( SET
    "setReferenceIconSetClause" ) | ( UNSET "unsetReferenceIconSetClause"
    ) ] ) ) | ( ( SET "setReferenceIconSetClause" ) | ( UNSET
    "unsetReferenceIconSetClause" ) ) ] ) | ( "setReferenceLocationClause"
    [ ( SET "setReferenceIconSetClause" ) | ( UNSET
    "unsetReferenceIconSetClause" ) ] ) | "setReferenceIconSetClause" ) )
    | ( UNSET ( "unsetReferenceLocationClause" [ ( SET
    "setReferenceIconSetClause" ) | ( UNSET "unsetReferenceIconSetClause"
    ) ] ) | "unsetReferenceIconSetClause" )
    ) ] ) | "unsetReferenceIconSetClause" )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

setPropertiesClause

Basic properties for CALENDAR_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Calendar Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Calendar Module

Properties for CALENDAR_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location to which the schedule will be deployed.

Name: EVAL_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location in which the scheduled object will be evaluated.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBALTER CALENDAR_MODULE 'CAL_MOD1' SET PROPERTIES (NAME)
VALUES
('MOD_RENAME')
```

See Also

OMBALTER, OMBCREATE CALENDAR_MODULE, OMBDROP CALENDAR_MODULE

OMBALTER CHANGE_DATA_CAPTURE

Purpose

This command is used to change the settings of a change data capture

Prerequisites

This command can only be executed in the context of a module and operates only on already existing change data captures.

Syntax

```
alterChangeSetCommand = OMBALTER ( CHANGE_DATA_CAPTURE "QUOTED_STRING" [ (
    CAPTURE CHANGES | USE CHANGES ) ] [ "renameClause" ] { SET
    "setPropertyClause" | "alterChangeSetSCOClauses" } )
renameClause = RENAME TO "QUOTED_STRING"
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
alterChangeSetSCOClauses = ( "addChangeSetSCOClauses" |
    "modifyChangeSetSCOClauses" | "deleteChangeSetSCOClauses" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addChangeSetSCOClauses = ( ( ADD CAPTURE ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING" ) { "addCaptureSCOClauses" } [
    "specialCaptureColumnsClause"+ ] )
modifyChangeSetSCOClauses = ( ( MODIFY CAPTURE ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING" ) ( ( "addCaptureSCOClauses" |
    "modifyCaptureSCOClauses" | "deleteCaptureSCOClauses" |
    "specialCaptureColumnsClause" ) )+ )
deleteChangeSetSCOClauses = ( DELETE CAPTURE ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
addCaptureSCOClauses = ( "addCaptureColumnsClause" | "addChangeSpecClause"
    | "addRowIdentifierClause" | "addTxnIdentifierClause" )
specialCaptureColumnsClause = [ DONT ] CAPTURE ( OLD_VALUES | USER_NAME )
modifyCaptureSCOClauses = ( "modifyCaptureColumnsClause" |
    "modifyChangeSpecClause" )
deleteCaptureSCOClauses = ( "deleteCaptureColumnsClause" |
    "deleteChangeSpecClause" | "deleteRowIdentifierClause" |
    "deleteTxnIdentifierClause" )
addCaptureColumnsClause = ( "addEachCaptureColumnClause" | (
    CAPTURE_COLUMNS ( NULL | ( "(" "columnNameList" ")" ) ) ) )
addChangeSpecClause = ( IDENTIFY ( INSERT | UPDATE | DELETE ) BY
    CHANGE_COLUMN "QUOTED_STRING" [ USING CHANGE_EXPRESSION
    "QUOTED_STRING" ] )
addRowIdentifierClause = ( "addEachRowIdentifierColumnClause" | (
    ROW_IDENTIFIER_COLUMNS ( NULL | ( "(" "columnNameList" ")" ) ) ) )
addTxnIdentifierClause = ( "addEachTxnIdentifierColumnClause" | (
    TXN_IDENTIFIER_COLUMNS ( NULL | ( "(" "columnNameList" ")" ) ) ) )
modifyCaptureColumnsClause = ( MODIFY CAPTURE_COLUMN "QUOTED_STRING" SET
    POSITION "INTEGER_LITERAL" )
modifyChangeSpecClause = ( SET ( INSERT | UPDATE | DELETE ) ( (
    CHANGE_COLUMN "QUOTED_STRING" ) | ( CHANGE_EXPRESSION "QUOTED_STRING"
    ) ) ) )
deleteCaptureColumnsClause = ( DELETE CAPTURE_COLUMN "QUOTED_STRING" )
deleteChangeSpecClause = ( DONT IDENTIFY ( INSERT | UPDATE | DELETE ) )
```

```
deleteRowIdentifierClause = ( ( DELETE ROW_IDENTIFIER_COLUMN
    "QUOTED_STRING" ) )
deleteTxnIdentifierClause = ( ( DELETE TXN_IDENTIFIER_COLUMN
    "QUOTED_STRING" ) )
addEachCaptureColumnClause = ( ADD CAPTURE_COLUMN "QUOTED_STRING" AT
    POSITION "INTEGER_LITERAL" )
columnNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
addEachRowIdentifierColumnClause = ( ( ADD ROW_IDENTIFIER_COLUMN
    "QUOTED_STRING" ) )
addEachTxnIdentifierColumnClause = ( ( ADD TXN_IDENTIFIER_COLUMN
    "QUOTED_STRING" ) )
```

Keywords And Parameters

setPropertyClause

Basic properties for CHANGE_DATA_CAPTURE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Change Data Capture

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Change Data Capture

ombalter_changeset\$alterChangeSetCommand = This clause alters a change data capture.

Properties for CHANGE_DATA_CAPTURE:

Name: CAPTUREFROM

Type: STRING

Valid Values: FROM_START_DATE, ALL_AVAILABLE

Default: ALL_AVAILABLE

This property is used to specify whether the Change Data Capture object will capture all available changes or changes that occurred after a specified date.

Name: CAPTUREFROMDATE

Type: STRING

Valid Values: N/A

Default: "

This property is used to specify the date from which changes will be captured by the Change Data Capture object.

Name: DBA_LOCATION

Type: STRING

Valid Values: N/A

Default: "

This property specifies the DBA location from which the Supplemental Log scripts will need to be deployed. This property is used if the schema in which the Change Data Capture will be deployed is not the owner of the source table and is also not a DBA.

Name: DELETE_DANGLING_REF

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the dangling references to tables need to be deleted during a reconcile operation on the Change Data Capture.

Name: DEPLOY_GET_TIME_FUNCTION

Type: BOOLEAN

Valid Values: true, false

Default: true

This property is used to indicate whether a function that returns the system time on the Source system needs to be deployed. This is used if the Change Data Capture object is used in a mapping that will be deployed on a database instance different from the instance containing the source table.

Name: DEPLOY_SOURCE_SCRIPTS

Type: BOOLEAN

Valid Values: true, false

Default: true

This property specifies whether OWB should generate and deploy supplemental log and instantiation script for the source tables.

Name: FIRSTSCN

Type: STRING

Valid Values: N/A

Default: "

This value is needed if the Change Data Capture and the source table exist on different database instances.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GET_TIME_FUNCTION

Type: STRING(28)

Valid Values: N/A

Default: OWB\$SYSDATE

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.CHANGELOGS.GETTIMEFUNC:DESCRIPTION"

Name: LONG_TRANSACTION_WAIT_TIME

Type: NUMBER

Valid Values: >= 0

Default: 0

This property specifies the number of seconds to wait for long running active transactions to complete during change extraction.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

This property is used to specify the name of tablespace where all Change Data Capture structures are to be created.

Name: TRANSACTION_WAIT_TIME

Type: NUMBER

Valid Values: >= 0

Default: 0

This property specifies the number of seconds to wait for active transactions to complete during change extraction.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

Examples

```
OMBALTER CHANGE_DATA_CAPTURE 'EMPLOYEE_CHANGES' ADD CAPTURE  
TABLE 'COMPANY'
```

This adds a new table whose changes are to be captured to the change data capture EMPLOYEE_CHANGES

See Also

OMBALTER, OMBCREATE CHANGE_DATA_CAPTURE, OMBRETRIEVE
CHANGE_DATA_CAPTURE, OMBDROP CHANGE_DATA_CAPTURE

OMBALTER CMI_DEFINITION

Purpose

Alter the CMI definition by renaming it, and/or reset its properties.

Prerequisites

Should be in the root context.

Syntax

```
alterMIVDefinitionCommand = OMBALTER ( CMI_DEFINITION "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | ( SET
    "setPropertiesClause" ) | ( USING DEFINITION_FILE "QUOTED_STRING" ) )
)
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterMIVDefinitionCommand

This command modifies an existing CMI definition.

QUOTED_STRING

Name of the existing CMI definition in single quotes.

renameClause

Rename an CMI definition.

setPropertiesClause

Associate a set of properties with the existing CMI definition.

Basic properties for CMI_DEFINITION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of an CMI Definition

Name: DESCRIPTION
 Type: STRING(4000)
 Valid Values: N/A
 Default: "
 Description of an CMI Definition

Name: MIV_TYPE
 Type: STRING(40)
 Valid Values: Applications,Databases
 Default: N/A
 Type of an CMI Definition

Properties for CMI_DEFINITION:

Name: LOCATION_UOID
 Type: STRING
 Valid Values: N/A
 Default: "
 Location Warehouse Builder should use to retrieve the data.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBALTER CMI_DEFINITION 'src_definition' RENAME TO 'tgt_definition' SET
PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a target
definition.', 'target definition')
```

This will rename the CMI definition "src_definition" to "tgt_definition", and set its description to "This becomes a target definition.", set its business name to "target definition".

```
OMBALTER CMI_DEFINITION 'src_definition' USING DEFINITION_FILE  
'/private/user1/miv_navision.xml'
```

This will change the CMI definition 'src_definition' to use the new definition file.

See Also

OMBALTER, OMBCREATE CMI_DEFINITION, OMBDROP CMI_DEFINITION

OMBALTER CMI_MODULE

Purpose

Alter the CMI module by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of project.

Syntax

```
alterMIVModuleCommand = OMBALTER ( CMI_MODULE "QUOTED_STRING" (
    "renameClause" [
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ] |
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataMetadataModule = ( ( SET ( (
    "alterPropertiesClause" [ ( SET
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) ) | (
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) ) ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) [
        "addOrRemoveOrModifyModuleReferenceLocationClause" ]
alterPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
unsetReferenceClauseForDataMetadataModule = (
    "unsetReferenceLocationClause" [ UNSET
    "unsetReferenceMetadataLocationOrIconSetClause" ] |
    "unsetReferenceMetadataLocationOrIconSetClause" )
addOrRemoveOrModifyModuleReferenceLocationClause = (
    "addReferenceLocationClause" | "removeReferenceLocationClause" |
    "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
    "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceMetadataLocationOrIconSetClause = (
    "unsetReferenceMetadataLocationClause" [ UNSET
    "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" SET AS DEFAULT
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
```

```
"FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
"QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceMetadataLocationClause = ( REFERENCE | REF )
METADATA_LOCATION "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
```

Keywords And Parameters

alterMIVModuleCommand

This command modifies an existing CMI module.

QUOTED_STRING

Name of the existing CMI module in single quotes.

renameClause

Rename an CMI module.

alterPropertiesOrReferenceClauseForDataMetadataModule

Alter existing CMI module's properties and/or locations and/or icon sets.

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the CMI module.

unsetReferenceClauseForDataMetadataModule

Unset location and/or icon set for the CMI module.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the CMI module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a location to the existing CMI module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the CMI module.

`unsetReferenceLocationClause`

Unset a location to the existing CMI module.

`unsetReferenceMetadataLocationOrIconSetClause`

Unset metadata location and/or icon set for the CMI module.

`addReferenceLocationClause`

Add a runtime location to the CMI module.

`removeReferenceLocationClause`

Remove a runtime location from the CMI module.

`modifyReferenceLocationClause`

Modify a runtime location of the CMI module.

`propertyValue`

Value of a property.

`setReferenceMetadataLocationClause`

Set metadata location for the CMI module.

`setReferenceIconSetClause`

Set icon set for the CMI module.

`unsetReferenceMetadataLocationClause`

Unset metadata location for the CMI module.

`unsetReferenceIconSetClause`

Unset icon set for the CMI module.

Examples

```
OMBALTER CMI_MODULE 'src_module' RENAME TO 'tgt_module' SET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a target module.',  
'target module')
```

This will rename the CMI module "src_module" to "tgt_module", and set its

description to "This becomes a target module.", set its business name to "target module".

See Also

OMBALTER, OMBCREATE CMI_MODULE, OMBDROP CMI_MODULE

OMBALTER COLLECTION

Purpose

Alter the collection by adding, removing or modifying shortcuts.

Prerequisites

Should be in the context of a project, before altering a collection.

Syntax

```
alterCollectionCommand = OMBALTER ( COLLECTION "QUOTED_STRING" ( [
    "renameClause" ] [ SET "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    SET "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause"
    ] [ "alterCollectionReferences" ] ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterCollectionReferences = ( "addReferenceClause" |
    "removeReferenceClause" | "reconcileReferenceClause" )+
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addReferenceClause = ADD REFERENCE TO "componentRefClause"
removeReferenceClause = REMOVE REFERENCE TO "componentRefClause"
reconcileReferenceClause = RECONCILE
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
componentRefClause = ( EXTERNAL_TABLE | TABLE | VIEW | MATERIALIZED_VIEW |
    SEQUENCE | VARYING_ARRAY | OBJECT_TYPE | NESTED_TABLE | MAPPING |
    DIMENSION | CUBE | ADVANCED_QUEUE | STREAMS_QUEUE | QUEUE_TABLE |
    ORACLE_MODULE | TRANSFORMATION_MODULE | FLAT_FILE_MODULE | FLAT_FILE |
    PROCESS_FLOW | PROCESS_FLOW_PACKAGE | PROCESS_FLOW_MODULE |
    SAP_MODULE | CMI_MODULE | COLLECTION | FUNCTION | PROCEDURE | PACKAGE
    | BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE |
    PRESENTATION_TEMPLATE | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL
    | ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
    GATEWAY_MODULE | CONFIGURATION | REGISTERED_FUNCTION |
    PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER | DATA_AUDITOR |
    TRANSPORTABLE_MODULE | EXPERT_MODULE | EXPERT | CALENDAR_MODULE |
    CALENDAR | DATA_PROFILE | DATA_RULE_MODULE | DATA_RULE )
    "QUOTED_STRING"
```

Keywords And Parameters

`alterCollectionCommand`

Alter a collection of objects.

`QUOTED_STRING`

Name of the existing collection in quotes.

`renameClause`

Rename a collection.

setPropertyClause

Associate a set of properties with a collection.

Basic properties for COLLECTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the collection

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the collection

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

alterCollectionReferences

Alter the collections references, includes adding, dropping and reconciling the collection.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

addReferenceClause

Add a reference to the collection.

removeReferenceClause

Remove a reference from the collection.

reconcileReferenceClause

Reconcile the collection, deleting references which now refer to deleted objects.

propertyValue

Value of a property.

componentRefClause

Specify the type of the object to reference.

Examples

```
OMBALTER COLLECTION 'PURCHASING_WAREHOUSE' SET PROPERTIES  
(DESCRIPTION)
```

```
VALUES ('Group for purchasing usage.') ADD REFERENCE TO TABLE  
'PURCHASING/PRODUCT'
```

See Also

OMBALTER, OMBCREATE COLLECTION

OMBALTER CONFIGURATION

Purpose

Alter the Configuration by renaming it, and/or reset its properties and Icon Set.

Prerequisites

Should be in the context of a project.

Syntax

```
alterConfigurationCommand = OMBALTER ( CONFIGURATION "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterConfigurationCommand

This command modifies an existing Configuration.

renameClause

Rename Configuration.

alterPropertiesOrIconSetClause

Alter existing Configuration properties and/or Icon Set.

setPropertiesClause

Associate a set of properties with the existing Configuration.

Basic properties for CONFIGURATION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Configuration

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Configuration.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set an Icon Set to the existing Configuration.

unsetReferenceIconSetClause

Unset an Icon Set from the existing Configuration.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBALTER CONFIGURATION 'QA_CONFIG' RENAME TO 'QA_MY_CONFIG' SET
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('This is my set of configuration.',
'Qa My Configuration')
```

This will rename the Configuration "QA_CONFIG" to "QA_MY_CONFIG", and set its description to "This is my set of configuration.", set its business name to "Qa My Configuration".

See Also

OMBALTER, OMBCREATE CONFIGURATION

OMBALTER CONNECTOR

Purpose

Alter the connector by renaming it, and/or reset its referenced location or properties or icon set.

Prerequisites

Can be in any context; the name is a name of the connector's owning location and a connector name separated by slash.

Syntax

```
alterConnectorCommand = OMBALTER ( CONNECTOR "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrReferenceClause" ] |
    "alterPropertiesOrReferenceClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClause = SET ( "setPropertiesClause" [ SET
    "setReferenceClause" [ UNSET "unsetReferenceIconSetClause" ] | UNSET
    "unsetReferenceIconSetClause" [ SET "setReferenceClause" ] ] |
    "setReferenceClause" [ UNSET "unsetReferenceIconSetClause" ] ) | UNSET
    "unsetReferenceIconSetClause" [ SET "setReferenceClause" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( "setReferenceToLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceToLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterConnectorCommand

Alter the connector specified by the quoted string.

renameClause

Rename the connector.

setPropertiesClause

Set specified properties of the connector.

unsetReferenceIconSetClause

Unset specified Icon Set.

propertyNameList

The names of the properties whose values you want to set.

Properties for CONNECTOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the connector.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the connector.

Name: DATABASE_LINK_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Database Link name.

Properties for CONNECTOR:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList

The values for the named properties.

setReferenceToLocationClause

Set the name of the location which the connector references.

setReferenceIconSetClause

Set specified Icon Set.

propertyValue

A property value.

Examples

```
OMBALTER CONNECTOR 'LOCATION_NAME/OLD_CONNECTOR' RENAME TO  
'NEW_CONNECTOR'
```

```
SET PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a new  
connector.', 'new connector')
```

This will rename the connector "OLD_CONNECTOR" to "NEW_CONNECTOR", and set

its description to "This becomes a new connector", set its business name to "new connector".

```
OMBALTER CONNECTOR 'LOCATION_NAME/A_CONNECTOR' SET REF  
LOCATION
```

```
'MY_NEW_TARGETLOCATION'
```

See Also

OMBALTER, OMBCREATE CONNECTOR, OMBDROP CONNECTOR

OMBALTER CONTROL_CENTER

Purpose

Alter the control center by renaming it, and/or resetting its properties, and/or adding/removing a referenced location, and/or resetting properties of a referenced location.

Prerequisites

Can be in any context.

Syntax

```
alterControlCenterCommand = OMBALTER ( CONTROL_CENTER "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" [ "alterIconSetClause" ] {
        "alterReferenceLocationClause" } ] | SET "setPropertiesClause" [
        "alterIconSetClause" ] { "alterReferenceLocationClause" } |
        "alterIconSetClause" { "alterReferenceLocationClause" } | {
        "alterReferenceLocationClause" } ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
alterIconSetClause = SET "setReferenceIconSetClause" [ UNSET
    "unsetReferenceIconSetClause" ] | UNSET "unsetReferenceIconSetClause"
    [ SET "setReferenceIconSetClause" ]
alterReferenceLocationClause = "addReferenceLocationClause" |
    "modifyReferenceLocationClause" | "deleteReferenceLocationClause"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addReferenceLocationClause = ADD ( REF | REFERENCE ) LOCATION
    "QUOTED_STRING" [ SET "setPropertiesClause" ]
modifyReferenceLocationClause = MODIFY ( REF | REFERENCE ) LOCATION
    "QUOTED_STRING" SET "setPropertiesClause"
deleteReferenceLocationClause = DELETE ( REF | REFERENCE ) LOCATION
    "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterControlCenterCommand

Alter a control center.

renameClause

Rename the control center to the value of the following quoted string.

setPropertiesClause

Set the specified properties of the control center.

alterIconSetClause

Set or unset the Icon Set.

alterReferenceLocationClause

Alter the referenced location.

propertyNameList

The names of the properties whose values you want to set.

Properties for CONTROL_CENTER:

Name: HOST

Type: STRING

Valid Values: N/A

Default: N/A

The host machine the control center is installed on.

Name: PORT

Type: NUMBER

Valid Values: 1 - 65535

Default: 1521

The port number of the database in which the control center is installed.

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The service name of the database in which the control center is installed.

Name: USER

Synonym: USER_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The name of the database user you wish to connect to the control center as.

Name: PASSWORD

Type: STRING
Valid Values: N/A
Default: N/A
The password.

Name: SCHEMA
Type: STRING
Valid Values: N/A
Default: N/A
The name of the schema in which the control center is installed.

All of the preceding properties (except of PASSWORD) are mandatory for
OMBCREATE CONTROL_CENTER.

Basic properties for CONTROL_CENTER:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the control center.

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the control center.

Properties for a referenced location of the control center:

Name: IS_SOURCE
Type: BOOLEAN
Valid Values: true, false
Default: true
If true, then a referenced location is a source location.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, then a referenced location is a target location.

Properties for CONTROL_CENTER:

Name: HOST

Type: STRING

Valid Values: N/A

Default: "

Host of the location

Name: NET_SERVICE_NAME

Type: STRING(2000)

Valid Values: N/A

Default: "

Net Service Name of the location

Name: PASSWORD

Type: STRING(30)

Valid Values: N/A

Default: "

Password for the location

Name: PORT

Type: NUMBER

Valid Values: N/A

Default: 1521

Port of the location

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: "

Schema name for the location

Name: SERVICE_NAME
Type: STRING
Valid Values: N/A
Default: "
Service Name of the location

Name: USER
Type: STRING
Valid Values: N/A
Default: "
User name for the location

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList
The values for the named properties.

setReferenceIconSetClause
Set the icon set.

unsetReferenceIconSetClause
Unset the icon set.

addReferenceLocationClause
Add a referenced location to the control center and/or set its properties.

modifyReferenceLocationClause
Set/reset properties of the referenced location.

deleteReferenceLocationClause
Delete a referenced location from the control center.

propertyValue
A property value.

Examples

```
OMBALTER CONTROL_CENTER 'MY_CONNECTION' RENAME TO 'NEW_
CONNECTION' SET
```

```
PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a new
repository.', 'new repository')
```

This will rename the control center "MY_CONNECTION" to "NEW_CONNECTION", and set its description to "This becomes a new repository", set its business name to "new repository".

```
OMBALTER CONTROL_CENTER 'MY_CONNECTION' ADD REF LOCATION 'MY_
TGT_LOCATION'
```

```
SET PROPERTIES (IS_TARGET, IS_SOURCE) VALUES ('true', 'false')
```

This will add a new referenced location MY_TGT_LOCATION and set this referenced location for using as a target only.

See Also

OMBALTER, OMBCREATE CONTROL_CENTER, OMBDROP CONTROL_CENTER

OMBALTER CORRECTION_MAPS_ACTION_PLAN

Purpose

Alter an action plan for creating a correction map.

Prerequisites

In the context of a data profile.

Syntax

```
AlterActionPlanCommand = ( OMBALTER ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ( (
    "renameActionPlanClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "addActionClause" { "addActionClause" |
    "modifyActionClause" | "deleteActionClause" } ) | (
    "modifyActionClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "deleteActionClause" { "addActionClause"
    | "modifyActionClause" | "deleteActionClause" } ) ) )
renameActionPlanClause = RENAME TO "QUOTED_STRING"
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
modifyActionClause = MODIFY ACTION "QUOTED_STRING" ( (
    "renameActionClause" [ "setUnsetClause" ] ) | "setUnsetClause" )
deleteActionClause = DELETE ACTION "QUOTED_STRING"
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
    "setReferenceClause"
renameActionClause = RENAME TO "QUOTED_STRING"
setUnsetClause = ( ( SET "setClauseForAlter" ) | ( UNSET
    "unsetReferenceClause" ) )
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
    "useClause" }
setClauseForAlter = ( "propertiesClause" [ SET "setReferenceClause" |
    UNSET "unsetReferenceClause" ] ) | "setReferenceClause"
unsetReferenceClause = ( REF | REFERENCE )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

AlterActionPlanCommand

Alter an action plan.

QUOTED_STRING

Action plan name.

renameActionPlanClause

Rename an action plan.

QUOTED_STRING

Action plan name.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Action name.

modifyActionClause

Modify an action of an action plan.

QUOTED_STRING

Action name.

deleteActionClause

Delete an action of an action plan.

QUOTED_STRING

Action name.

setClause

Set the properties of an action and/or associate an object with an action.

renameActionClause

Rename an action of an action plan.

setUnsetClause

Set the properties and/or associate/disassociate an object with an action.

propertiesClause

Set the properties and/or associate/disassociate an object with an action.

setReferenceClause

Associate an object with an action.

ObjType

Object type. The only valid value is DATA_PROFILE_TABLE.

setClauseForAlter

Set the properties and/or associate/disassociate an object with an action.

unsetReferenceClause

Disassociate a previously associated object from an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Currently used only in CORRECTION_SCHEMA_ACTION_PLAN.

propertyValue

Value of a property.

Examples

```
OMBALTER CORRECTION_MAPS_ACTION_PLAN 'CORRECT_INV_LOC_MAP'  
ADD ACTION  
'UK_LOC_U' SET PROPERTIES (DATA_RULE_USAGE_NAME, ERROR_  
HANDLING_STRATEGY,  
CORRECTION_STRATEGY) VALUES ('UK_LOC_U', 'CLEANSE', 'UK_  
MATCHMERGE') SET  
REF PROFILE_REFERENCE 'LOC'
```

See Also

OMBCREATE CORRECTION_MAPS_ACTION_PLAN, OMBPROFILE

OMBALTER CORRECTION_SCHEMA_ACTION_PLAN

Purpose

Alter an action plan for creating a correction schema, identifying the tables that will be mapped to that schema.

Prerequisites

In the context of a data profile.

Syntax

```
AlterActionPlanCommand = ( OMBALTER ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ( (
    "renameActionPlanClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "addActionClause" { "addActionClause" |
    "modifyActionClause" | "deleteActionClause" } ) | (
    "modifyActionClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "deleteActionClause" { "addActionClause"
    | "modifyActionClause" | "deleteActionClause" } ) ) )
renameActionPlanClause = RENAME TO "QUOTED_STRING"
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
modifyActionClause = MODIFY ACTION "QUOTED_STRING" ( (
    "renameActionClause" [ "setUnsetClause" ] ) | "setUnsetClause" )
deleteActionClause = DELETE ACTION "QUOTED_STRING"
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
    "setReferenceClause"
renameActionClause = RENAME TO "QUOTED_STRING"
setUnsetClause = ( ( SET "setClauseForAlter" ) | ( UNSET
    "unsetReferenceClause" ) )
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
    "useClause" }
setClauseForAlter = ( "propertiesClause" [ SET "setReferenceClause" |
    UNSET "unsetReferenceClause" ] ) | "setReferenceClause"
unsetReferenceClause = ( REF | REFERENCE )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

AlterActionPlanCommand

Alter an action plan.

QUOTED_STRING

Action plan name.

renameActionPlanClause

Rename an action plan.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Action name.

modifyActionClause

Modify an action of an action plan.

QUOTED_STRING

Action name.

deleteActionClause

Delete an action of an action plan.

QUOTED_STRING

Action name.

setClause

Set the properties of an action and/or associate an object with an action.

renameActionClause

Rename an action of the action plan.

setUnsetClause

Set the properties and/or associate/disassociate an object with an action.

propertiesClause

Set the properties and/or associate/disassociate an object with an action.

setReferenceClause

Associate an object with an action.

ObjType

Object type. The only valid value is DATA_PROFILE_TABLE.

setClauseForAlter

Set the properties and/or associate/disassociate an object with an action.

unsetReferenceClause

Disassociate a previously associated object from an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Name of data rule usage.

propertyValue

Value of a property.

Examples

```
OMBALTER CORRECTION_SCHEMA_ACTION_PLAN 'CORRECT_INV_LOC' ADD  
ACTION  
'GEN_INV' SET REFERENCE PROFILE_REFERENCE 'INV' USE DATA_RULE_  
USAGE  
'PLANT_ID_ORG_I_U'
```

See Also

OMBCREATE CORRECTION_SCHEMA_ACTION_PLAN, OMBPROFILE

OMBALTER CUBE

Purpose

This command alters a Cube.

Prerequisites

Should be in Oracle Module context.

Syntax

```
alterCubeCommand = OMBALTER CUBE "cubeName" ( ( "renameClause" [
    "alterPropertiesOrIconSetClause" ] [ "setCubeAggFunctionClause" ] [
    "alterCubeDescendentsClause" ] [ "cubeImplementationClause" ] ) |
    "cubePartitionClause" | ( "alterPropertiesOrIconSetClause" [
    "setCubeAggFunctionClause" ] [ "alterCubeDescendentsClause" ] [
    "cubeImplementationClause" ] ) | ( "setCubeAggFunctionClause" [
    "alterCubeDescendentsClause" ] [ "cubeImplementationClause" ] ) | (
    "alterCubeDescendentsClause" [ "cubeImplementationClause" ] ) |
    "cubeImplementationClause" | "alterCubeUnBindingClause" )
cubeName = "QUOTED_STRING"
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( ( "setPropertiesClause" [ ( SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ) ]
    ) | "setReferenceIconSetClause" ) | UNSET
    "unsetReferenceIconSetClause"
setCubeAggFunctionClause = SET AGGREGATE_FUNCTION TO "aggFunctionName"
alterCubeDescendentsClause = ( ADD ( "addMeasureClause" |
    "addDimensionUseClause" | "addCompositeDimensionClause" ) | MODIFY (
    "modifyMeasureClause" | "modifyDimensionUseClause" |
    "modifyCompositeDimensionClause" ) | DELETE ( "measureLocator" |
    "compositeDimensionLocator" | "dimensionUseLocator" ) )+
cubeImplementationClause = IMPLEMENTED BY ( ( "cubeBindingClause" {
    "measureBindingClause" | "dimensionUseBindingClause" } ) |
    "cubeAutoBindingClause" )
cubePartitionClause = ( SET PARTITION LEVEL "levelName" OF HIERARCHY
    "hierName" OF DIMENSION "dimensionName" )
alterCubeUnBindingClause = DELETE BINDING
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
aggFunctionName = "QUOTED_STRING"
addMeasureClause = MEASURE "measureName" [ SET "setPropertiesClause" ] [
    "setAggFunctionClause" ] [ "setPreComputedLevelsClause" ] [
    "setCalcExprClause" ]
addDimensionUseClause = DIMENSION_USE "dimensionUseName" [ SET
    "setPropertiesClause" ] [ "setDimensionUseReferenceClause" ] [ AT
    POSITION "INTEGER_LITERAL" ]
addCompositeDimensionClause = COMPOSITE_DIMENSION "compositeDimensionName"
    [ SET "setPropertiesClause" ] [
    "setCompositeDimensionReferenceClause" ]
modifyMeasureClause = "measureLocator" ( "renameClause" | SET
    "setPropertiesClause" | "setAggFunctionClause" |
    "setPreComputedLevelsClause" | "setCalcExprClause" )
modifyDimensionUseClause = "dimensionUseLocator" ( "renameClause" |
    "moveToClause" | SET "setPropertiesClause" |
    "modifyDimensionUseRoleClause" )
```

```

modifyCompositeDimensionClause = "compositeDimensionLocator" (
    "renameClause" | SET "setPropertyClause" |
    "setCompositeDimensionReferenceClause" )
measureLocator = MEASURE "measureName"
compositeDimensionLocator = COMPOSITE_DIMENSION "compositeDimensionName"
dimensionUseLocator = DIMENSION_USE "dimensionUseName"
cubeBindingClause = ( TABLE "tableName" | VIEW "viewName" )
measureBindingClause = "measureLocator" BOUND TO COLUMN "columnName"
dimensionUseBindingClause = "dimensionUseLocator" BOUND TO COLUMN
    "columnName"
cubeAutoBindingClause = SYSTEM
levelName = "QUOTED_STRING"
hierName = "QUOTED_STRING"
dimensionName = "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
measureName = "QUOTED_STRING"
setAggFunctionClause = SET AGGREGATE_FUNCTIONS "aggFunctionList" FOR
    DIMENSIONS "dimensionList"
setPreComputedLevelsClause = SET PRECOMPUTE_LEVELS "preComputedLevelList"
setCalcExprClause = SET CALCULATED_EXPRESSION "(" "calcExpr" ")"
dimensionUseName = "QUOTED_STRING"
setDimensionUseReferenceClause = [ USE ROLE "roleName" ] SET ( REF |
    REFERENCE ) [ LEVEL "levelName" OF ] DIMENSION "dimensionName"
compositeDimensionName = "QUOTED_STRING"
setCompositeDimensionReferenceClause = SET ( REF | REFERENCE ) DIMENSIONS
    "(" "dimensionName" { "," "dimensionName" } ")"
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
modifyDimensionUseRoleClause = SET "roleLocator" | UNSET "roleLocator"
tableName = "QUOTED_STRING"
viewName = "QUOTED_STRING"
columnName = "QUOTED_STRING"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
aggFunctionList = "(" "aggFunction" { "," "aggFunction" } ")"
dimensionList = "(" "dimension" { "," "dimension" } ")"
preComputedLevelList = "(" "preComputedLevel" { "," "preComputedLevel" }
    ")"
calcExpr = "QUOTED_STRING"
roleName = "QUOTED_STRING"
roleLocator = ROLE "roleName"
aggFunction = "QUOTED_STRING"
dimension = "QUOTED_STRING"
preComputedLevel = "QUOTED_STRING"

```

Keywords And Parameters

setPropertyClause

Basic properties for CUBE, MEASURE and DIMENSION_USE:

Basic properties for CUBE :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Cube

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Cube

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: 'true', 'false'

Default: 'true'

The dimension is visible to OLAP end user

Name: UNIQUE_KEY_CONSTRAINT

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

set the Unique Key constraint on the Business Key

Name: STORAGE

Type: STRING

Valid Values: 'RELATIONAL', 'MOLAP'

Default: 'RELATIONAL'

The storage of a cube can be relational or molap

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the analytical workspace name where the cube is implemented

Name: AW_CUBE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace cube physical object name

Name: USE_GLOBAL_INDEX

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether to generate a composite for measure partition combination

Name: BITMAP_INDEX

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether to generate a bitmap for a cube

Name: SUMMARY_REFRESH_METHOD

Type: STRING

Valid Values: 'On Demand', 'On Commit'

Default: 'On Commit'

Sets the Solve flag for Relational Cube whether to solve the cube 'On Demand' or 'On Commit'

Name: SPARSITY_ULTRA_COMPRESS

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether its a compressed cube. Only valid for Molap cube.

Basic properties for Measure :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Cube

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Cube

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: 'true', 'false'

Default: 'true'

The dimension is visible to OLAP end user

Name: CALCULATED_MEASURE

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the measure to be aggregatable

Name: AUTO_SOLVE

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether to measure is Auto Solve

Name: AW_MEASURE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace measure physical object name

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH

LOCAL TIME ZONE, VARCHAR, VARCHAR2

Default: NUMBER

Sets the datatype of a Cube Measure

Name: SCALE

Type: NUMBER

Valid Values: -85 - 125

Default: 1

The scale of a number.

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 39

Default: 1

The precision of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Basic properties for Cube Dimension Use :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Cube Dimension Use

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Cube Dimension Use

Properties for CUBE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS

Type: STRING

Valid Values: DEPLOY_AGGREGATION, DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY,

DEPLOY_TO_CATALOG_ONLY

Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for cube, they are DDL Scripts for Relational Cube or Scripts for OLAP API-II or Scripts for AW.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MATERIALIZED_VIEW_INDEX_TABLESPACE

Type: STRING

Valid Values: N/A

Default: USERS

Tablespace for materialized view indexes.

Name: MATERIALIZED_VIEW_TABLESPACE

Type: STRING

Valid Values: N/A

Default: USERS

Tablespace for materialized view.

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Cube is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

alterCubeCommandExampleTag??

See Also

OMBALTER, OMBCREATE CUBE, OMBRETRIEVE CUBE, OMBDROP CUBE

OMBALTER DATA_AUDITOR

Purpose

Alter the content of a data auditor.

Prerequisites

1. The current context of scripting must be an Oracle Module
2. No concurrent user should be modifying the data auditor

Syntax

```
alterDataAuditorCommand = OMBALTER DATA_AUDITOR "dataAuditorName" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] [
    "alterDataAuditorDecendantsClause" ] |
    "alterPropertiesOrIconSetClause" [ "alterDataAuditorDecendantsClause"
    ] | "alterDataAuditorDecendantsClause" )
dataAuditorName = "QUOTED_STRING"
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] )
    | "setReferenceIconSetClause" ) ) | UNSET
    "unsetReferenceIconSetClause"
alterDataAuditorDecendantsClause = ( ADD "addOperatorClause" | MODIFY (
    "modifyOperatorClause" | "modifyChildClause" ) | DELETE
    "operatorLocator" )+
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addOperatorClause = "operatorType" OPERATOR "operatorName" [ SET
    "setPropertiesClause" ] [ BOUND TO "setBindingClause" ]
modifyOperatorClause = "operatorLocator" ( SET "setPropertiesClause" )
modifyChildClause = "childBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
operatorLocator = OPERATOR "operatorName"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
operatorType = "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
setBindingClause = "bindableLocator"
childBottomUpLocator = "childType" "childName" { OF "childType"
    "childName" } [ OF "mappableBottomUpLocator" ]
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
bindableLocator = "bindableType" "bindableName"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
mappableBottomUpLocator = "operatorLocator"
bindableType = "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"
```

Keywords And Parameters

alterDataAuditorCommand

Alter the content of a data auditor.

dataAuditorName

Name of data auditor.

renameClause

Rename a data auditor

alterPropertiesOrIconSetClause

Alter properties or the icon set.

alterDataAuditorDecendantsClause

Add, modify, delete or alter operators.

setPropertiesClause

Set the properties of the data auditor.

setReferenceIconSetClause

Set icon set for the data auditor.

unsetReferenceIconSetClause

Unset icon set for the data auditor.

addOperatorClause

Adds an operator to a data auditor. `OMBCREATE DATA_AUDITOR 'MAP1'`

`SET PROPERTIES (business_name, description)`

`VALUES ('My map', 'Audit table foo')`

`ADD TABLE OPERATOR 'CUST_SRC'`

`BOUND TO TABLE '../SRC_MODULE/CUST_SRC'`

modifyOperatorClause

Modify operator.

modifyChildClause

Modify a child that belongs to a data auditor.

operatorLocator

Locate operator.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

operatorType

Type of a mapping operator. The following operator types are available:
CUBE, DIMENSION, MATERIALIZED_VIEW, TABLE, VIEW.

operatorName

Name of an operator.

setBindingClause

Set the binding during the creation of a operator.

childBottomUpLocator

Locate object in data auditor.

propertyKey

A property key for an object.

Basic properties for DATA_AUDITOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the data auditor

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the daa auditor

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

#

Properties for DATA_AUDITOR:

Name: ANALYZE_TABLE_SAMPLE_PERCENTAGE

Type: NUMBER

Valid Values: N/A

Default: 90

The default percentage of rows to be sampled when the target tables are analyzed for statistics to improve performance during insertion.

Name: ANALYZE_TABLE_STATEMENTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate statistics collection statement if this is true.

Name: ANSI_SQL_SYNTAX

Type: BOOLEAN

Valid Values: true, false

Default: true

A switch between ANSI and Oracle SQL Syntax.

Name: BULK_PROCESSING_CODE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate bulk processing code if this is true.

Name: BULK_SIZE

Type: NUMBER

Valid Values: N/A

Default: 50

The default number of rows to be fetched in batch during cursor processing.

Name: COMMIT_CONTROL

Type: STRING

Valid Values: AUTO_COMMIT, AUTO_CORR_COMMIT, MANUAL_COMMIT

Default: AUTO_COMMIT

Options for how commit is performed

Name: COMMIT_FREQUENCY

Type: NUMBER

Valid Values: N/A

Default: 1000

The default number of rows processed before a commit is issued.

Name: DEFAULT_AUDIT_LEVEL

Type: STRING

Valid Values: COMPLETE, ERROR_DETAILS, NONE, STATISTICS

Default: ERROR_DETAILS

The default audit level when the step is executed.

Name: DEFAULT_OPERATING_MODE

Type: STRING

Valid Values: ROW_BASED, ROW_BASED_TARGET_ONLY, SET_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY

Default: SET_BASED_FAIL_OVER_TO_ROW_BASED

The default operating mode.

Name: DEFAULT_PURGE_GROUP

Type: STRING

Valid Values: N/A

Default: WB

The default purge group to be used when the step is executed.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: ENABLE_PARALLEL_DML

Type: BOOLEAN

Valid Values: true, false

Default: true

Determine if PDML is enabled at runtime.

Name: ERROR_TRIGGER

Type: STRING

Valid Values: N/A

Default: "

Error trigger procedure name.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_MODE

Type: STRING

Valid Values: ALL_MODES, ROW_BASED, ROW_BASED_TARGET_ONLY, SET_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY

Default: ALL_MODES

The operating modes for which code should be generated.

Name: MAXIMUM_NUMBER_OF_ERRORS

Type: NUMBER

Valid Values: N/A

Default: 50

The default maximum number of errors encountered before terminating the step execution.

Name: OPTIMIZED_CODE

Type: BOOLEAN

Valid Values: true, false

Default: true

Attempt to generate optimized code if this is true.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TARGET_LOAD_ORDERING

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate target load ordering code.

Name: THRESHOLD_MODE

Type: STRING

Valid Values: PERCENTAGE, SIX_SIGMA

Default: PERCENTAGE

Use six sigma or percentage for failure thresholds.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

bindableLocator

Location of the object to be bound to an operator.

childType

Type of a child that belongs to data auditor or operator

childName

Name of a child that belongs to data auditor, operator.

mappableBottomUpLocator

Locate mappable object in data auditor.

bindableType

Type of object bound to an operator.

bindableName

Name of the object bound to operator.

Examples

```
OMBALTER DATA_AUDITOR 'MAP1' RENAME TO 'MAP2'
```

```
OMBALTER DATA_AUDITOR 'MAP1' DELETE OPERATOR 'OP1'
```


See Also

OMBALTER, OMBCREATE DATA_AUDITOR, OMBRETRIEVE DATA_AUDITOR,
OMBDROP DATA_AUDITOR

OMBALTER DATA_PROFILE

Purpose

Alter the Data Profile by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of project.

Syntax

```
alterDataProfileCommand = OMBALTER ( DATA_PROFILE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrReferenceClause" ] | ( [
        "alterPropertiesOrReferenceClause" ] ( { ( ADD | DELETE ) ( TABLE |
        VIEW | EXTERNAL_TABLE | MATERIALIZED_VIEW | DIMENSION | CUBE )
        "QUOTED_STRING" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClause = SET ( "setPropertiesClause" [ SET
    "setReferenceClause" [ UNSET "unsetReferenceClause" ] | UNSET
    "unsetReferenceClause" [ SET "setReferenceClause" ] ] |
    "setReferenceClause" [ UNSET "unsetReferenceClause" ] ) | UNSET
    "unsetReferenceClause" [ SET "setReferenceClause" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
unsetReferenceClause = ( "unsetReferenceLocationClause" [ UNSET
    "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterDataProfileCommand

This command modifies an existing Data Profile.

QUOTED_STRING

Name of the existing Data Profile in single quotes.

renameClause

Rename a Data Profile.

alterPropertiesOrReferenceClause

Alter existing Data Profile's core properties, locations and icon sets.

setPropertiesClause

Associate a set of properties with the existing Data Profile.

Configuration properties for DATA_PROFILE that affect loading:

Name: COPY_DATA

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable copying of data from source to profile workspace.

Name: CALCULATE_DATATYPES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data type discovery for the selected table.

Name: CALCULATE_COMMON_FORMATS

Type: BOOLEAN

Valid Values: true | false

Default: false

This tells the profiler if common formats are to be discovered for all sources in this profile.

Name: NULL_VALUE

Type: STRING

Valid Values: any string value

Default: null

This value will be considered as the null value when profiling. Please enclose the value in single quotes. An unquoted null (the current default value) will be considered a database null.

Name: SAMPLE_RATE

Type: NUMBER

Valid Values: 1-100

Default: 100

This value will be the percent of total rows that will be randomly selected during loading.

Configuration properties for DATA_PROFILE that affect profiling:

Name: CALCULATE_DOMAINS

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable domain discovery.

Name: DOMAIN_MAX_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

The maximum number of distinct values in a column in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MAX_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

The maximum number of distinct values in a column, expressed as a percentage of the total number of rows in the table, in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MIN_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

The minimum number of rows for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: DOMAIN_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

The minimum number of rows, expressed as a percentage of the total number of rows, for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: CALCULATE_UK

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable unique key discovery.

Name: UK_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a unique key relationship.

Name: CALCULATE_FD

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable functional dependency discovery.

Name: FD_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a functional dependency relationship.

Name: FD_UK_LHS_COUNT

Type: NUMBER

Valid Values: 1-number of attributes of source less 1

Default: 1

This is the maximum number of attributes for unique key and functional dependency profiling.

Name: CALCULATE_FK

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable foreign key discovery.

Name: FK_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a foreign key relationship.

Name: CALCULATE_REDUNDANT_COLUMNS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable redundant column discovery with respect to a foreign key-unique key pair.

Name: REDUNDANT_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that are redundant.

Name: CALCULATE_DATA_RULES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data rule profiling for the selected table.

Name: CALCULATE_PATTERNS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable pattern discovery.

Name: MAX_NUM_PATTERNS

Type: NUMBER

Valid Values: any number less than the number of rows of the source

Default: 10

This tells the profiler to get the top-N patterns for the attribute.

Properties for DATA_PROFILE:

Name: ABAP_DIRECTORY

Type: STRING

Valid Values: N/A

Default: /tmp

Location where SAP data is dumped as flat files

Name: CALCULATE_COMMON_FORMATS

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable common format discovery for all the columns in this profile.

Name: CALCULATE_DATATYPES

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable data type discovery for all the columns in this profile.

Name: CALCULATE_DATA_RULES

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable data rule profiling for the selected table.

Name: CALCULATE_DOMAINS

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable domain discovery.

Name: CALCULATE_FD

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable functional dependency discovery.

Name: CALCULATE_FK

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable foreign key discovery.

Name: CALCULATE_PATTERNS

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable pattern discovery.

Name: CALCULATE_REDUNDANT_COLUMNS

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable redundant column discovery.

Name: CALCULATE_UK

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable unique key discovery.

Name: COPY_DATA

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable copying of data from source to profile workspace.

Name: DOMAIN_MAX_COUNT

Type: NUMBER

Valid Values: N/A

Default: 100

The maximum number of distinct values in a column in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values

Percent property.

Name: DOMAIN_MAX_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 50

The maximum number of distinct values in a column, expressed as a percentage of the total number of rows in the table, in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MIN_COUNT

Type: NUMBER

Valid Values: N/A

Default: 2

The minimum number of rows for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: DOMAIN_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 1

The minimum number of rows, expressed as a percentage of the total number of rows, for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: FD_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 100

This is the minimum percentage of rows that need to satisfy a functional dependency relationship.

Name: FD_UK_FK_LHS_COUNT

Type: NUMBER

Valid Values: N/A

Default: 1

This is the maximum number of attributes for unique key and functional dependency profiling.

Name: FK_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 75

This is the minimum percentage of rows that need to satisfy a foreign key relationship.

Name: MAX_NUM_PATTERNS

Type: NUMBER

Valid Values: N/A

Default: 10

This tells the profiler to get the top-N patterns for the attribute.

Name: NULL_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 10

If the percentage of null values in a column is less than this threshold percent, then that column will be discovered as a possible Not Null column.

Name: NULL_VALUE

Type: STRING

Valid Values: N/A

Default: null

This value will be considered as the null value when profiling. Please enclose the value in single quotes. An unquoted null (the current default value) will be considered a database null.

Name: REDUNDANT_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 75

This is the minimum percentage of rows that are redundant with respect to a foreign key-unique key pair.

Name: SAMPLE_RATE

Type: NUMBER

Valid Values: 0 - 100

Default: 100

This value will be the percent of total rows that will be randomly selected during loading.

Name: UK_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 75

This is the minimum percentage of rows that need to satisfy a unique key relationship.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceClause

A general clause for setting references for a Data Profile.

unsetReferenceClause

A general clause for unsetting references for a Data Profile.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a location for an existing Data Profile.

setReferenceIconSetClause

Set icon set for the Data Profile.

unsetReferenceLocationClause

Unset a location for an existing Data Profile.

unsetReferenceIconSetClause

Unset icon set for the Data Profile.

propertyValue

Value of a property.

Examples

```
OMBALTER DATA_PROFILE 'src_profile' RENAME TO 'tgt_profile' SET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a renamed data  
profile.', 'target profile')
```

This will rename the Data Profile "src_profile" to "tgt_profile", and set its description to "This becomes a renamed data profile.", set its business name to "target profile".

See Also

OMBALTER, OMBCREATE DATA_PROFILE, OMBDROP DATA_PROFILE

OMBALTER DATA_RULE

Purpose

= Alter the content of a data rule.

Prerequisites

1. The current context of scripting must be an DataRule Rule Module
2. No concurrent user should be modifying the data rule

Syntax

```
alterDataRuleCommand = OMBALTER ( DATA_RULE "QUOTED_STRING" (
    "ruleRenameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" | ( ADD "addDomainValueClause" )+ | (
        REMOVE "removeDomainValueClause" )+ | "alterGroupClause"+ |
    "alterDomainValueClause"+ ) )
ruleRenameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
addDomainValueClause = ( DOMAIN_VALUE "QUOTED_STRING" )
removeDomainValueClause = DOMAIN_VALUE "QUOTED_STRING"
alterGroupClause = GROUP "QUOTED_STRING" ( "setGroupPropertiesClause" |
    "addAttributeClause"+ | "removeAttributeClause"+ |
    "alterAttributeClause"+ )
alterDomainValueClause = DOMAIN_VALUE "QUOTED_STRING"
    "domainValueRenameClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
setGroupPropertiesClause = SET PROPERTIES "(" "propertyNameList" ")"
    VALUES "(" "propertyValueList" ")"
addAttributeClause = ADD ATTRIBUTE ( "QUOTED_STRING" [ SET
    "setPropertiesClause" ] )
removeAttributeClause = REMOVE ATTRIBUTE "QUOTED_STRING"
alterAttributeClause = ATTRIBUTE ( "QUOTED_STRING" SET
    "setPropertiesClause" )
domainValueRenameClause = RENAME TO "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterDataRuleCommand

Alter a data rule.

QUOTED_STRING

Data rule name.

ruleRenameClause

Rename a data rule.

QUOTED_STRING

New name.

alterPropertiesOrIconSetClause

Alter properties or the icon set.

addDomainValueClause

Add a domain value.

QUOTED_STRING

Domain value.

removeDomainValueClause

Remove a domain value.

QUOTED_STRING

Domain value.

alterGroupClause

Alter a data rule group.

QUOTED_STRING

Data rule group name.

alterDomainValueClause

Alter a domain value.

QUOTED_STRING

Domain value name.

setPropertiesClause

Set the properties of the data rule.

Properties of Data Rule are:

Basic properties for DATA_RULE:

Name: DATA_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Data Rule

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Data Rule

Name: RULE_TYPE

Type: STRING

Valid Values: ATTR_VALUE_RULE

DOMAIN_NO_NULL_RULE

DOMAIN_LIST_RULE

DOMAIN_PATTERN_LIST_RULE

DOMAIN_RANGE_RULE

DOMAIN_FORMAT_TELEPHONE_RULE

DOMAIN_FORMAT_IP_RULE

DOMAIN_FORMAT_SSN_RULE

DOMAIN_FORMAT_DATE_RULE

DOMAIN_FORMAT_NUMBER_RULE

DOMAIN_FORMAT_URL_RULE

DOMAIN_FORMAT_EMAIL_RULE

FUNCTIONAL_DEP_RULE

SET_RULE

IDENTITY_RULE

REFERENCE_RULE

Default: ATTR_VALUE_RULE

Data Rule type

Name: ATTR_VALUE_CLAUSE

Type: STRING(4000)

Valid Values: N/A

Default: "

Expression used in the ATTR_VALUE_RULE

Name: LOCAL_MIN_COUNT

Type: STRING

Valid Values: N/A

Default: 1

Used in the REFERENCE_RULE to set the max and min counts for the local and remote side of a relation. For a 1-n relationship, set the local max count to 1 and remote min count to 0 or 1. For an n-1 relationship, set the local max count to n and remote max count to 1.

Name: LOCAL_MAX_COUNT

Type: STRING

Valid Values: N/A

Default: n

Used in the REFERENCE_RULE to set the max and min counts for the local and remote side of a relation. For a 1-n relationship, set the local max count to 1 and remote min count to 0 or 1. For an n-1 relationship, set the local max count to n and remote max count to 1.

Name: REMOTE_MIN_COUNT

Type: STRING

Valid Values: N/A

Default: 0

Used in the REFERENCE_RULE to set the max and min counts for the local and remote side of a relation. For a 1-n relationship, set the local max count to 1 and remote min count to 0 or 1. For an n-1 relationship, set the local max count to n and remote max count to 1.

Name: REMOTE_MAX_COUNT

Type: STRING

Valid Values: N/A

Default: 'n'

Used in the REFERENCE_RULE to set the max and min counts for the local and remote side of a relation. For a 1-n relationship, set the local max count

to 1 and remote min count to 0 or 1. For an n-1 relationship, set the local max count to n and remote max count to 1.

Name: MAX_VALUE

Type: STRING

Valid Values: N/A

Default: "

Used in the DOMAIN_RANGE_RULE. Leave the min or max value blank is not used. For instance, the rule: attr >= 10 would have a max value and no min_value.

Name: MIN_VALUE

Type: STRING

Valid Values: N/A

Default: "

Used in the DOMAIN_RANGE_RULE. Leave the min or max value blank is not used. For instance, the rule: attr >= 10 would have a max value and no min_value.

Name: IGNORE_NULLS

Type: BOOLEAN

Valid Values: N/A

Default: false

Used in identity and reference rules. If set, then records containing null values will not be evaluated.

Properties of Data Rule Attribute are:

Name: DATATYPE

Type: STRING

Valid Values: DATE, NUMBER, VARCHAR2

Default: NUMBER

Set the data type for Parameter. Should be set for attributes on ATTR_VALUE and DOMAIN_RANGE rules

Properties for DATA_RULE:

Name: ATTR_VALUE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

Description not available.

Name: FUNCTIONAL_DEP_THRESHOLD

Type: NUMBER

Valid Values: N/A

Default: 0

Description not available.

Name: IGNORE_NULLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LOCAL_MAX_COUNT

Type: STRING

Valid Values: N/A

Default: n

Description not available.

Name: LOCAL_MIN_COUNT

Type: STRING

Valid Values: N/A

Default: 1

Description not available.

Name: MAX_VALUE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: MIN_VALUE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: NAMEADDR_PASS_CONDITION

Type: STRING

Valid Values: PASS_PARSED, PASS_POSTALMATCHED_AVAIL

Default: PASS_PARSED

Description not available.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Description not available.

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD,
NA_NOR,
NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS,
NA_SGP,
NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF
Default: NA_USA

Description not available.

Name: REMOTE_MAX_COUNT

Type: STRING

Valid Values: N/A

Default: n

Description not available.

Name: REMOTE_MIN_COUNT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: RULE_TYPE

Type: STRING

Valid Values: , ATTR_VALUE_RULE, DOMAIN_FORMAT_DATE_RULE,
DOMAIN_FORMAT_EMAIL_RULE, DOMAIN_FORMAT_IP_RULE, DOMAIN_
FORMAT_NUMBER_RULE,
DOMAIN_FORMAT_SSN_RULE, DOMAIN_FORMAT_TELEPHONE_RULE,
DOMAIN_FORMAT_URL_RULE, DOMAIN_LIST_RULE, DOMAIN_NO_NULL_
RULE,
DOMAIN_PATTERN_LIST_RULE, DOMAIN_RANGE_RULE, FUNCTIONAL_DEP_
RULE,
IDENTITY_RULE, NAMEADDRESS_RULE, REFERENCE_RULE, SET_RULE

Default: "

Description not available.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set icon set for the data rule.

unsetReferenceIconSetClause

Unset icon set for the data rule.

setGroupPropertiesClause

Set the properties in a group.

addAttributeClause

Add an attribute to a group.

removeAttributeClause

Remove an attribute from a group.

alterAttributeClause

ALter an attribute.

domainValueRenameClause

Rename a domain value.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBALTER DATA_RULE 'NOT_NULL'  
SET PROPERTIES(ATTR_VALUE_CLAUSE)  
VALUES('"THIS"."ATTR1" is not null')
```

```
OMBALTER DATA_RULE 'STATE_NAME'  
ADD DOMAIN_VALUE 'MN'  
ADD DOMAIN_VALUE 'WISC'
```

REMOVE DOMAIN_VALUE "WT"

OMBALTER DATA_RULE 'NOT_NULL' GROUP 'THIS' ATTRIBUTE 'VALUE' SET
PROPERTIES

(DATATYPE) VALUES ('VARCHAR2')

See Also

OMBALTER, OMBCREATE DATA_RULE, OMBDROP DATA_RULE

OMBALTER DATA_RULE_MODULE

Purpose

To create a data rule module.

Prerequisites

Should be in the context of project.

Syntax

```
alterDataRuleModuleCommand = OMBALTER ( DATA_RULE_MODULE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterDataRuleModuleCommand

Alter data rule module.

QUOTED_STRING

Data rule module name.

renameClause

Rename a data rule.

QUOTED_STRING

New name.

alterPropertiesOrIconSetClause

Alter properties or the icon set.

setPropertiesClause

Set the properties of the data rule module.

Basic properties for DATA_RULE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Data Rule Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Data Rule Module

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set icon set for the data rule module.

unsetReferenceIconSetClause

Unset icon set for the data rule module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBALTER DATA_RULE_MODULE 'br_mod' SET PROPERTIES (DESCRIPTION)
VALUES
```

('this is new description')

This will alter the description of a data rule module named "br_module"

See Also

OMBALTER, OMBCREATE DATA_RULE_MODULE, OMBDROP DATA_RULE_MODULE

OMBALTER DEPLOYMENT

Purpose

Alter the Deployment by renaming it, and/or reset its properties and referenced Control Center.

Prerequisites

Should be in the context of a Configuration.

Syntax

```
alterDeploymentCommand = OMBALTER ( DEPLOYMENT "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] [
    "setReferenceControlCenterClause" ] [ "setAsActiveClause" ] | SET
    "setPropertiesClause" [ "setReferenceControlCenterClause" ] [
    "setAsActiveClause" ] | "setReferenceControlCenterClause" [
    "setAsActiveClause" ] | "setAsActiveClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceControlCenterClause = SET ( REFERENCE | REF ) CONTROL_CENTER
    "QUOTED_STRING"
setAsActiveClause = SET AS ACTIVE
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterDeploymentCommand

This command modifies an existing Deployment.

renameClause

Rename Deployment.

setPropertiesClause

Associate a set of properties with the existing Deployment.

Basic properties for DEPLOYMENT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Deployment

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Deployment.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

setReferenceControlCenterClause

Set a reference of a Control Center to the existing Deployment.

setAsActiveClause

Set this Deployment as Active.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBALTER DEPLOYMENT 'QA_DEPLOYMENT' RENAME TO 'QA_MY_
DEPLOYMENT' SET REF
```

```
CONTROL_CENTER 'QA_MY_CC' SET AS ACTIVE
```

This will rename the Deployment "QA_DEPLOYMENT" to "QA_MY_DEPLOYMENT", set

reference to Control Center "QA_MY_CC", set this Deployment as Active.

See Also

OMBALTER, OMBCREATE DEPLOYMENT

OMBALTER DEPLOYMENT_ACTION_PLAN

Purpose

Modify an existing deployment action plan.

Prerequisites

There must be a current working project.

Syntax

```
AlterActionPlanCommand = ( OMBALTER ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ( (
    "renameActionPlanClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "addActionClause" { "addActionClause" |
    "modifyActionClause" | "deleteActionClause" } ) | (
    "modifyActionClause" { "addActionClause" | "modifyActionClause" |
    "deleteActionClause" } ) | ( "deleteActionClause" { "addActionClause"
    | "modifyActionClause" | "deleteActionClause" } ) ) )
renameActionPlanClause = RENAME TO "QUOTED_STRING"
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
modifyActionClause = MODIFY ACTION "QUOTED_STRING" ( (
    "renameActionClause" [ "setUnsetClause" ] ) | "setUnsetClause" )
deleteActionClause = DELETE ACTION "QUOTED_STRING"
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
    "setReferenceClause"
renameActionClause = RENAME TO "QUOTED_STRING"
setUnsetClause = ( ( SET "setClauseForAlter" ) | ( UNSET
    "unsetReferenceClause" ) )
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
    "useClause" }
setClauseForAlter = ( "propertiesClause" [ SET "setReferenceClause" |
    UNSET "unsetReferenceClause" ] ) | "setReferenceClause"
unsetReferenceClause = ( REF | REFERENCE )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

AlterActionPlanCommand

Modify an existing deployment action plan.

renameActionPlanClause

Rename an action plan.

addActionClause

Add an action to an action plan.

modifyActionCode

Modify an action of an action plan.

deleteActionCode

Remove an action from an action plan.

setCode

Set the properties of an action and/or associate an object with an action.

renameActionCode

Rename an action.

setUnsetCode

Set the properties and/or associate/disassociate an object with an action.

propertiesCode

Associate a set of properties with an action.

PROPERTIES

The only valid property is OPERATION, which specifies the type of action to be taken.

setReferenceCode

Associate an object with an action.

ObjType

Object type. Valid values are ADVANCED_QUEUE, CUBE, DIMENSION, EXTERNAL_TABLE, CONNECTOR, FUNCTION, MAPPING, MATERIALIZED_VIEW, PROCEDURE, PROCESS_FLOW_PACKAGE, SCHEDULABLE, SEQUENCE, TABLE, and VIEW.

QUOTED_STRING

Absolute or relative path name of an object (for example '/MY_PROJECT/MODULE_X/TABLE_Y').

setCodeForAlter

Set the properties and/or associate/disassociate an object with an action.

unsetReferenceClause

Disassociate a previously associated object from an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Currently used only in CORRECTION_SCHEMA_ACTION_PLAN.

propertyValue

Value of a property. Valid values for OPERATION are DROP, CREATE, REPLACE and UPGRADE.

Examples

```
OMBALTER DEPLOYMENT_ACTION_PLAN 'MY_PLAN' RENAME TO 'MY_
PLAN_2'
```

```
OMBALTER DEPLOYMENT_ACTION_PLAN 'MY_PLAN' ADD ACTION 'MY_
ALTER_TABLE'
```

```
SET PROPERTIES (OPERATION) VALUES ('CREATE') SET REFERENCE TABLE
'TABLE_X'
```

```
OMBALTER DEPLOYMENT_ACTION_PLAN 'MY_PLAN' MODIFY ACTION 'MY_
VIEW_CREATE'
```

```
RENAME TO 'MY_VIEW_DROP' SET PROPERTIES (OPERATION) VALUES ('DROP')
```

```
OMBALTER DEPLOYMENT_ACTION_PLAN 'MY_PLAN' DELETE ACTION 'MY_
TABLE_DEPLOY'
```

```
OMBALTER DEPLOYMENT_ACTION_PLAN 'MY_PLAN'
```

```
ADD ACTION 'MY_ALTER_VIEW' SET PROPERTIES (OPERATION) VALUES
('CREATE')
```

```
SET REFERENCE VIEW '/MY_PROJECT/MY_MODULE/VIEW_Y'
```

```
MODIFY ACTION 'MY_TABLE_DEPLOY' SET REFERENCE TABLE 'MY_
MODULE/TABLE_Z'
```

DELETE ACTION 'MY_VIEW_CREATE'

See Also

OMBCREATE DEPLOYMENT_ACTION_PLAN, OMBDEPLOY

OMBALTER DIMENSION

Purpose

This command alters a dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
alterDimensionCommand = OMBALTER DIMENSION "dimensionName" (
    "renameClause" [ "setPropertiesClause" ] [ "setReferenceIconSetClause"
    [ "setDimensionKeySequenceClause" ] | "unsetReferenceIconSetClause" [
    "setDimensionKeySequenceClause" ] | "setDimensionKeySequenceClause" ]
    [ ( ADD "addSurrogateKeyDimensionAttributeClause" | MODIFY
    "modifySurrogateKeyDimensionAttributeClause" ) ] [ ( ADD
    "addParentKeyDimensionAttributeClause" | MODIFY
    "modifyParentKeyDimensionAttributeClause" ) ] [ ( ADD
    "addBusinessKeyDimensionAttributeClause" | MODIFY
    "modifyBusinessKeyDimensionAttributeClause" ) ] [
    "alterDimensionDescendantsClause" ] [ "alterDimensionBindingClause" ]
    | "setPropertiesClause" [ "setReferenceIconSetClause" [
    "setDimensionKeySequenceClause" ] | "unsetReferenceIconSetClause" [
    "setDimensionKeySequenceClause" ] | "setDimensionKeySequenceClause" ]
    [ ( ADD "addSurrogateKeyDimensionAttributeClause" | MODIFY
    "modifySurrogateKeyDimensionAttributeClause" ) ] [ ( ADD
    "addParentKeyDimensionAttributeClause" | MODIFY
    "modifyParentKeyDimensionAttributeClause" ) ] [ ( ADD
    "addBusinessKeyDimensionAttributeClause" | MODIFY
    "modifyBusinessKeyDimensionAttributeClause" ) ] [
    "alterDimensionDescendantsClause" ] [ "alterDimensionBindingClause" ]
    | ( "setReferenceIconSetClause" [ "setDimensionKeySequenceClause" ] |
    "unsetReferenceIconSetClause" [ "setDimensionKeySequenceClause" ] |
    "setDimensionKeySequenceClause" ) [ ( ADD
    "addSurrogateKeyDimensionAttributeClause" | MODIFY
    "modifySurrogateKeyDimensionAttributeClause" ) ] [ ( ADD
    "addParentKeyDimensionAttributeClause" | MODIFY
    "modifyParentKeyDimensionAttributeClause" ) ] [ ( ADD
    "addBusinessKeyDimensionAttributeClause" | MODIFY
    "modifyBusinessKeyDimensionAttributeClause" ) ] [
    "alterDimensionDescendantsClause" ] [ "alterDimensionBindingClause" ]
    | ( ADD "addSurrogateKeyDimensionAttributeClause" | MODIFY
    "modifySurrogateKeyDimensionAttributeClause" ) [ ( ADD
    "addParentKeyDimensionAttributeClause" | MODIFY
    "modifyParentKeyDimensionAttributeClause" ) ] [ ( ADD
    "addBusinessKeyDimensionAttributeClause" | MODIFY
    "modifyBusinessKeyDimensionAttributeClause" ) ] [
    "alterDimensionDescendantsClause" ] [ "alterDimensionBindingClause" ]
    | ( ADD "addParentKeyDimensionAttributeClause" | MODIFY
    "modifyParentKeyDimensionAttributeClause" ) [ ( ADD
    "addBusinessKeyDimensionAttributeClause" | MODIFY
    "modifyBusinessKeyDimensionAttributeClause" ) ] [
    "alterDimensionDescendantsClause" ] [ "alterDimensionBindingClause" ]
    | ( ADD "addBusinessKeyDimensionAttributeClause" | MODIFY
    "modifyBusinessKeyDimensionAttributeClause" ) [
    "alterDimensionDescendantsClause" ] [ "alterDimensionBindingClause" ]
    | "alterDimensionDescendantsClause" [ "alterDimensionBindingClause" ]
```

```

    | "alterDimensionBindingClause" | "alterDimensionUnBindingClause" )
dimensionName = "QUOTED_STRING"
renameClause = RENAME TO "QUOTED_STRING"
setPropertyClause = SET PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = SET ( REFERENCE | REF ) ICONSET
    "QUOTED_STRING"
setDimensionKeySequenceClause = SET ( REF | REFERENCE ) SEQUENCE
    "sequenceName"
unsetReferenceIconSetClause = UNSET ( REFERENCE | REF ) ICONSET
addSurrogateKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS SURROGATE_KEY [ "setPropertyClause"
]
modifySurrogateKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS SURROGATE_KEY [ "setPropertyClause"
]
addParentKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS PARENT_KEY [ "setPropertyClause" ]
modifyParentKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS PARENT_KEY [ "setPropertyClause" ]
addBusinessKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS BUSINESS_KEY [ "setPropertyClause" ]
modifyBusinessKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS BUSINESS_KEY [ "setPropertyClause" ]
alterDimensionDescendantsClause = ( ADD ( "addDimensionAttributeClause" |
    "addDimensionRoleClause" | "addLevelClause" |
    "addLevelAttributeClause" | "addHierarchyClause" |
    "addSkipLevelsClause" ) | MODIFY ( "modifyDimensionAttributeClause" |
    "modifyDimensionRoleClause" | "modifyLevelClause" |
    "modifyLevelAttributeClause" | "modifyHierarchyClause" |
    "modifySkipLevelsClause" ) | DELETE ( "dimensionAttributeLocator" |
    "roleLocator" | "levelLocator" | "levelAttributeBottomUpLocator" |
    "hierarchyLocator" | "skipLevelLocator" ) )+
alterDimensionBindingClause = IMPLEMENTED BY ( SYSTEM ( STAR | SNOWFLAKE
) | STAR ( ( [ DIMENSION_KEY BOUND TO COLUMN "columnName" ] (
    "levelBindingClause" | "levelAttributeBindingClause" |
    "setBindRelationshipClause" | "setSkipBindRelationshipClause" )+ ) ) |
    SNOWFLAKE ( ( "levelBindingClause" | "levelAttributeBindingClause" |
    "setBindRelationshipClause" | "setSkipBindRelationshipClause" )+ ) )
alterDimensionUnBindingClause = DELETE BINDING
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
sequenceName = "QUOTED_STRING"
dimensionAttributeName = "QUOTED_STRING"
addDimensionAttributeClause = DIMENSION_ATTRIBUTE "dimensionAttributeName"
    [ SET AS BUSINESS_KEY ] [ "setPropertyClause" ]
addDimensionRoleClause = DIMENSION_ROLE "roleName" [ "setPropertyClause"
]
addLevelClause = LEVEL "levelName" [ "setPropertyClause" ]
addLevelAttributeClause = LEVEL_ATTRIBUTE "levelAttributeName" OF
    "levelLocator" [ "setPropertyClause" ]
    "setLevelAttributeReferenceClause"
addHierarchyClause = HIERARCHY "hierarchyName" [ "setPropertyClause" ] [
    "hierarchyLevelReferenceClause" ]
addSkipLevelsClause = SKIP_LEVELS FROM "levelLocator" TO "levelLocator" IN
    "hierarchyLocator"
modifyDimensionAttributeClause = "dimensionAttributeLocator" (
    "renameClause" | "setPropertyClause" )
modifyDimensionRoleClause = "roleLocator" ( "renameClause" |
    "setPropertyClause" )

```

```

modifyLevelClause = "levelLocator" ( "renameClause" |
    "setPropertyClause" )
modifyLevelAttributeClause = "levelAttributeBottomUpLocator" (
    "renameClause" | "setPropertyClause" |
    "setLevelAttributeReferenceClause" )
modifyHierarchyClause = "hierarchyLocator" ( "renameClause" |
    "setPropertyClause" | "hierarchyLevelReferenceClause" )
modifySkipLevelsClause = SKIP_LEVELS FROM "levelLocator" TO "levelLocator"
    IN "hierarchyLocator"
dimensionAttributeLocator = DIMENSION_ATTRIBUTE "dimensionAttributeName"
roleLocator = DIMENSION_ROLE "roleName"
levelLocator = LEVEL "levelName"
levelAttributeBottomUpLocator = LEVEL_ATTRIBUTE "levelAttributeName" OF
    "levelLocator"
hierarchyLocator = HIERARCHY "hierarchyName"
skipLevelLocator = SKIP_LEVELS FROM "levelLocator" TO "levelLocator" IN
    "hierarchyLocator"
columnName = "QUOTED_STRING"
levelBindingClause = "levelLocator" BOUND TO ( TABLE "tableName" | VIEW
    "viewName" )
levelAttributeBindingClause = LEVEL_ATTRIBUTE "levelAttributeName" OF
    LEVEL "levelName" BOUND TO COLUMN "columnName"
setBindRelationshipClause = LEVEL_RELATIONSHIP OF "levelLocator" IN
    "hierarchyLocator" BOUND TO COLUMN "columnName"
setSkipBindRelationshipClause = SKIP_LEVEL_RELATIONSHIP OF "levelLocator"
    IN "hierarchyLocator" BOUND TO COLUMN "columnName"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
roleName = "QUOTED_STRING"
levelName = "QUOTED_STRING"
levelAttributeName = "QUOTED_STRING"
setLevelAttributeReferenceClause = SET ( REF | REFERENCE ) (
    "dimensionAttributeLocator" [ TYPE_THREE_SCD_PREVIOUS
    "levelAttributeLocator" ] )
hierarchyName = "QUOTED_STRING"
hierarchyLevelReferenceClause = SET ( REF | REFERENCE ) LEVELS "(
    "levelName" { "," "levelName" } )"
tableName = "QUOTED_STRING"
viewName = "QUOTED_STRING"
levelAttributeLocator = LEVEL_ATTRIBUTE "levelAttributeName"

```

Keywords And Parameters

setPropertyClause

Basic properties for DIMENSION, DIMENSION_ATTRIBUTE, LEVEL, LEVEL_ATTRIBUTE

and HIERARCHY:

Basic properties for DIMENSION :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Dimension

Name: SCD_TYPE

Type: INTEGER

Valid Values: 1, 2, 3

Default: 1

Slowly changing policy to be applied on the dimension. Give Integer values

1, 2, 3

for Slowly changing type one, two and three

Name: TYPE

Type: STRING

Valid Values: 'NONE', 'TIME'

Default: 'NONE'

'NONE' it does not recognize it as any specific type of dimension.

'TIME' dimension is a time dimension

Name: OLAP_TYPE

Type: STRING

Valid Values: NONE, TIME

Default: NONE

Dimension type for OLAP, get regular dimension 'NONE' and for OLAP Time

Dimension 'TIME'

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The dimension is visible to OLAP end user

Name: UNIQUE_KEY_CONSTRAINT

Type: STRING

Valid Values: true, false

Default: false

set the Unique Key constraint on the Business Key

Name: STORAGE

Type: STRING

Valid Values: RELATIONAL, AW

Default: 'RELATIONAL'

The storage of a dimension can be AW or relational

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the analytical workspace name where the dimension is implemented

Name: AW_DIMENSION_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace dimension physical object name

Name: USE_BUSINESS_KEYS

Type: STRING

Valid Values: true, false

Default: false

Set the flag for Analytical Workspace dimension to use Business Keys as data source

Basic properties for DIMENSION_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension_Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Dimension_Attribute

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH

LOCAL TIME ZONE, VARCHAR, VARCHAR2

Default: NUMBER

Sets the datatype of a Dimension_Attribute

Name: SCALE

Type: NUMBER

Valid Values: -85 - 125

Default: 1

The scale of a number.

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 39

Default: 1

The precision of a number.

Name: DESCRIPTOR

Type: STRING

Valid Values: NONE, SHORT_DESCRIPTION, LONG_DESCRIPTION, END_DATE, TIME_SPAN, PRIOR_PERIOD, YEAR_AGO_PERIOD

Default: NONE

The following properties are set on dimension attribute so that it is recognized by OLAP service

'NONE' it is not specially recognized type by OLAP service

'SHORT_DESCRIPTION' sets as a short description

'LONG_DESCRIPTION' sets as a long description

'END_DATE' sets as a last date of a period.

'TIME_SPAN' sets as a number of days in a period.

'PRIOR_PERIOD' sets as the prior period number.

'YEAR_AGO_PERIOD' sets as the time period a year before this period

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The dimension attribute is visible to OLAP end user

Name: TYPE

Type: STRING

Valid Values: NONE, START_DATE, END_DATE, TIME_SPAN

Default: NONE

'NONE' dimension attribute so OWB does not recognize it as any specific type.

'START_DATE' dimension attribute of time dimension as the start date of a period

'END_DATE' dimension attribute of time dimension as the end date of a period

'TIME_SPAN' dimension attribute of time dimension as the time span

Name: AW_ATTRIBUTE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

set the AW object name implementing the dimension attribute

Basic properties for Level :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Level

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Level

Name: TYPE

Type: STRING

Valid Values: NONE, DAY, FISCAL_WEEK, FISCAL_MONTH, FISCAL_QUARTER, FISCAL_YEAR, CALENDAR_WEEK, CALENDAR_MONTH, CALENDAR_QUARTER, CALENDAR_YEAR

Default: NONE

For regular relational dimension level (non-time dimension level) use 'NONE'. For relational time dimension use other values.

Name: OLAP_TYPE

Type: STRING

Valid Values: NONE, DAY, MONTH, QUARTER, YEAR, TOTAL

Default: NONE

Level has an olap-type for OLAP-based levels; use 'NONE' for regular levels, and other values for OLAP Time Dimension

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The level is visible to OLAP end user

Basic properties for Level_Attribute :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Level_Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Level_Attribute

Name: DEFAULT_VALUE

Type: STRING(200)

Valid Values: N/A

Default: "

This is used to construct the default parent policy for loading data into dimension.

Name: OLAP_TYPE

Type: STRING

Valid Values: NONE, SHORT_DESCRIPTION, END_DATE, TIME_SPAN, PRIOR_PERIOD,

YEAR_AGO_PERIOD

Default: NONE

The following properties are set on level attribute so that it is recognized by OLAP service

'NONE' it is not specially recognized type by OLAP service

'SHORT_DESCRIPTION' sets as a short description

'END_DATE' sets as a last date of a period.

'TIME_SPAN' sets as a number of days in a period.

'PRIOR_PERIOD' sets as the prior period number.

'YEAR_AGO_PERIOD' sets as the time period a year before this period

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The level attribute is visible to OLAP end user

Name: TYPE

Type: STRING

Valid Values: NONE, START_DATE, END_DATE, TIME_SPAN

Default: 'NONE'

'NONE' level attribute so OWB does not recognize it as any specific type.

'START_DATE' level attribute of time dimension as the start date of a period

'END_DATE' level attribute of time dimension as the end date of a period

'TIME_SPAN' level attribute of time dimension as the time span

Name: TYPE_TWO_SCD_EFFECTIVE_DATE

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Effective Date for Slowly changing type 2

Name: TYPE_TWO_SCD_EXPIRATION_DATE

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Expiration Date for Slowly changing type

2

Name: TYPE_TWO_SCD_TRIGGER

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Trigger for saving history for Slowly changing type 2

Name: TYPE_THREE_SCD_EFFECTIVE_DATE

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Effective Date for Slowly changing type 3

Basic properties for Hierarchy :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Hierarchy

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Hierarchy

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The hierarchy is visible to OLAP end user

Name: TYPE

Type: STRING

Valid Values: NONE, FISCAL, CALENDAR_YEAR, CALENDAR_WEEK

Default: NONE

'NONE' hierarchy so OWB does not recognize it as any specific type.

'FISCAL' fiscal hierarchy for time dimension

'CALENDAR_YEAR' calendar year hierarchy time dimension

'CALENDAR_WEEK' calendar week hierarchy time dimension

Name: DEFAULT_DISPLAY

Type: STRING

Valid Values: true, false

Default: false

The hierarchy is set as Default display hierarchy

Name: VALUE_BASED

Type: STRING

Valid Values: true, false

Default: false

Sets the flag to define a Value Based Hierarchy for AW only

Properties for DIMENSION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS

Type: STRING

Valid Values: DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY, DEPLOY_TO_CATALOG_ONLY

Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for Dimension, they are DDL

Scripts for Relational Dimensional or Scripts for ROLAP or or Scripts for AW.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: VIEW_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Name of the view that is generated to hide the control rows on the dimension implementation table of a star schema. If this field is left blank, the view name will default to '<Name of Dimension>_v'

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Dimension is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```

OMBCREATE DIMENSION 'PRODUCT'
ADD DIMENSION_ATTRIBUTE 'ID' SET AS SURROGATE_KEY
OMBALTER DIMENSION 'PRODUCT' ADD DIMENSION_ATTRIBUTE 'KEY' SET
AS
BUSINESS_KEY
SET PROPERTIES (DATATYPE,LENGTH) VALUES ('VARCHAR2',16)
OMBALTER DIMENSION 'PRODUCT' ADD DIMENSION_ATTRIBUTE 'LONG_
DESC'
SET PROPERTIES (DATATYPE,LENGTH) VALUES ('VARCHAR2',55)

```

```
OMBALTER DIMENSION 'PRODUCT' ADD DIMENSION_ATTRIBUTE 'SHORT_
DESC'
SET PROPERTIES (DATATYPE,LENGTH) VALUES ('VARCHAR2',40)
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL 'EQUIPMENT_PARTS'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'EPT_KEY' OF
LEVEL
'EQUIPMENT_PARTS'
SET REF DIMENSION_ATTRIBUTE 'KEY'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'EPT_LONG_
DESC' OF LEVEL
'EQUIPMENT_PARTS'
SET REF DIMENSION_ATTRIBUTE 'LONG_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'EPT_SHORT_
DESC' OF LEVEL
'EQUIPMENT_PARTS'
SET REF DIMENSION_ATTRIBUTE 'SHORT_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'EPT_ID' OF
LEVEL
'EQUIPMENT_PARTS'
SET REF DIMENSION_ATTRIBUTE 'ID'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL 'COMPONENTS'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'CPT_KEY' OF
LEVEL
'COMPONENTS'
SET REF DIMENSION_ATTRIBUTE 'KEY'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'CPT_LONG_
DESC' OF LEVEL
'COMPONENTS'
SET REF DIMENSION_ATTRIBUTE 'LONG_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'CPT_SHORT_
DESC' OF LEVEL
'COMPONENTS'
SET REF DIMENSION_ATTRIBUTE 'SHORT_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'CPT_ID' OF
LEVEL
'COMPONENTS'
SET REF DIMENSION_ATTRIBUTE 'ID'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL 'DIVISIONS'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'DVN_KEY' OF
LEVEL
```

```

'DIVISIONS'
SET REF DIMENSION_ATTRIBUTE 'KEY'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'DVN_LONG_
DESC' OF LEVEL
'DIVISIONS'
SET REF DIMENSION_ATTRIBUTE 'LONG_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'DVN_SHORT_
DESC' OF LEVEL
'DIVISIONS'
SET REF DIMENSION_ATTRIBUTE 'SHORT_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'DVN_ID' OF
LEVEL
'DIVISIONS'
SET REF DIMENSION_ATTRIBUTE 'ID'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL 'TOTAL_PRODUCTS'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'TPT_KEY' OF
LEVEL
'TOTAL_PRODUCTS'
SET REF DIMENSION_ATTRIBUTE 'KEY'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'TPT_LONG_
DESC' OF LEVEL
'TOTAL_PRODUCTS'
SET REF DIMENSION_ATTRIBUTE 'LONG_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'TPT_SHORT_
DESC' OF LEVEL
'TOTAL_PRODUCTS'
SET REF DIMENSION_ATTRIBUTE 'SHORT_DESC'
OMBALTER DIMENSION 'PRODUCT' ADD LEVEL_ATTRIBUTE 'TPT_ID' OF
LEVEL
'TOTAL_PRODUCTS'
SET REF DIMENSION_ATTRIBUTE 'ID'
OMBALTER DIMENSION 'PRODUCT' ADD HIERARCHY 'PRODUCT_
HIERARCHY'
SET REF LEVELS
('TOTAL_PRODUCTS','DIVISIONS','COMPONENTS','EQUIPMENT_PARTS')
OMBALTER DIMENSION 'PRODUCT' IMPLEMENTED BY SYSTEM STAR

```

See Also

OMBALTER, OMBCREATE DIMENSION, OMBRETRIEVE DIMENSION, OMBDROP DIMENSION

OMBALTER DRILL_PATH

Purpose

Alters a drill path.

Prerequisites

Should be in the context of a business definition module or use the full path.

Syntax

```
alterDrillPathCommand = ( OMBALTER DRILL_PATH "QUOTED_STRING" ( (
    "renameClause" [ SET "setPropertiesClause" ] [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterDrillPathLevelClauses" } ) ) | ( SET "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterDrillPathLevelClauses" } ) ) | ( SET
    "setReferenceIconSetClause" [ UNSET "unsetReferenceIconSetClause" ] {
    "alterDrillPathLevelClauses" } ) ) | ( UNSET
    "unsetReferenceIconSetClause" { "alterDrillPathLevelClauses" } ) ) | (
    "alterDrillPathLevelClauses" { "alterDrillPathLevelClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterDrillPathLevelClauses = ADD "addDrillPathLevelClauseForAlter" |
    MODIFY "modifyDrillPathLevelClause" | DELETE
    "deleteDrillPathLevelClause"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
addDrillPathLevelClauseForAlter = DRILL_LEVEL "QUOTED_STRING" [ BELOW
    DRILL_LEVEL "QUOTED_STRING" ] [ SET "setPropertiesClause" ] {
    "setDrillItemClause" }
modifyDrillPathLevelClause = DRILL_LEVEL "QUOTED_STRING" [ "renameClause"
    ] [ "moveDrillLevelToClause" ] [ SET "setPropertiesClause" ] {
    "setDrillItemClause" | "unsetDrillItemClause" | "unsetJoinUsage" }
deleteDrillPathLevelClause = DRILL_LEVEL "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setDrillItemClause = SET ( REF | REFERENCE ) ITEM "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING" "itemJoinUsages"
moveDrillLevelToClause = MOVE ( UP | DOWN )
unsetDrillItemClause = UNSET ( REF | REFERENCE ) ITEM "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING"
unsetJoinUsage = UNSET ( REF | REFERENCE ) USING JOIN "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING"
itemJoinUsages = { SET ( REF | REFERENCE ) USING JOIN "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING" }
```

Keywords And Parameters

alterDrillPathCommand

This clause alters a drill path.

QUOTED_STRING

name of the drill path.

renameClause

Renames a drill path with a different name.

setPropertyClause

This clause sets the properties of the object.

Basic properties for DRILL_PATH:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill path

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill path

Basic properties for DRILL_LEVEL:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill level

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill level

Properties for DRILL_PATH:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterDrillPathLevelClauses

This clause modifies a drill path.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

addDrillPathLevelClauseForAlter

This adds a drill level to a drill path.

QUOTED_STRING

name of the drill level.

modifyDrillPathLevelClause

This modifies a drill level in a drill path.

QUOTED_STRING

name of the drill level.

deleteDrillPathLevelClause

This deletes a drill level from a drill path.

QUOTED_STRING

name of the drill level.

propertyValue

This is a property value.

setDrillItemClause

This clause adds a reference to an item for the level.

moveDrillLevelToClause

This clause is for positioning a drill level in a drill path.

unsetDrillItemClause

This clause removes a reference to an item for the level.

unsetJoinUsage

removes a join usage.

itemJoinUsages

The specific joins to be used.

Examples

```
OMBALTER DRILL_PATH 'DP1' SET PROPERTIES (DESCRIPTION) VALUES ('DP1')
```

See Also

```
OMBCREATE DRILL_PATH, OMBRETRIEVE DRILL_PATH
```

OMBALTER DRILL_TO_DETAIL

Purpose

Alters a drill to detail.

Prerequisites

Should be in the context of a business definition module or use the full path.

Syntax

```
alterDrillToDetailCommand = ( OMBALTER DRILL_TO_DETAIL "QUOTED_STRING" ( (
    "renameClause" [ SET "setPropertiesClause" ] [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterDrillToDetailClauses" } ) | ( SET "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterDrillToDetailClauses" } ) | ( SET "setReferenceIconSetClause"
    [ UNSET "unsetReferenceIconSetClause" ] { "alterDrillToDetailClauses"
    } ) ) | ( UNSET "unsetReferenceIconSetClause" {
    "alterDrillToDetailClauses" } ) | ( "alterDrillToDetailClauses" {
    "alterDrillToDetailClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterDrillToDetailClauses = "addDrillToDetailReferenceClause" | UNSET (
    REF | REFERENCE ) "deleteDrillToDetailClauses"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addDrillToDetailReferenceClause = SET ( REF | REFERENCE ) ITEM
    "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
deleteDrillToDetailClauses = ( ITEM "QUOTED_STRING" OF ITEM_FOLDER
    "QUOTED_STRING" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterDrillToDetailCommand

This clause alters a drill to detail.

QUOTED_STRING

name of the drill to detail.

renameClause

Renames a drill to detail with a different name.

setPropertiesClause

This clause sets the properties of the object.

Basic properties for DRILL_TO_DETAIL:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill to detail

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill to detail

Properties for DRILL_TO_DETAIL:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterDrillToDetailClauses

This clause modifies a drill to detail.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

addDrillToDetailReferenceClause

This adds a reference to an item to a drill to detail.

deleteDrillToDetailClauses

This deletes a reference to an item from a drill to detail.

propertyValue

This is a property value.

Examples

```
OMBALTER DRILL_TO_DETAIL 'DRILL1' SET PROPERTIES (DESCRIPTION)
VALUES
('DRILL1')
```

See Also

OMBCREATE DRILL_TO_DETAIL, OMBRETRIEVE DRILL_TO_DETAIL

OMBALTER EXPERT

Purpose

To alter an expert.

Prerequisites

In the context of an expert module.

Syntax

```

alterExpertCommand = OMBALTER EXPERT "QUOTED_STRING" ( "renameClause" [
    "alterPropertiesOrIconSetClause" ] { "alterExpertDetailClauses" } |
    "alterPropertiesOrIconSetClause" { "alterExpertDetailClauses" } |
    "alterExpertDetailClauses"+ )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterExpertDetailClauses = ( ADD ( "addParameterClause" |
    "addVariableClause" | "addNestedExpertClause" | "addTaskClause" |
    "addTransitionClause" ) | MODIFY ( "modifyParameterClause" |
    "modifyVariableClause" | "modifyTaskClause" | "modifyTransitionClause"
    ) | DELETE ( "deleteParameterClause" | "deleteVariableClause" |
    "deleteTaskClause" | "deleteTransitionClause" ) )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addParameterClause = PARAMETER "QUOTED_STRING" [ OF TASK "QUOTED_STRING" ]
    [ SET "setPropertiesClause" ] [ "parameterBindingClause" ]
addVariableClause = VARIABLE "QUOTED_STRING" [ SET "setPropertiesClause" ]
addNestedExpertClause = NESTED_EXPERT TASK "QUOTED_STRING" SET [
    "collectPropertiesClause" SET ] ( REF | REFERENCE ) EXPERT
    "QUOTED_STRING" [ SET "setReferenceIconSetClause" ]
addTaskClause = "TASK_TYPE" TASK "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ]
addTransitionClause = TRANSITION "QUOTED_STRING" FROM TASK "QUOTED_STRING"
    TO "QUOTED_STRING" [ SET "setPropertiesClause" ]
modifyParameterClause = "parameterLocator" ( "renameClause" [ SET
    "setPropertiesClause" ] [ "parameterBindingClause" ] | SET
    "setPropertiesClause" [ "parameterBindingClause" ] |
    "parameterBindingClause" )
modifyVariableClause = "variableLocator" ( "renameClause" [ SET
    "setPropertiesClause" ] | SET "setPropertiesClause" )
modifyTaskClause = TASK "QUOTED_STRING" ( "renameClause" [
    "alterPropertiesOrIconSetClause" ] | "alterPropertiesOrIconSetClause"
    )
modifyTransitionClause = TRANSITION "QUOTED_STRING" ( "renameClause" [ SET
    "setPropertiesClause" ] | SET "setPropertiesClause" )
deleteParameterClause = "parameterLocator"
deleteVariableClause = "variableLocator"
deleteTaskClause = TASK "QUOTED_STRING"
deleteTransitionClause = TRANSITION "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
parameterBindingClause = UNBIND | BIND TO ( "parameterLocator" |

```

```
    "variableLocator" )
collectPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
parameterLocator = PARAMETER "QUOTED_STRING" [ OF TASK "QUOTED_STRING" ]
variableLocator = VARIABLE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterExpertCommand

Alter the definition of an expert. The following lists all the default expert parameters:

1) EXP_LAUNCH_CONTEXT, type: STRING, direction: IN

The console context in which this expert is being launched.

2) EXP_LAUNCH_CONTEXT_TYPE, type: STRING, direction: IN

The type of the console context in which this expert is being launched.

3) EXP_OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) EXP_TYPE_TO_CREATE, type: STRING, direction: IN

The object type selected to be created when this expert is launched.

renameClause

Rename the expert.

alterPropertiesOrIconSetClause

Alter existing expert module properties and/or Icon Set.

alterExpertDetailClauses

Alter child objects of an expert.

setPropertiesClause

This clause sets properties for the corresponding object.

Basic properties for EXPERT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the expert

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the expert

Basic properties for TASK:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the task

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the task. This is equivalent to the Goal of task in the expert editor.

Name: INSTRUCTION

Type: STRING(4000)

Valid Values: N/A

Default: "

The instruction for the task

Name: PREPROCESSING

Type: STRING

Valid Values: N/A

Default: N/A

The pre-processing script for the task

Name: MAIN

Type: STRING

Valid Values: N/A

Default: N/A

The main script for the task

Name: POSTPROCESSING

Type: STRING

Valid Values: N/A

Default: N/A

The post-processing script for the task

Basic properties for START TASK:

Name: PROC_DECL

Type: STRING(4000)

Valid Values: N/A

Default: "

The procedure declaration for the expert.

Basic properties for TRANSITION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the transition

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the transition

Name: TRANSITION_CONDITION

Type: STRING(4000)

Valid Values: N/A

Default: "

Condition of the transition

Name: TRANSITION_ORDER

Type: NUMBER

Valid Values: N/A

Default: N/A

Order of the transition

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the parameter

Name: DATATYPE

Type: STRING

Valid Values: STRING, NUMBER, BOOLEAN, ARRAY

Default: STRING

Datatype of the parameter

Name: DIRECTION

Type: STRING

Valid Values: IN, OUT, INOUT

Default: IN

Direction of the parameter

Name: VALUE

Type: Same as datatype of the parameter

Valid Values: N/A

Default: N/A

The static value of the parameter

Basic properties for VARIABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the variable

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the variable

Name: DATATYPE

Type: STRING

Valid Values: STRING, NUMBER, BOOLEAN, ARRAY

Default: STRING

Datatype of the variable

Name: VALUE

Type: Same as datatype of the variable

Valid Values: N/A

Default: N/A

The static value of the variable

Properties for EXPERT:

Name: CLOSE_ASSISTANT_ON_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the task assistant should be automatically closed after the expert has been run.

Name: CLOSE_WINDOWS_ON_EXECUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Close all open windows when this expert is run.

Name: FINISH_DIALOG_ON_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: true

Shows the finish dialog upon completion of expert.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LOGGING

Type: BOOLEAN

Valid Values: true, false

Default: false

Log output to file when this expert is being run. A log file will be created in <shiphome>/owb/log directory whenever this expert is run.

Name: MENU_ITEM_DISPLAY_STRING

Type: STRING

Valid Values: N/A

Default: "

The display string when this expert is added as a menu item.

Name: ONLY_RUN_FROM_MENU

Type: BOOLEAN

Valid Values: true, false

Default: false

Only allow this expert to be run when it is attached to a menu item.

Name: REVERT_TO_SAVED_ON_ERROR

Type: BOOLEAN

Valid Values: true, false

Default: false

Revert to saved metadata if error occurs when the expert is run.

Name: RUN_STANDALONE

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the expert should be run as a standalone in expert assistant mode or not.

Name: SAVE_ALL_BEFORE_START

Type: BOOLEAN

Valid Values: true, false

Default: false

Save all metadata before running the expert.

Name: SHOW_BUSY_DIALOG

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether busy dialog should be shown when OMB or Java tasks are executed in non-standalone mode.

Name: SHOW_LOG_WINDOW

Type: BOOLEAN

Valid Values: true, false

Default: false

Sets whether the log window should be shown when running the expert.

Name: SHOW_PROGRESS_GRAPH

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the progress graph dialog should be shown when running the expert.

Name: SHOW_TASK_ASSISTANT

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the task assistant should be shown when running the expert.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Set icon set for the expert module.

addParameterClause

Add a parameter to the expert or one of it's tasks.

addVariableClause

Add a variable to the expert.

addNestedExpertClause

Add a nested expert to the expert. The nested expert is added by reference.

addTaskClause

Add a task to the expert. Valid task types include: ADVANCED_QUEUE, ALTERNATIVE_SORT_ORDER, ANALYZE_IMPACT, ANALYZE_LINEAGE, BUSINESS_AREA,

CHANGE_MANAGER, COMMIT, CONTROLCENTERJOBS, CUBE, CUSTOM_DIALOG,
DATA_AUDITOR, DATA_PROFILE, DATA_RULE, DATA_VIEWER, DEPLOY,
DERIVATION,
DIMENSION, DRILL_PATH, DRILL_TO_DETAIL, END, EXTERNAL_TABLE, FLAT_FILE,
FUNCTION, GENERATION, ITEM_FOLDER, JAVA, LIST_OF_VALUES, MAPPING,
MATERIALIZED_VIEW, NESTED_EXPERT, OBJECT_SELECTOR, OMB,
PLUGGABLE_MAPPING,
PRESENTATION_TEMPLATE, PROCEDURE, PROCESS_FLOW, REGISTERED_FUNCTION,
SELECT_SOURCE, SELECT_TARGET, SEQUENCE, SOURCE_IMPORT, START,
STARTJOB,
TABLE, VALIDATION, VIEW.

Definition of each task is as follow:

Task type : ADVANCED_QUEUE

Group : MML

Description: A task to create or alter an advanced queue.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : ALTERNATIVE_SORT_ORDER

Group : MML

Description: A task to create or alter an alternative sort order.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : ANALYZE_IMPACT

Group : Service

Description: A task to analyze impact of an object.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object for analyzing impact.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object whose impact will be analyzed.

Task type : ANALYZE_LINEAGE

Group : Service

Description: A task to analyze lineage of an object.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object for analyzing lineage.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object whose lineage will be analyzed.

Task type : BUSINESS_AREA

Group : MML

Description: A task to create or alter a business area.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : CHANGE_MANAGER

Group : Service

Description: A task to invoke the change manager.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : COMMIT

Group : Service

Description: A task to perform commit in the design repository.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : CONTROLCENTERJOBS

Group : Service

Description: A task to launch the Control Center Job Monitor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) CONTROL_CENTER_NAME, type: STRING, direction: IN

The Control Center to use.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : CUBE

Group : MML

Description: A task to create or alter a cube.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : CUSTOM_DIALOG

Group : UI

Description: A task to show a custom dialog for user interaction.

Built-in parameters:

1) GUI_RETURN_VALUE, type: ARRAY, direction: OUT

The return value of type ARRAY that stores the name and return value pair for each UI component.

2) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

3) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : DATA_AUDITOR

Group : MML

Description: A task to create or alter a data auditor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DATA_PROFILE

Group : MML

Description: A task to create or alter a data profile.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DATA_RULE

Group : MML

Description: A task to create or alter a data rule.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DATA_VIEWER

Group : Service

Description: A task to launch the data viewer.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of the OWB FCO, such as TABLE.

4) OBJECT_NAME, type: STRING, direction: IN

The name of the OWB FCO.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : DEPLOY

Group : Service

Description: A task to launch the Control Center deployment monitor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) ACTION_PLAN_NAME, type: STRING, direction: IN

The deployment action plan to use.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : DERIVATION

Group : Service

Description: A task to run derivation service.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to derive.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object to be derived.

Task type : DIMENSION

Group : MML

Description: A task to create or alter a dimension.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DRILL_PATH

Group : MML

Description: A task to create or alter a drill path.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DRILL_TO_DETAIL

Group : MML

Description: A task to create or alter a drill to detail.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : END

Group : FlowControl

Description: A task that marks the end of the flow.

Built-in parameters:

Task type : EXTERNAL_TABLE

Group : MML

Description: A task to create or alter an external table.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : FLAT_FILE

Group : MML

Description: A task to create or alter a file.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : FUNCTION

Group : MML

Description: A task to create or alter a function.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : GENERATION

Group : Service

Description: A task to invoke the generation dialog.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to generate.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object to be generated.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : ITEM_FOLDER

Group : MML

Description: A task to create or alter an item folder.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : JAVA

Group : Integration

Description: A task to execute a Java program.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) CLASS_URL, type: STRING, direction: IN

The URL specification of the Jar file. An example for a local jar file xyz.jar can be specified as file:/xyz.jar

4) CLASS_NAME, type: STRING, direction: IN

The class to load. This includes the package as well in dotted notation. For example, oracle.owb.Test

5) METHOD_NAME, type: STRING, direction: IN

The static method to execute.

6) ARGUMENT_LIST, type: STRING, direction: IN

The argument list to be passed into the static method.

Task type : LIST_OF_VALUES

Group : MML

Description: A task to create or alter a list of values.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : MAPPING

Group : MML

Description: A task to create or alter a mapping.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : MATERIALIZED_VIEW

Group : MML

Description: A task to create or alter a materialized view.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : NESTED_EXPERT

Group : FlowControl

Description: A task that references another expert.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : OBJECT_SELECTOR

Group : UI

Description: A task that shows a dialog for user to select an object.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) OBJECT_TYPE, type: STRING, direction: IN

The type of object for selection.

Task type : OMB

Group : Integration

Description: A task to launch a generic OMB script.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : PLUGGABLE_MAPPING

Group : MML

Description: A task to create or alter a pluggable mapping.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : PRESENTATION_TEMPLATE

Group : MML

Description: A task to create or alter a presentation template.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : PROCEDURE

Group : MML

Description: A task to create or alter a procedure.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : PROCESS_FLOW

Group : MML

Description: A task to create or alter a process flow.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : REGISTERED_FUNCTION

Group : MML

Description: A task to create or alter a registered function.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : SELECT_SOURCE

Group : UI

Description: A task that allows users to pick a metadata source.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) MODULE_NAME, type: STRING, direction: OUT

The name of the module.

5) LOCATION_NAME, type: STRING, direction: OUT

The name of the location.

6) OBJECT_TYPE, type: STRING, direction: OUT

The type of object selected as the source.

7) SOURCE_TYPE, type: STRING, direction: INOUT

The type of the source, for example file or database.

Task type : SELECT_TARGET

Group : UI

Description: A task that allows users to pick a target.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) MODULE_NAME, type: STRING, direction: OUT

The name of the module.

5) LOCATION_NAME, type: STRING, direction: OUT

The name of the location.

6) OBJECT_TYPE, type: STRING, direction: OUT

The type of object selected as the target.

7) SOURCE_TYPE, type: STRING, direction: INOUT

The type of the source, for example file or database.

Task type : SEQUENCE

Group : MML

Description: A task to create or alter a sequence.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : SOURCE_IMPORT

Group : Service

Description: A task to invoke the import wizard.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) OBJECT_TYPE, type: STRING, direction: IN

The type of object to import.

5) IMPORT_MODE, type: STRING, direction: IN

The mode for import. Select FULL_MODE for importing multiple objects, or MINIMAL_MODE for single import.

Task type : START

Group : FlowControl

Description: A task that marks the beginning of the flow.

Built-in parameters:

Task type : STARTJOB

Group : Service

Description: A task to launch the start job monitor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to be started.

4) OBJECT_PATH, type: STRING, direction: IN

The context path of the object to be started.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : TABLE

Group : MML

Description: A task to create or alter a table.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : VALIDATION

Group : Service

Description: A task to invoke the validation dialog.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to validate.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object to be validated.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : VIEW

Group : MML

Description: A task to create or alter a view.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is

specified as EDIT.

addTransitionClause

Add a transition to the expert.

modifyParameterClause

Modify the definition of a parameter.

modifyVariableClause

Modify the definition of a variable.

modifyTaskClause

Modify the definition of a task.

modifyTransitionClause

Modify the definition of a transition.

deleteParameterClause

Delete a parameter.

deleteVariableClause

Delete a variable.

deleteTaskClause

Delete a task expert.

deleteTransitionClause

Delete a transition.

propertyNameList

The list of property names.

propertyValueList

The list of property values being set.

parameterBindingClause

Bind or unbind two parameters. Note that this is setting the binding attribute of the parameter object and is not necessarily the same as the direction of data flow. The parameter to be set should be the one whose binding attribute is modified by the Object Inspector in the expert editor.

collectPropertiesClause

This clause sets properties for the corresponding object.

parameterLocator

Specify a parameter, either one of the expert or of a task.

variableLocator

Specify a variable in the expert.

propertyValue

Value of a property.

Examples

This command will alter a simple expert by adding a table task between tasks 'T1' and 'T2':

```
OMBALTER EXPERT 'EXP1' \  
ADD TABLE TASK 'MY_TABLE_TASK' \  
DELETE TRANSITION 'X3' \  
ADD TRANSITION 'X1' FROM TASK 'T1' TO 'MY_TABLE_TASK' \  
ADD TRANSITION 'X2' FROM TASK 'MY_TABLE_TASK' TO 'T2'
```

See Also

OMBALTER, OMBCREATE EXPERT, OMBRETRIEVE EXPERT, OMBDROP EXPERT

OMBALTER EXPERT_MODULE

Purpose

To alter an expert module.

Prerequisites

In the context of a project.

Syntax

```
alterExpertModuleCommand = OMBALTER ( EXPERT_MODULE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterExpertModuleCommand

Alter the definition of an expert module.

renameClause

Rename the expert module.

alterPropertiesOrIconSetClause

Alter existing expert module properties and/or Icon Set.

setPropertiesClause

Set properties for the expert module.

Basic properties for EXPERT_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the expert module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the expert module

propertyNameList

The list of properties to set.

propertyValueList

The list of property values to set.

propertyValue

The value of the property.

Examples

This command will alter an expert module with name 'EM1' and set its description property:

```
OMBALTER EXPERT_MODULE 'EM1' SET PROPERTIES (DESCRIPTION) VALUES  
(New  
Desc')
```

See Also

OMBALTER, OMBCREATE EXPERT_MODULE, OMBDROP EXPERT_MODULE

OMBALTER EXTERNAL_TABLE

Purpose

Alter the external table by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```

alterExternalTableCommand = OMBALTER ( EXTERNAL_TABLE "QUOTED_STRING" (
    "renameClause" [ "alterExternalTablePropertiesAndReferencesClause" ] [
    "alterExternalTableObjectClauses" ] |
    "alterExternalTablePropertiesAndReferencesClause" [
    "alterExternalTableObjectClauses" ] |
    "alterExternalTableObjectClauses" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterExternalTablePropertiesAndReferencesClause = SET (
    "setPropertiesClause" [ SET ( REF | REFERENCE )
    "alterReferencesToRecordAndLocationClauses" | UNSET ( REF | REFERENCE
    ) "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "alterReferencesToRecordAndLocationClauses" ) | UNSET ( REF |
    REFERENCE ) "unsetReferenceIconSetClause"
alterExternalTableObjectClauses = ADD ( "addExternalTableObjectClause" [
    "alterExternalTableObjectClauses" ] | "addDataRuleUsageClause" {
    "alterDataRuleUsageClauses" } ) | MODIFY (
    "modifyExternalTableObjectClause" [ "alterExternalTableObjectClauses"
    ] | "addDataRuleUsageClause" { "alterDataRuleUsageClauses" } ) |
    DELETE ( "deleteExternalTableObjectClause" [
    "alterExternalTableObjectClauses" ] | "deleteDataRuleUsageClause" {
    "alterDataRuleUsageClauses" } )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
alterReferencesToRecordAndLocationClauses =
    "setReferencesToRecordFileModuleClause" [
    "setReferencesToLocationClause" ] [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | "setReferencesToFileAndModuleClause"
    [ "setReferencesToLocationClause" ] [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | "setReferencesToLocationClause" [
    SET ( REF | REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF |
    REFERENCE ) "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause"
unsetReferenceIconSetClause = ICONSET
addExternalTableObjectClause = COLUMN "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesAndReferencesToFieldClauses" ]
    | "addExternalTableDatafileClause"
addDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" SET REF DATA_RULE
    "QUOTED_STRING" ( GROUP "QUOTED_STRING" SET REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) "QUOTED_STRING" ( ATTRIBUTE
    "QUOTED_STRING" SET REF COLUMN "QUOTED_STRING" )+ )+ [ SET
    "setPropertiesClause" ]
alterDataRuleUsageClauses = ADD "addDataRuleUsageClause" | MODIFY
    "modifyDataRuleUsageClause" | DELETE "deleteDataRuleUsageClause"
modifyExternalTableObjectClause = COLUMN "QUOTED_STRING" ( "renameClause"
    [ "moveExternalTableColumnToClause" ] [ SET

```

```

    "setPropertiesAndReferencesToFieldClauses" ] |
    "moveExternalTableColumnToClause" [ SET
    "setPropertiesAndReferencesToFieldClauses" ] | SET
    "setPropertiesAndReferencesToFieldClauses" ) | DATA_FILE
    "QUOTED_STRING" ( "renameClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" )
deleteExternalTableObjectClause = COLUMN "QUOTED_STRING" | DATA_FILE
    "QUOTED_STRING"
deleteDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferencesToRecordFileModuleClause = RECORD "QUOTED_STRING" OF
    FLAT_FILE "QUOTED_STRING"
setReferencesToLocationClause = DEFAULT_LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
setReferencesToFileAndModuleClause = FLAT_FILE "QUOTED_STRING"
setPropertiesAndReferencesToFieldClauses = "setPropertiesClause" [ SET (
    REF | REFERENCE ) "setReferencesToFieldClause" ] | ( REF | REFERENCE )
    "setReferencesToFieldClause"
addExternalTableDatafileClause = DATA_FILE "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
modifyDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" )
moveExternalTableColumnToClause = MOVE TO POSITION "INTEGER_LITERAL"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferencesToFieldClause = FIELD "QUOTED_STRING"

```

Keywords And Parameters

`alterExternalTableCommand`

Alter the existing external table.

`QUOTED_STRING`

The name of the external table to alter.

`renameClause`

Rename the external table to the value of the following quoted string.

`alterExternalTableObjectClauses`

Add, modify, or delete columns or data files.

`setPropertiesClause`

Set specified properties of the external table.

`addExternalTableObjectClause`

Add a column to the external table. The name of the new column will be the quoted string.

addDataRuleUsageClause

Add a data rule usage to the relation.

alterDataRuleUsageClauses

Add, modify, or delete data rule usages.

modifyExternalTableObjectClause

Modify the properties of a column or data file or move a column to a new position.

deleteExternalTableObjectClause

Delete a column or data file.

deleteDataRuleUsageClause

Delete a data rule usage.

propertyNameList

The names of the properties whose values you want to set.

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for EXTERNAL_TABLE:

Name: BAD_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the bad file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: BAD_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the bad file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

When deployable is set to true, a script to create an External Table is generated.

Name: DISCARD_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the discard file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: DISCARD_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the discard file. When the Access Parameters

property is specified for the External Table, this configuration parameter is ignored.

Name: ENDIAN

Type: STRING

Valid Values: BIG, LITTLE, PLATFORM

Default: PLATFORM

Data endian should be platform default, little or big. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LOAD_NULLS_WHEN_MISSING_VALUES

Type: BOOLEAN

Valid Values: true, false

Default: false

If TRUE, then NULLs are loaded for any missing values in the record. If FALSE, then records with missing values are rejected. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: LOG_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the log file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: LOG_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the log file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: NLS_CHARACTERSET

Type: STRING

Valid Values: N/A

Default: "

NLS Characterset of the file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: NUMBER_OF_REJECTS_ALLOWED

Type: NUMBER

Valid Values: 0 - 2147483647

Default: 0

The number of rejects allowed before processing is terminated.

Name: PARALLEL_ACCESS_DRIVERS

Type: NUMBER

Valid Values: 1 - 63999

Default: 1

The number of parallel access drivers to enable.

Name: PARALLEL_ACCESS_MODE

Type: BOOLEAN

Valid Values: true, false

Default: false

Enable or disable parallel processing.

Name: REJECTS_ARE_UNLIMITED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable or disable limiting the number of rejected records.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: STRING_SIZES_IN

Type: STRING

Valid Values: BYTES, CHARACTERS

Default: BYTES

String sizes are in bytes or characters. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: TRIM

Type: STRING

Valid Values: BOTH, LEFT, NONE, RIGHT, SQL*LOADER

Default: NONE

Specification from trim option on input fields. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Properties for DATA_FILE:

Name: DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

The location of this data file for the external table.

Name: DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of this data file.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

propertyValueList

The values for the named properties.

setReferencesToRecordFileModuleClause

Specify the record and full path to the flat file for the external table to reference.

setReferencesToLocationClause

Specify the default location for the external table.

setReferencesToFileAndModuleClause

Specify the full path to the flat file for the external table to reference.

setPropertyAndReferencesToFieldClauses

Set the properties and/or field reference of the external table column.

addExternalTableDatafileClause

Add a new data file to the external table. The name of the new data file will be the quoted string. You may also set the properties of the new data file.

modifyDataRuleUsageClause

Rename or modify the properties of a data rule usage.

moveExternalTableColumnToClause

Move a column of the external table.

propertyValue

A property value.

setReferencesToFieldClause

Set the name of the field which the external table column references.

Examples

```
OMBALTER EXTERNAL_TABLE 'SRC_TABLE' RENAME TO 'MY_TABLE' SET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('a new description', 'My Table')
```

This will rename the external table "SRC_TABLE" to "MY_TABLE", and set its description to "a new description", and set its business name to "My Table".

See Also

OMBALTER, OMBCREATE EXTERNAL_TABLE, OMBDROP EXTERNAL_TABLE,
OMBRETRIEVE EXTERNAL_TABLE

OMBALTER FLAT_FILE to OMBALTER STREAMS_QUEUE

This chapter lists commands associated with OMBALTER in alphabetical order, concluding with the command OMBALTER STREAMS_QUEUE. Subsequent commands associated with OMBALTER are contained in the next chapter.

OMBALTER FLAT_FILE

Purpose

Alter the flat file by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of a flat file module.

Syntax

```

alterFlatFileCommand = OMBALTER ( FLAT_FILE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] {
        "alterRecordClauses" } | "alterPropertiesOrIconSetClause" {
            "alterRecordClauses" } | "alterRecordClauses" { "alterRecordClauses" }
    ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterRecordClauses = ADD "addRecordClauseForAlter" | MODIFY
    "modifyRecordClause" | DELETE "deleteRecordClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addRecordClauseForAlter = RECORD "QUOTED_STRING" [ SET
    "setPropertiesClause" ] { ADD "addFieldClauseForAlter" }
modifyRecordClause = RECORD "QUOTED_STRING" ( "renameClause" [ SET
    "setPropertiesClause" ] { "alterFieldClauses" } | SET
    "setPropertiesClause" { "alterFieldClauses" } | "alterFieldClauses" {
        "alterFieldClauses" } )
deleteRecordClause = RECORD "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addFieldClauseForAlter = FIELD "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ]
alterFieldClauses = ADD "addFieldClauseForAlter" | MODIFY
    "modifyFieldClause" | DELETE "deleteFieldClause"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
modifyFieldClause = FIELD "QUOTED_STRING" ( "renameClause" [
    "moveFieldToClause" ] [ SET "setPropertiesClause" ] |
    "moveFieldToClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" )
deleteFieldClause = FIELD "QUOTED_STRING"
moveFieldToClause = MOVE TO POSITION "INTEGER_LITERAL"

```

Keywords And Parameters

alterFlatFileCommand

Alter a flat file.

QUOTED_STRING

The name of the flat file to alter.

renameClause

Rename the flat file to the following quoted string.

alterRecordClauses

Add, modify, or drop a record of the flat file.

setPropertyClause

Set the properties of the flat file, record, or field.

addRecordClauseForAlter

Add a record named by the following quoted string.

modifyRecordClause

Modify a record specified by the following quoted string.

deleteRecordClause

Delete a record specified by the following quoted string.

propertyNameList

The names of the properties whose values you want to set.

Properties for FLAT_FILE:

Name: DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default "

The name of the "sampled" file. Also the default data file value used in SQL*Loader maps and External Tables.

Name: IS_DELIMITED

Type: BOOLEAN

Valid Values: true, false, 1, 0

Default: true

True indicates that this flat file is delimited. False indicates that its fields are defined by fixed lengths

Name: CHARACTERSET

Type: STRING

Valid Values:

AL24UTFFSS,AR8ARABICMAC,AR8ARABICMACS,AR8ISO8859P6,AR8MSAWIN,AR8MSWIN1256,BLT8CP921,BLT8EBCDIC1112,BLT8MSWIN1257,BLT8PC775,CDN8PC863,CL8EBCDIC1025,CL8EBCDIC1025X,CL8ISO8859P5,CL8KOI8R,CL8MACCYRILLIC,CL8MACCYRILLICS,CL8MSWIN1251,D8EBCDIC273,DK8EBCDIC277,EE8EBCDIC870,EE8ISO8859P2,EE8MACCE,EE8MACCES,EE8MACCROATIAN,EE8MACCROATIANS,EE8MSWIN1250,EE8PC852,EL8EBCDIC875,EL8ISO8859P7,EL8MACGREEK,EL8MACGREEKS,EL8MSWIN1253,EL8PC437S,EL8PC737,EL8PC869,F8EBCDIC297,I8EBCDIC280,IS8MACICELANDIC,IS8MACICELANDICS,IS8PC861,IW8EBCDIC424,IW8ISO8859P8,IW8MACHEBREW,IW8MACHEBREWS,IW8MSWIN1255,JA16EBCDIC930,JA16EUC,JA16EUCYEN,JA16MACSJIS,JA16SJIS,JA16SJSYEN,JA16VMS,KO16KSC5601,LT8MSWIN921,N8PC865,NEE8ISO8859P4,RU8PC855,RU8PC866,S8EBCDIC278,SE8ISO8859P3,TH8MACTHAI,TH8MACTHAIS,TH8TISASCII,TR8EBCDIC1026,TR8MACTURKISH,TR8MACTURKISHS,TR8MSWIN1254,TR8PC857,US7ASCII,US8PC437,UTF8,WE8EBCDIC284,WE8EBCDIC285,WE8EBCDIC37,WE8EBCDIC37C,WE8EBCDIC500,WE8EBCDIC500C,WE8EBCDIC871,WE8ISO8859P1,WE8ISO8859P9,WE8MACROMAN8,WE8MACROMAN8S,WE8MSWIN1252,WE8PC850,WE8PC860,ZHS16CGB231280,ZHS16GBK,ZHS16MACCGB231280,ZHT16BIG5,ZHT16MSWIN950,ZHT32EUC

Default: WE8MSWIN1252

The character set of the data file.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

The character(s) which denote the end of a physical record in a data file.

A hex value may be entered by entering embedded single quotes twice as:

'x"0f"' (all are single quotes). The outside single quote indicates a quoted string and the inside single quotes single-quote x single-quote single-quote 0F single-quote single-quote single-quote. (Please note that this is not the FIELD_DELIMITER.)

Name: RECORD_LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0 (Records are delimited by default)

The length (in characters) of the records in the data file.

Name: RECORD_TYPE_COLUMN_NUMBER

Type: NUMBER

Valid Values: 0+

Default: 0

The column which contains the record type values for a delimited, multi-record type file.

Name: RECORD_TYPE_START_POSITION

Type: NUMBER

Valid Values: 0+

Default: 0

The starting position of the field (relative to 0) which contains the record type values for a fixed-length, multi-record type file.

Name: RECORD_TYPE_END_POSITION

Type: NUMBER

Valid Values: 0+

Default: 0

The ending position of the field (relative to 0) which contains the record type values for a fixed-length, multi-record type file.

Name: NUMBER_OF_RECORDS_TO_SKIP

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The default number of records to skip when loading this file.

Name: FIELD_DELIMITER

Type: STRING

Valid Values: Any single character

Default: ',' (Comma)

The character to divide the fields in a delimited file.

Name: FIELD_LEFT_ENCLOSURE

Type: STRING

Valid Values: Any single character

Default: None

A character to enclose fields which may contain the field delimiter.

Name: FIELD_RIGHT_ENCLOSURE

Type: STRING

Valid Values: Any single character

Default: None

A character to enclose fields which may contain the field delimiter.

Name: NUMBER_OF_PHYSICAL_RECORDS_PER_LOGICAL

Type: Number

Valid Values: 0+

Default: 0

Set this value if you wish to concatenate a fixed number of physical records to form a single logical record.

Name: CONTINUE_IF_ENDS_WITH

Type: STRING

Valid Values: Any single character

Default: None

Set this value if you wish to concatenate a variable number of physical records to form a single logical record, determined by records ending with this character.

Name: CONTINUE_IF_STARTS_WITH

Type: STRING

Valid Values: N/A

Default: None

Set this value if you wish to concatenate a variable number of physical records to form a single logical record, determined by records beginning with this character.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the record

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the record

Properties for RECORD:

Name: RECORD_TYPE_VALUE
Type: STRING
Valid Values: N/A
Default: None

This is a mandatory property for each record of a multi-record type file.
It is the string which will identify this record type in the data file.

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the record

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the record

Properties for FIELD:

Name: DATATYPE
Type: STRING
Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE,
FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER,
INTEGER
EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL
YEAR TO MONTH,
LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP,
TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC,
VARRAW,

VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

This is the SQL*Loader data type for the field.

Name: MAXIMUM_LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0

This is the maximum length of the field.

Name: LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0

Deprecated. This is the length of the field in a fixed length file. This is the max length of the field in a delimited file.

Name: PRECISION

Type: NUMBER

Valid Values: Depends on data type

Default: 0

Precision of the field.

Name SCALE

Type: NUMBER

Valid Values: Depends on data type

Default: 0

Scale of the field

Name: START_POSITION

Type: NUMBER

Valid Values: 1+

Default: 1

The starting position of a field for a fixed length file.

Name: END_POSITION

Type: NUMBER

Valid Values: 1+

Default: 1

The ending position of a field for a fixed length file.

Name: SQL_DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: DEFAULT. This will derive the SQL_DATATYPE from the value of DATATYPE.

The data type which the field will be treated as in mapping and for External Tables.

Name: SQL_LENGTH

Type: NUMBER

Valid Values: 1 - 4000

Default: 0

Name: SQL_PRECISION

Type: NUMBER

Valid Values: 1 - 38

Default: 1

Name: SQL_SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 0

Name: MASK

Type: STRING

Valid Values: N/A

Default: None

This is the mask used to define the format of DATE fields in the data file.

Name: NULL_IF

Type: STRING

Valid Values: Either = or != followed by either 'BLANKS', a single quoted string, or a hexadecimal string

Default: None

If this condition is true for a field, the value loaded will be NULL.

Name: DEFAULT_IF

Type: STRING

Valid Values: Either = or != followed by either 'BLANKS', a single quoted string, or a hexadecimal string

Default: None

If this condition is true for a field, the value loaded will be either NULL or 0, dependent on data type.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the field

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the field

propertyValueList

The values for the named properties.

addFieldClauseForAlter

Add a field to the record.

`alterFieldClauses`

Add, modify, or drop a field.

`propertyValue`

A property value.

`modifyFieldClause`

Modify the properties of a field specified by the following quoted string.

`deleteFieldClause`

Delete a field specified by the following quoted string.

`moveFieldToClause`

Move a field to a new position in the record.

Examples

```
OMBALTER FLAT_FILE 'OLD_NAME' RENAME TO 'NEW_NAME' SET
PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a target file.', 'target
file')
```

This will rename the flat file "OLD_NAME" to "NEW_NAME", set its description to "This becomes a target file", and set its business name to "target file".

See Also

OMBALTER, OMBCREATE FLAT_FILE, OMBDROP FLAT_FILE

OMBALTER FLAT_FILE_MODULE

Purpose

Alter the flat file module by renaming it, and/or resetting its properties.

Prerequisites

Should be in the context of a project.

Syntax

```
alterFlatFileModuleCommand = OMBALTER ( FLAT_FILE_MODULE "QUOTED_STRING" (
    "renameClause" [
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ] |
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataMetadataModule = ( ( SET ( (
    "alterPropertiesClause" [ ( SET
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) ) | (
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) ) ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) [
        "addOrRemoveOrModifyModuleReferenceLocationClause" ]
alterPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
unsetReferenceClauseForDataMetadataModule = (
    "unsetReferenceLocationClause" [ UNSET
    "unsetReferenceMetadataLocationOrIconSetClause" ] |
    "unsetReferenceMetadataLocationOrIconSetClause" )
addOrRemoveOrModifyModuleReferenceLocationClause = (
    "addReferenceLocationClause" | "removeReferenceLocationClause" |
    "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
    "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceMetadataLocationOrIconSetClause = (
    "unsetReferenceMetadataLocationClause" [ UNSET
    "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" SET AS DEFAULT
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
```

```

    "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceMetadataLocationClause = ( REFERENCE | REF )
    METADATA_LOCATION "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET

```

Keywords And Parameters

alterFlatFileModuleCommand

Alter a flat file module.

QUOTED_STRING

The name of the flat file module to alter.

renameClause

Rename the flat file module.

QUOTED_STRING

The new name for the flat file module.

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the flat file module.

unsetReferenceClauseForDataMetadataModule

Unset location and/or icon set for the flat file module.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the flat file module.

propertyNameList

The names of the properties whose values you want to set.

Basic properties for FLAT_FILE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of the flat file module.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the flat file module.

propertyValueList

The values for the named properties.

setReferenceLocationClause

Set a location to the existing flat file module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the flat file module.

unsetReferenceLocationClause

Unset a location to the existing flat file module.

unsetReferenceMetadataLocationOrIconSetClause

Unset metadata location and/or icon set for the flat file module.

addReferenceLocationClause

Add a runtime location to the flat file module.

removeReferenceLocationClause

Remove a runtime location from the flat file module.

modifyReferenceLocationClause

Modify a runtime location of the flat file module.

propertyValue

A property value.

setReferenceMetadataLocationClause

Set metadata location for the flat file module.

`setReferenceIconSetClause`

Set icon set for the flat file module.

`unsetReferenceMetadataLocationClause`

Unset metadata location for the flat file module.

`unsetReferenceIconSetClause`

Unset icon set for the flat file module.

Examples

```
OMBALTER FLAT_FILE_MODULE 'src_module' RENAME TO 'tgt_module' SET
PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a target
module.', 'target module')
```

This will rename the flat file module "src_module" to "tgt_module", and set its description to "This becomes a target module", set its business name to "target module".

See Also

OMBALTER, OMBCREATE FLAT_FILE_MODULE, OMBDROP FLAT_FILE_MODULE

OMBALTER FUNCTION

Purpose

Alter the Function by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
alterFunctionCommand = OMBALTER ( FUNCTION "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] [
    "alterFuncProcParameterSCOClause" ] | "alterPropertiesOrIconSetClause"
    [ "alterFuncProcParameterSCOClause" ] |
    "alterFuncProcParameterSCOClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterFuncProcParameterSCOClause = ( ADD ( "alterFuncProcParameterClause" |
    "addRelationalDependentClause" ) | MODIFY
    "modifyFuncProcParameterClause" | DELETE (
    "deleteFuncProcParameterClause" | "deleteRelationalDependentClause" )
    ) [ "alterFuncProcParameterSCOClause" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterFuncProcParameterClause = PARAMETER "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ]
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
modifyFuncProcParameterClause = ( PARAMETER "QUOTED_STRING" (
    "renameClause" | "moveToClause" | [ SET "setPropertiesClause" ] ) )
deleteFuncProcParameterClause = ( PARAMETER "QUOTED_STRING" )
deleteRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterFunctionCommand

This command modifies an existing Function.

QUOTED_STRING

Name of the existing Function in single quotes.

renameClause

Rename a Function.

alterFuncProcParameterSCOClause

Modify, delete or add a Parameter for Function/Procedure, or add or delete dependencies to some other relational objects.

setPropertiesClause

Used to set properties (core, user-defined) for function. Valid properties are as shown:

Basic properties for FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Function

Name: RETURN_TYPE

Type: STRING

Valid Values: PLS_INTEGER, BINARY_INTEGER, BOOLEAN, NUMBER, FLOAT, CHAR,

VARCHAR, VARCHAR2, DATE

Default: NUMBER

Set the Return Type for Function

Name: IMPLEMENTATION

Type: STRING

Valid Values: N/A

Default: "

Set the code for Function which is included global variable declaration and code between BEGIN and END.

Name: IS_DETERMINISTIC

Type: BOOLEAN

Valid Values: true, false

Default: false

This setting helps the optimizer avoid redundant function calls.

Name: IS_PARALLEL_ENABLE

Type: BOOLEAN

Valid Values: true, false

Default: false

This option sets flag to a stored function can be used safely in the slave sessions of parallel DML evaluations.

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB, BOOLEAN,
CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO
MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, TIMESTAMP,
TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARHRCAR,
VARCHAR2, XMLTYPE
Default: NUMBER
Set the data type for Parameter

Name: IN_OUT
Type: STRING
Valid Values: IN, OUT, INOUT
Default: 'IN'
Set the parameter mode for Parameter

Name: DEFAULT_VALUE
Type: STRING
Valid Values: N/A
Default: "
Set the default value for Parameter

Properties for FUNCTION:

Name: AUTHID
Type: STRING
Valid Values: Current_User, Definer, None
Default: None
Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

alterFuncProcParameterClause

This clause alters Parameter of a Function.

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

modifyFuncProcParameterClause

Modify one or more Parameters to this Function/Procedure.

deleteFuncProcParameterClause

Delete one or more Parameters to this Function/Procedure.

deleteRelationalDependentClause

This clause deletes referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

moveToClause

Move a Parameters of this Function/Procedure.

propertyValue
Value of a property.

Examples

```
OMBALTER FUNCTION 'func' RENAME TO 'function_1' SET PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a function_1',
'function_1')
```

This will rename the Function "func" to "function_1", and set its description to "This becomes a function_1", set its business name to "function_1"

If Packaged Function is overloaded, first find the Signature by using OMBLIST command, and then use OMBALTER command using appropriate signature.

Example, if OMBLIST FUNCTIONS gives following two signatures,

```
FUNC_1 (NUMBER) RETURN NUMBER
```

```
FUNC_1 (VARCHAR2, NUMBER) RETURN NUMBER
```

The OMBALTER Syntax to modify the first one will be as follows

```
OMBALTER FUNCTION 'FUNC_1 \ (NUMBER\ ) RETURN NUMBER' SET
PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('descri_FUNC_1', 'FUNC_1')
```

See Also

OMBALTER, OMBCREATE FUNCTION, OMBDROP FUNCTION

OMBALTER GATEWAY_MODULE

Purpose

Alter the Gateway module by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of project.

Syntax

```
alterGatewayModuleCommand = OMBALTER ( GATEWAY_MODULE "QUOTED_STRING" (
    "renameClause" [
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ] |
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataMetadataModule = ( ( SET ( (
    "alterPropertiesClause" [ ( SET
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) ) | (
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) ) ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) [
        "addOrRemoveOrModifyModuleReferenceLocationClause" ]
alterPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
unsetReferenceClauseForDataMetadataModule = (
    "unsetReferenceLocationClause" [ UNSET
    "unsetReferenceMetadataLocationOrIconSetClause" ] |
    "unsetReferenceMetadataLocationOrIconSetClause" )
addOrRemoveOrModifyModuleReferenceLocationClause = (
    "addReferenceLocationClause" | "removeReferenceLocationClause" |
    "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
    "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceMetadataLocationOrIconSetClause = (
    "unsetReferenceMetadataLocationClause" [ UNSET
    "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" SET AS DEFAULT
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
```



```

"FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
"QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceMetadataLocationClause = ( REFERENCE | REF )
METADATA_LOCATION "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET

```

Keywords And Parameters

alterGatewayModuleCommand

This command modifies an existing Gateway module.

QUOTED_STRING

Name of the existing Gateway module in single quotes.

renameClause

Rename the Gateway module.

alterPropertiesOrReferenceClauseForDataMetadataModule

Alter existing Gateway module's properties and/or locations and/or icon sets.

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the gateway module.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the Gateway module.

setReferenceLocationClause

Set a location to the existing Gateway module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the gateway module.

unsetReferenceLocationClause

Unset a location to the existing Gateway module.

unsetReferenceMetadataLocationOrIconSetClause

Unset metadata location and/or icon set for the gateway module.

`addReferenceLocationClause`

Add a runtime location to the gateway module.

`removeReferenceLocationClause`

Remove a runtime location from the gateway module.

`setReferenceMetadataLocationClause`

Set metadata location for the gateway module.

`setReferenceIconSetClause`

Set icon set for the gateway module.

`unsetReferenceMetadataLocationClause`

Unset metadata location for the gateway module.

`unsetReferenceIconSetClause`

Unset icon set for the gateway module.

Examples

```
OMBALTER GATEWAY_MODULE 'db2_module' RENAME TO 'db2_module_2'
```

This will rename the Gateway module "db2_module" to "db2_module_2".

See Also

OMBALTER, OMBCREATE GATEWAY_MODULE, OMBDROP GATEWAY_MODULE

OMBALTER ICONSET

Purpose

To modify one or more properties of an iconset, including renaming it.

Prerequisites

Any context.

Syntax

```
alterIconSetCommand = OMBALTER ( ICONSET "QUOTED_STRING" ( "renameClause"
    [ SET "setPropertiesClause" ] | SET "setPropertiesClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterIconSetCommand

This command alters the properties of an iconset.

QUOTED_STRING

The name of the iconset to alter.

renameClause

Renames an iconset to a different name.

QUOTED_STRING

New name to set for the iconset.

setPropertiesClause

This clause sets a list properties to the specified values.

Basic properties for ICONSET:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the iconset

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the iconset

Name: BELONGS_TO_GROUP

Type: STRING

Valid Values: N/A

Default: "

Name of the Group to which the iconset belongs

Name: CANVAS_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the canvas icon (36x36)

Name: PALETTE_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the palette icon (18x18)

Name: TREE_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the tree icon (16x16)

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

A property value.

Examples

```
OMBALTER ICONSET 'ICON1' RENAME TO 'ICON2' SET PROPERTIES ( CANVAS_
ICON )
VALUES ('new_canvas.gif')
```

See Also

OMBALTER, OMBCREATE ICONSET, OMBDROP ICONSET

OMBALTER IMPORT_ACTION_PLAN

Purpose

To modify a transient import action plan.

Prerequisites

In the context of a project.

Syntax

```
alterImportActionPlanCommand = ( OMBALTER ( IMPORT_ACTION_PLAN )
    "QUOTED_STRING" ( "alterActionPlanClause" { "alterActionPlanClause" }
    ) )
alterActionPlanClause = "renameActionPlanClause" | "addActionClause" |
    "deleteActionClause" | "modifyActionClause"
renameActionPlanClause = RENAME TO "QUOTED_STRING"
addActionClause = ADD ACTION "QUOTED_STRING" ( "setPropertiesClause" [
    "setRefSourceAndTargetClause" ] | "setRefSourceAndTargetClause" )
deleteActionClause = DELETE ACTION "QUOTED_STRING"
modifyActionClause = MODIFY ACTION "QUOTED_STRING" "modifyActionOperation"
    { "modifyActionOperation" }
setPropertiesClause = SET PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setRefSourceAndTargetClause = SET ( REF | REFERENCE ) "sourcesClause" SET
    ( REF | REFERENCE ) "targetClause"
modifyActionOperation = "renameActionClause" | "setPropertiesClause" |
    "setRefSourceClause" | "setRefTargetClause" | "deleteSourceClause" |
    "unsetTargetReferenceClause"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
sourcesClause = SOURCE "ObjType" "QUOTED_STRING" [ SET ( REF | REFERENCE )
    "sourcesClause" ]
targetClause = TARGET "ObjType" "QUOTED_STRING"
renameActionClause = RENAME TO "QUOTED_STRING"
setRefSourceClause = SET ( REF | REFERENCE ) "sourcesClause"
setRefTargetClause = SET ( REF | REFERENCE ) "targetClause"
deleteSourceClause = UNSET ( REF | REFERENCE ) SOURCE "ObjType"
    "QUOTED_STRING"
unsetTargetReferenceClause = UNSET ( REF | REFERENCE ) TARGET
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterImportActionPlanCommand

This command is for changing the details of an existing import action plan.

QUOTED_STRING

The name of the transient import action plan to be altered.

alterActionPlanClause

One modification to the import action plan.

renameActionPlanClause

For renaming the action plan.

QUOTED_STRING

The new name for the import action plan.

addActionClause

For adding an action to the import action plan.

QUOTED_STRING

The name of the action to be added.

deleteActionClause

For deleting an existing action.

QUOTED_STRING

The name of the action to be deleted.

modifyActionClause

For changing details of an existing action in the import action plan.

setPropertyClause

For setting any properties for the import action. For the current release, there are no predefined property for import actions.

setRefSourceAndTargetClause

For specifying source and target objects for the import action. The source objects are to be imported into target.

modifyActionOperation

The clause for modifying details of an action.

renameActionClause

For renaming the action.

QUOTED_STRING

The new name for the action.

`deleteSourceClause`

For deleting a source item from the list of source items of this action.

`QUOTED_STRING`

The name of the source object. Note that the name of the source object must be qualified with schema name, such as 'SCOTT.EMP'.

`unsetTargetReferenceClause`

For removing the pointer to target object from the import action.

Examples

```
OMBALTER IMPORT_ACTION_PLAN 'PLAN1'  
ADD ACTION 'A3'  
SET REF SOURCE DIMENSION 'SCOTT.PRODUCTS_DIM'  
SET REF TARGET TRANSPORTABLE_MODULE 'TM101'
```

This example will add one more action to the set of actions in import action plan PLAN1.

```
OMBALTER IMPORT_ACTION_PLAN 'PLAN1' DELETE ACTION 'A1'
```

This command will delete an action from import action plan PLAN1.

```
OMBALTER IMPORT_ACTION_PLAN 'PLAN1'  
MODIFY ACTION 'A2'  
SET REF SOURCE TABLE 'SCOTT.EMP'
```

This example will change the details of the import action A2 by adding one more item SCOTT.EMP to the source item list.

```
OMBALTER IMPORT_ACTION_PLAN 'PLAN1'  
MODIFY ACTION 'A2'  
DELETE SOURCE TABLE 'SCOTT.T1'
```

This example will delete one item from the set of items in import action A2 of existing import action plan PLAN1.

See Also

OMBCREATE IMPORT_ACTION_PLAN, OMBIMPORT

OMBALTER ITEM_FOLDER

Purpose

Alters an item folder.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```

alterItemFolderCommand = ( OMBALTER ITEM_FOLDER "QUOTED_STRING" ( (
    "renameClause" [ SET "setpropertiesClauseDelayed" ] [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterItemFolderSCOClauses" } ) ) | ( SET "setpropertiesClauseDelayed"
    [ SET "setReferenceIconSetClause" ] [ UNSET
    "unsetReferenceIconSetClause" ] { "alterItemFolderSCOClauses" } ) ) | (
    SET "setReferenceIconSetClause" [ UNSET "unsetReferenceIconSetClause"
    ] { "alterItemFolderSCOClauses" } ) ) | ( UNSET
    "unsetReferenceIconSetClause" { "alterItemFolderSCOClauses" } ) ) | (
    "alterItemFolderSCOClauses" { "alterItemFolderSCOClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setpropertiesClauseDelayed = PROPERTIES "(" "propertyNameListVector" ")"
    VALUES "(" "propertyValueListVector" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterItemFolderSCOClauses = ADD ( "addItemClauseForAlter" |
    "addConditionClauseForAlter" | "addJoinClause" ) | MODIFY (
    "modifyItemClause" | "modifyConditionClause" | "modifyJoinClause" ) |
    DELETE ( "deleteItemClause" | "deleteConditionClause" |
    "deleteJoinClause" ) | UNSET "unsetJoinUsageClause"
propertyNameListVector = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueListVector = "propertyValue" { ", " "propertyValue" }
addItemClauseForAlter = ITEM "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ] [ SET ( REF |
    REFERENCE ) ( "ItemItemReferencesClause" |
    "ItemListOfValuesReferencesClause" |
    "ItemDrillToDetailReferencesClause" |
    "ItemAlternativeSortOrderReferencesClause" |
    "ItemColumnReferencesClause" ) ]
addConditionClauseForAlter = CONDITION "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
addJoinClause = JOIN "QUOTED_STRING" [ SET "setPropertiesClause" ] [ SET (
    REF | REFERENCE ) "JoinForeignKeyReferencesClause" ] {
    "joinComponentClause" }
modifyItemClause = ITEM "QUOTED_STRING" [ "renameClause" ] [
    "moveItemToClause" ] [ SET "setPropertiesClause" ] [ UNSET ( REF |
    REFERENCE ) ( COLUMN | LIST_OF_VALUES | DRILL_TO_DETAIL |
    ALTERNATIVE_SORT_ORDER ) ] [ SET ( REF | REFERENCE ) (
    "ItemItemReferencesClause" | "ItemListOfValuesReferencesClause" |
    "ItemDrillToDetailReferencesClause" |
    "ItemAlternativeSortOrderReferencesClause" |
    "ItemColumnReferencesClause" ) ]
modifyConditionClause = CONDITION "QUOTED_STRING" ( [ "renameClause" ] [
    SET "setPropertiesClause" ] )
modifyJoinClause = JOIN "QUOTED_STRING" ( [ "renameClause" ] [ SET

```

```

    "setPropertyClause" ] [ UNSET ( REF | REFERENCE ) FOREIGN_KEY ] {
    "alterJoinComponentSCOClauses" } )
deleteItemClause = ITEM "QUOTED_STRING"
deleteConditionClause = CONDITION "QUOTED_STRING"
deleteJoinClause = JOIN "QUOTED_STRING"
unsetJoinUsageClause = ( REF | REFERENCE ) USING JOIN "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
ItemItemReferencesClause = ITEM "QUOTED_STRING" OF ITEM_FOLDER
    "QUOTED_STRING" "itemJoinUsages"
ItemListOfValuesReferencesClause = LIST_OF_VALUES "QUOTED_STRING"
ItemDrillToDetailReferencesClause = DRILL_TO_DETAIL "QUOTED_STRING"
ItemAlternativeSortOrderReferencesClause = ALTERNATIVE_SORT_ORDER
    "QUOTED_STRING"
ItemColumnReferencesClause = COLUMN "QUOTED_STRING" OF ( TABLE | (
    EXTERNAL_TABLE | VIEW ) ) "QUOTED_STRING"
JoinForeignKeyReferencesClause = FOREIGN_KEY "QUOTED_STRING" OF ( TABLE |
    VIEW ) "QUOTED_STRING"
joinComponentClause = ADD JOIN_COMPONENT "QUOTED_STRING" [ SET
    "setPropertyClause" ] { SET ( REF | REFERENCE )
    "setJoinComponentClauseDetails" }
moveItemToClause = MOVE TO POSITION "INTEGER_LITERAL"
alterJoinComponentSCOClauses = ADD "joinComponentClauseforAlter" | MODIFY
    "modifyJoinComponentClause" | DELETE "deleteJoinComponentClause"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
itemJoinUsages = { SET ( REF | REFERENCE ) USING JOIN "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING" }
setJoinComponentClauseDetails = LOCAL ITEM "QUOTED_STRING" | REMOTE ITEM
    "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
joinComponentClauseforAlter = JOIN_COMPONENT "QUOTED_STRING" [ SET
    "setPropertyClause" ] { SET ( REF | REFERENCE )
    "setJoinComponentClauseDetails" }
modifyJoinComponentClause = JOIN_COMPONENT "QUOTED_STRING" [
    "renameClause" ] [ SET "setPropertyClause" ] [ SET ( REF | REFERENCE
    ) "setJoinComponentClauseDetails" ] [ UNSET ( REF | REFERENCE )
    "unsetJoinComponentClauseDetails" ]
deleteJoinComponentClause = JOIN_COMPONENT "QUOTED_STRING"
unsetJoinComponentClauseDetails = LOCAL ITEM | REMOTE ITEM

```

Keywords And Parameters

alterItemFolderCommand

This clause alters an item folder.

QUOTED_STRING

name of the item folder.

renameClause

Renames an item folder with a different name.

setPropertyClauseDelayed

This clause sets the properties.

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterItemFolderSCOClauses

This clause modifies the contents of the item folder.

propertyNameListVector

This clause holds the names of the properties.

propertyValueListVector

This clause holds the values of the properties.

addItemClauseForAlter

This clause adds an item to an item folder.

QUOTED_STRING

name of the item.

addConditionClauseForAlter

This clause adds a condition to an item folder.

QUOTED_STRING

name of the condition.

addJoinClause

This clause adds a join to an item folder.

QUOTED_STRING

name of the join.

modifyItemClause

This clause modifies an item in an item folder.

QUOTED_STRING

name of the item.

modifyConditionClause

This clause modifies a condition in an item folder.

QUOTED_STRING

name of the condition.

modifyJoinClause

This clause modifies a join in an item folder.

QUOTED_STRING

name of the join.

deleteItemClause

This clause deletes an item from an item folder.

QUOTED_STRING

name of the item.

deleteConditionClause

This clause deletes a condition from an item folder.

QUOTED_STRING

name of the condition.

deleteJoinClause

This clause deletes a join from an item folder.

QUOTED_STRING

name of the join.

unsetJoinUsageClause

removes a join usage.

propertyValue

This is a property value.

setPropertyClause

This clause sets the properties of the object.

Basic properties for ITEM_FOLDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the item folder

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the item folder

Name: EXTERNAL_TABLE_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The physical name for the corresponding table or view. This is automatically set if the Folder is associated with a Table

Name: VISIBLE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the item folder should be visible to the user

Name: FOLDER_TYPE

Type: STRING(40)

Valid Values: SIMPLE, COMPLEX

Default: "

The type of item folder

Basic properties for ITEM:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the item

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the item

Name: ALIGNMENT

Type: STRING(40)

Valid Values: GENERAL, LEFT, CENTER, RIGHT

Default: 'GENERAL'

The default alignment for displaying the item

Name: DISPLAY_CASE

Type: STRING(40)

Valid Values: GENERAL, LOWER, UPPER, INITCAPPED

Default: 'GENERAL'

How alphabetic characters should be displayed

Name: CASE_STORAGE

Type: STRING(40)

Valid Values: GENERAL, LOWER, UPPER, MIXED

Default: 'GENERAL'

How alphabetic characters are stored

Name: CONTENT_TYPE

Type: STRING(40)

Valid Values: No Value or FILE. For datatypes such as BLOB, it may contain

a file extension such as DOC, AVI, WAV, JPG

Default: "

Details on whether the Item contains a file name or should be processed by an external application

Name: DEFAULT_AGGREGATE

Type: STRING(255)

Valid Values: Detail, AVG, COUNT, MAX, MIN, SUM

Default: 'SUM' when the datatype is Numeric, 'Detail' otherwise

Name of the default rollup function for the item

Name: DEFAULT_POSITION

Type: STRING(40)

Valid Values: MEASURE, TOP OR SIDE, TOP, SIDE, PAGE

Default: 'MEASURE' when the datatype is NUMBER or FLOAT, 'TOP OR SIDE' otherwise

Default position for the item

Name: REPLACE_NULL_WITH

Type: STRING(255)

Valid Values: N/A

Default: "

The value to be displayed for null values

Name: FORMULA

Type: STRING

Valid Values: N/A

Default: "

The text of the derivation expression for a derived item

Name: EXTERNAL_COLUMN_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The external name of the corresponding column. This is automatically set if the Item is associated with a Column

Name: FORMAT_MASK

Type: STRING(255)

Valid Values: N/A

Default: "

The display format mask for the item

Name: HEADING

Type: STRING(255)

Valid Values: N/A

Default: "

The displayed heading text for the item

Name: DATATYPE

Type: STRING(40)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.ROW_LCR, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE

TIMESTAMP WITH TIME ZONE, UNSPECIFIED, VARCHAR, VARCHAR2, XMLTYPE,

SYS.XMLFORMAT, BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS, SYS.XMLSEQUENCETYPE,

SYS_REFCURSOR, BLAST_MATCH_PLSQLRECORDTYPE

Default: 'VARCHAR2'

The datatype for the item

Name: VISIBLE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the item should be visible to the user

Name: MAX_CHAR_FETCHED

Type: Number

Valid Values: N/A

Default: "

The maximum number of characters fetched for an item

Name: DEFAULT_WIDTH

Type: Number

Valid Values: N/A

Default: "

The default number of characters to display

Name: WORD_WRAP

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether wordwrap is allowed in the display

Basic properties for JOIN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the join

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the join

Name: OUTER_JOIN_ON_MASTER

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether detail rows with no related master row should be included in the join

Name: OUTER_JOIN_ON_DETAIL

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether master rows with no related detail rows should be included in the join

Name: EXTERNAL_KEY_NAME

Type: STRING(255)

Valid Values: N/A

Default: ''

The external name of the corresponding foreign key. This is automatically set if the Join is associated with a Foreign Key

Name: DETAIL_ALWAYS_HAS_MASTER

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether every detail row must reference a unique master row

Name: ONE_TO_ONE

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether a master row only ever has a single detail row

Basic properties for JOIN_COMPONENT:

Name: JOIN_OPERATOR

Type: STRING(200)

Valid Values: =, <>, <, <=, > or >=

Default: ''

Business name of the join

Basic properties for CONDITION:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the condition

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the condition

Name: MATCH_CASE
Type: BOOLEAN
Valid Values: Y,N
Default: 'Y'
Whether the case of alphabetic characters must match exactly

Name: FORMULA
Type: STRING
Valid Values: N/A
Default: "
The expression for the condition

Name: MANDATORY
Type: BOOLEAN
Valid Values: Y,N
Default: 'N'
Whether the Condition is optional or mandatory

Properties for ITEM_FOLDER:

Name: DB_LOCATION
Type: STRING
Valid Values: N/A
Default: "
Location for the referenced database object

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Name: OPTIMIZER_HINT

Type: STRING

Valid Values: N/A

Default: "

Optimizer Hint to be added when this Item Folder is used in a query

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

ItemItemReferencesClause

This clause is used to reference another item.

QUOTED_STRING

name of the referenced item.

ItemListOfValuesReferencesClause

This clause is used to reference a list of values.

QUOTED_STRING

name of the list of values.

ItemDrillToDetailReferencesClause

This clause is used to reference a drill to detail.

QUOTED_STRING

name of the drill to detail.

ItemAlternativeSortOrderReferencesClause

This clause is used to reference an alternative sort order.

QUOTED_STRING

name of the alternative sort order.

ItemColumnReferencesClause

This clause is used to reference a column.

QUOTED_STRING

name of the referenced column.

JoinForeignKeyReferencesClause

The foreign key reference.

joinComponentClause

The join components.

moveItemToClause

This clause is for positioning an item in an item folder.

alterJoinComponentSCOClauses

This clause alters the structure of the join component.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

itemJoinUsages

The specific joins to be used.

setJoinComponentClauseDetails

The structure of the join component.

joinComponentClauseforAlter

This clause adds a join component.

`modifyJoinComponentClause`

This clause modifies a join component for a join.

`deleteJoinComponentClause`

This clause deletes a join component.

`unsetJoinComponentClauseDetails`

This clause updates the join component details.

Examples

```
OMBALTER ITEM_FOLDER 'SALES' SET PROPERTIES (DESCRIPTION) VALUES ('SALES')
```

See Also

OMBCREATE ITEM_FOLDER, OMBRETRIEVE ITEM_FOLDER

OMBALTER LIST_OF_VALUES

Purpose

Alters a list of values.

Prerequisites

Should be in the context of a business definition module or use the full path.

Syntax

```
alterListOfValuesCommand = ( OMBALTER LIST_OF_VALUES "QUOTED_STRING" ( (
  "renameClause" [ SET "setPropertiesClauseforLOVandD2D" ] [ SET
  "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
  { "alterListOfValuesClauses" } ) ) | ( SET
  "setPropertiesClauseforLOVandD2D" [ SET "setReferenceIconSetClause" ]
  [ UNSET "unsetReferenceIconSetClause" ] { "alterListOfValuesClauses" }
  ) | ( SET "setReferenceIconSetClause" [ UNSET
  "unsetReferenceIconSetClause" ] { "alterListOfValuesClauses" } ) | (
  UNSET "unsetReferenceIconSetClause" { "alterListOfValuesClauses" } ) ) |
  ( "alterListOfValuesClauses" { "alterListOfValuesClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClauseforLOVandD2D = PROPERTIES "("
  "propertyNameListforLOVandD2D" ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterListOfValuesClauses = SET ( REF | REFERENCE ) DEFINING ITEM
  "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING" | UNSET ( REF |
  REFERENCE ) "deleteListOfValuesClauses" |
  "addListOfValuesReferenceClause"
propertyNameListforLOVandD2D = ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) {
  "," ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) }
propertyValueList = "propertyValue" { "," "propertyValue" }
deleteListOfValuesClauses = ( DEFINING ITEM ) | ( ITEM "QUOTED_STRING" OF
  ITEM_FOLDER "QUOTED_STRING" )
addListOfValuesReferenceClause = SET ( REF | REFERENCE ) ITEM
  "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterListOfValuesCommand

This clause alters a list of values.

QUOTED_STRING

name of the list of values.

renameClause

Renames a list of values with a different name.

setPropertyClauseforLOVandD2D

This clause sets the properties of the object.

Basic properties for LIST_OF_VALUES:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the list of values

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the list of values

Name: DRILL_TO_DETAIL

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the list of values enables drilling between the item folders containing the items that use the list of values

Name: RETRIEVE_VALUES_GROUP_SIZE

Type: Number

Valid Values: N/A

Default: '100'

The number of rows to be fetched from the database at a time

Name: CACHE_VALUES

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the list of values should be cached in memory

Name: REQUIRE_SEARCH

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether search criteria should be requested

Name: SHOW_IN_NAVIGATOR

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be shown in the item navigator

Name: SORTED_DISTINCT

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be displayed sorted with duplicates hidden

Properties for LIST_OF_VALUES:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterListOfValuesClauses

This clause modifies a list of values.

DEFINING

This sets the defining item for the list of values.

propertyNameListforLOVandD2D

This is the list of property names.

propertyValueList

This is the list of property values.

deleteListOfValuesClauses

This deletes a reference to an item from a list of values.

addListOfValuesReferenceClause

This adds a reference to an item to a list of values.

propertyValue

This is a property value.

Examples

```
OMBALTER LIST_OF_VALUES 'LOV' SET PROPERTIES (DESCRIPTION) VALUES ('LOV')
```

See Also

OMBCREATE LIST_OF_VALUES, OMBRETRIEVE LIST_OF_VALUES

OMBALTER LOCATION

Purpose

Alter the location by renaming it, and/or reset its properties.

Prerequisites

Can be in any context.

Syntax

```
alterLocationCommand = OMBALTER ( LOCATION "QUOTED_STRING" (
    "renameClause" [ "setPropertyForModifyClause" [ "alterIconSetClause"
    ] ] | "setPropertyForModifyClause" [ "alterIconSetClause" ] |
    "alterIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertyForModifyClause = SET PROPERTIES "(" "propertyNameList" ")"
VALUES "(" "propertyValueList" ")"
alterIconSetClause = SET "setReferenceIconSetClause" [ UNSET
    "unsetReferenceIconSetClause" ] | UNSET "unsetReferenceIconSetClause"
    [ SET "setReferenceIconSetClause" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterLocationCommand

Alter the location specified by the quoted string.

renameClause

Rename the location to the value of the following quoted string.

setPropertyForModifyClause

Set or alter specified properties of the location.

alterIconSetClause

Set or unset the Icon Set of the location.

propertyNameList

The names of the properties whose values you want to set.

Properties for LOCATION:

Basic properties:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the location.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the location.

Name: TYPE

Type: STRING

Valid Values:

'CONCURRENT_MANAGER'

'AUTOSYS_AGENT'

'AUTOSYS_INSTANCE'

'BIBEANS'

'DISCOVERER'

'FILE_SYSTEM'

'OEM_AGENT'

'ORACLE_DATABASE'

'ORACLE_GATEWAY'

'ORACLE_WORKFLOW'

'SAP'

'TRANSPORTABLE_MODULE_SOURCE'

'TRANSPORTABLE_MODULE_TARGET'

Default: N/A

The type of system the location represents.

Name: VERSION

Type: STRING

Valid Values:

for 'CONCURRENT_MANAGER' : '11i'

for 'AUTOSYS_AGENT' : '0'

for 'AUTOSYS_INSTANCE' : '0'
 for 'BIBEANS' : '10.1'
 for 'DISCOVERER' : '10.1'
 for 'FILE_SYSTEM' : do not set version
 for 'OEM_AGENT' : '9.0','9.2'
 for 'ORACLE_DATABASE' : '8.1','9.0','9.2','10.1','10.2'
 for 'ORACLE_GATEWAY' : do not set version
 for 'ORACLE_WORKFLOW' : '2.6.2','2.6.3','2.6.4','11i'
 for 'SAP' : '4.x','3.x'
 for 'TRANSPORTABLE_MODULE_SOURCE' : '8.1','9.0','9.2','10.1','10.2'
 for 'TRANSPORTABLE_MODULE_TARGET' : '8.1','9.0','9.2','10.1','10.2'
 Default: N/A
 The version of the system(s) the location represents.

Lists of available properties for different types of LOCATION:

for 'CONCURRENT_MANAGER' :

TYPE,PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA,VERSION,APPLICATION,APPLICATION_USER,RESPONSIBILITY

for 'AUTOSYS_AGENT':

TYPE,VERSION,PASSWORD,HOST

for 'AUTOSYS_INSTANCE':

TYPE,VERSION,USER (or USER_NAME),PASSWORD,INSTANCE

for 'BIBEANS':

TYPE,VERSION,USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME

for 'DISCOVERER':

TYPE,VERSION,USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME

for 'FILE_SYSTEM':

TYPE,USER (or USER_NAME),PASSWORD,HOST,ROOTPATH

for 'OEM_AGENT':

TYPE,USER (or USER_NAME),PASSWORD,VERSION,DOMAIN,AGENT

for 'ORACLE_DATABASE':

TYPE,VERSION,CONNECT_AS_USER (or

```

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_
NAME,DATABASE_NAME,SCHEMA
for 'ORACLE_GATEWAY':
TYPE,CONNECT_AS_USER (or
USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_
NAME,SCHEMA
for 'ORACLE_WORKFLOW':
TYPE,VERSION,PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_
NAME,SCHEMA
for 'SAP':
TYPE, VERSION, USER (or USER_NAME), PASSWORD, APPLICATION_SERVER,
SYSTEM_NUMBER, CLIENT, LANGUAGE, HOST_LOGIN_USER, HOST_LOGIN_
PASSWORD,
FTP_DIRECTORY, EXECUTION_FM
for 'TRANSPORTABLE_MODULE_SOURCE':
TYPE, VERSION,CONNECT_AS_USER (or
USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,FTP_USER,FTP_
PASSWORD
for 'TRANSPORTABLE_MODULE_TARGET':
TYPE, VERSION,CONNECT_AS_USER (or
USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME

```

Some other properties for LOCATIONS:

Name: CONNECTION_TYPE

Type: STRING

Valid Values: 'HOST_PORT_SERVICE', 'SQL_NET_CONNECTION', 'DATABASE_LINK'

Default: 'HOST_PORT_SERVICE'

The location connection details format.

Name: HOST

Type: STRING

Valid Values: N/A

Default: N/A

The machine name.

Name: PORT
Type: NUMBER
Valid Values: 1 - 65535
Default: 1521
The port number of a database listener.

Name: SERVICE_NAME
Type: STRING
Valid Values: N/A
Default: N/A
The database service name.

Name: NET_SERVICE_NAME
Type: STRING
Valid Values: N/A
Default: N/A
The database netservice name.

Name: SCHEMA
Type: STRING
Valid Values: N/A
Default: N/A
The database schema name.

Name: PASSWORD
Type: STRING
Valid Values: N/A
Default: N/A
The password.

Name: USER
Type: STRING
Valid Values: N/A
Default: N/A
The user name.

Name: CONNECT_AS_USER

Synonym: USER_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: DOMAIN

Type: STRING

Valid Values: N/A

Default: N/A

The address of a machine running the Oracle Management Service.

Name: AGENT

Type: STRING

Valid Values: N/A

Default: N/A

The name of an Oracle Enterprise Manager (OEM) node running an OEM Agent.

This name must be entered exactly as shown under the nodes in the Oracle Management Service.

Name: ROOTPATH

Type: STRING

Valid Values: N/A

Default: N/A

The file system directory.

Name: APPLICATION

Type: STRING

Valid Values: N/A

Default: N/A

The Application name.

Name: APPLICATION_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: DATABASE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The Data Base name.

Name: RESPONSIBILITY

Type: STRING

Valid Values: N/A

Default: N/A

The responsibility role.

Name: APPLICATION_SERVER

Type: STRING

Valid Values: N/A

Default: N/A

The application server.

Name: SYSTEM_NUMBER

Type: STRING

Valid Values: N/A

Default: N/A

The number of SAP system.

Name: CLIENT

Type: STRING

Valid Values: N/A

Default: N/A

The client.

Name: LANGUAGE

Type: STRING

Valid Values: N/A

Default: N/A

The language of SAP.

Name: HOST_LOGIN_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user.

Name: HOST_LOGIN_PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: EXECUTION_FM

Type: STRING

Valid Values: N/A

Default: N/A

RFC Function Module for remote ABAP report execution

Name: FTP_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name used for creating ftp connection.

Name: FTP_PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The ftp password.

Name: FTP_DIRECTORY

Type: STRING

Valid Values: N/A

Default: N/A

The directory used in a ftp session

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList

The values for the named properties.

propertyValue

A property value.

Examples

```
OMBALTER LOCATION 'OLD_LOCATION' RENAME TO 'NEW_LOCATION' SET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a new location.', 'new  
location')
```

This will rename the location "OLD_LOCATION" to "NEW_LOCATION", and set its description to "This becomes a new location", set its business name to "new location".

See Also

OMBALTER, OMBCREATE LOCATION, OMBDROP LOCATION

OMBALTER MAPPING

Purpose

Alter the content of a mapping.

Prerequisites

1. The current context of scripting must be an Oracle Module
2. No concurrent user should be modifying the mapping

Syntax

```

alterMappingCommand = OMBALTER MAPPING "mappingName"
    "alterMapDetailClause"
mappingName = "QUOTED_STRING"
alterMapDetailClause = "renameClause" [ "alterPropertiesOrIconSetClause" ]
    [ "alterOperatorOwnerDescendantsClause"+ ] |
    "alterPropertiesOrIconSetClause" [
    "alterOperatorOwnerDescendantsClause"+ ] |
    "alterOperatorOwnerDescendantsClause"+
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] )
    | "setReferenceIconSetClause" ) ) | UNSET
    "unsetReferenceIconSetClause"
alterOperatorOwnerDescendantsClause = ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) | MODIFY ( "modifyOperatorClause" |
    "modifyGroupClause" | "modifyAttributeClause" | "modifyChildClause" )
    | DELETE ( "operatorBottomUpLocator" | "groupBottomUpLocator" |
    "attributeBottomUpLocator" | "childBottomUpLocator" |
    "deleteConnectionLocator" )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
modifyOperatorClause = "operatorBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyGroupClause = "groupBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyAttributeClause = "attributeBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )

```

```

modifyChildClause = "childBottomUpLocator" ( "renameClause" | SET
    "setPropertyClause" )
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
childBottomUpLocator = "childType" "childName" { OF "childType"
    "childName" } [ OF "mappableBottomUpLocator" ]
deleteConnectionLocator = CONNECTION ( FROM "mappableBottomUpLocator" [ TO
    "mappableBottomUpLocator" ] | TO "mappableBottomUpLocator" )
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
attributeName = "QUOTED_STRING"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { "," IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
pluggableMapName = "QUOTED_STRING"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
attributeNameList = "( " "attributeName" { "," "attributeName" } )"
bindableType = PLUGGABLE_MAPPING | OBJECT_TYPE | "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

alterMappingCommand

Alter the content of a mapping.

mappingName

Name of the mapping.

alterMapDetailClause

Alter the detail of the mapping.

renameClause

Rename a mapping, mapping operator, mapping group, or mapping attribute.

alterOperatorOwnerDescendantsClause

Alter the desired child objects applicable to a mapping or a pluggable mapping.

setPropertyClause

Describe the keys of properties for the map or objects in the map.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

addChildClause

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

The following is an example for creating a child object under a mapping

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

Note: Key word "OPERATOR" "GROUP" "ATTRIBUTE" are important for their respective ADD clauses. Without the key words, OMBPlus will interpret the ADD clause as an addChildClause. Here is an example:

OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'

OMB02932: Error getting child objects of type TABLE in M1

OMBPLUS interprets the ADD clause as one for creating a non-operator child object under the mapping object. Therefore, it tries to find type definition for non-operator child object "TABLE" and cannot find it.

addConnectionClause

Add connections between mapping groups or mapping attributes.

modifyOperatorClause

Modify a mapping operator.

modifyGroupClause

Modify a mapping group.

modifyAttributeClause

Modify a mapping attribute.

modifyChildClause

Modify a child that belongs to a mapping, mapping operator, mapping group or mapping attribute.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

childBottomUpLocator

Location of the child that belongs to a map, mapping operator, mapping group or mapping attribute.

deleteConnectionLocator

Delete connections between mapping operators, mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCAST, LCRSPLITTER, MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION, TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

attributeName

Name of a mapping attribute.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributesBottomUpLocator

Location of a list of mapping attributes.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Length of the attribute.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Precision of the attribute.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for MAPPING:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_LANGUAGE

Type: STRING

Valid Values: ABAP, PLSQL, SQLLOADER, UNDEFINED

Default: UNDEFINED

The language used when generating code for the mapping.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box;

this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE, NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2, NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4, NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE, NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME, NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the

update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAM, NA_COUNTYNAM, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,

NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,

NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,

NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,

NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,

NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,

NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,

NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA, NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE, NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR, NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4, NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is

specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | |=

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data;

it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: ≥ 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)
Valid Values: INSERT, NONE, UPDATE
Default: INSERT
The loading operation to be performed

Name: OUTPUT_AS_XML
Type: BOOLEAN
Valid Values: true, false
Default: false
Output data to file in XML format.

Name: RECORD_DELIMITER
Type: STRING
Valid Values: N/A
Default: "
Character that indicates the end of the record.

Name: RECORD_SIZE
Type: NUMBER
Valid Values: N/A
Default: 0
Size of a fixed length record.

Name: RECORD_TYPE_LENGTH
Type: NUMBER
Valid Values: N/A
Default: 0
If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION
Type: NUMBER
Valid Values: N/A
Default: 0
If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default

it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"MATCHMERGE.GENERAL.MATCH_NEW_RECORD_
CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by

constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS

report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For

performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using

the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN
Valid Values: true, false
Default: true
Enable Constraints

Name: ERROR_SELECT_FILTER
Type: STRING(3)
Valid Values: NO, YES
Default: YES
Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP
Type: STRING(3)
Valid Values: NO, YES
Default: YES
Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME
Type: STRING(32)
Valid Values: N/A
Default: "
The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS
Type: BOOLEAN
Valid Values: true, false
Default: false
Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME
Type: STRING
Valid Values: N/A
Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING
Valid Values: N/A
Default: "
Partition Name

Name: PEL_ENABLED
Type: BOOLEAN
Valid Values: true, false
Default: false
PEL Enabled

Name: PRIMARY_SOURCE
Type: STRING
Valid Values: NO, YES
Default: NO
A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP
Type: NUMBER
Valid Values: >= 0
Default: 0
Number of records to skip

Name: REPLACE_DATA
Type: BOOLEAN
Valid Values: true, false
Default: false
Replace existing data in target partition if there is any.

Name: ROW_COUNT
Type: STRING
Valid Values: N/A
Default: "
Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: SINGLEROW
Type: BOOLEAN
Valid Values: true, false
Default: false
Singlerow

Name: SORTED_INDEXES_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
Sorted Indexes Clause

Name: SUBPARTITION_NAME
Type: STRING
Valid Values: N/A
Default: "
Subpartition Name

Name: TARGET_FILTER_FOR_DELETE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name

of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this
operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name,
so all errors generated by a particular input record will be rolled up into
a single record with the error names concatenated in the error name
attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: ≥ 0

Default: 0
Number of records to skip

Name: ROW_COUNT
Type: STRING
Valid Values: N/A
Default: "
Row count

Name: ROW_COUNT_ENABLED
Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: SINGLEROW
Type: BOOLEAN
Valid Values: true, false
Default: false
Singlerow

Name: SORTED_INDEXES_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
Sorted Indexes Clause

Name: SUBPARTITION_NAME
Type: STRING
Valid Values: N/A
Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

pluggableMapName

Name of the pluggable map.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBALTER MAPPING 'MAP1' RENAME TO 'MAP2'
```

```
OMBALTER MAPPING 'MAP1'  
ADD CONNECTION FROM GROUP 'INOUTGRP1' OF OPERATOR 'CUST_SRC'  
TO GROUP 'INOUTGRP1' OF OPERATOR 'CUST_LOOK_UP'
```

```
OMBALTER MAPPING 'MAP1' DELETE OPERATOR 'OP1'
```

```
OMBALTER MAPPING 'MAP1' DELETE VARIABLE 'LAST_CUST'
```

```
OMBALTER MAPPING 'MAP1'  
MODIFY VARIABLE 'LAST_CUST'  
SET PROPERTIES (DATATYPE, LENGTH) VALUES ('VARCHAR2', 100)
```

See Also

OMBALTER, OMBCREATE MAPPING, OMBRETRIEVE MAPPING, OMBDROP MAPPING

OMBALTER MATERIALIZED_VIEW

Purpose

To alter properties and definition of a materialized view.

Prerequisites

In the context of an Oracle Module.

Syntax

```
alterMaterializedViewCommand = OMBALTER ( MATERIALIZED_VIEW
    "QUOTED_STRING" ( "renameClause" [ "alterPropertiesOrIconSetClause" ]
    [ "alterMaterializedViewSCOandDependentClauses" ] |
    "alterPropertiesOrIconSetClause" [
    "alterMaterializedViewSCOandDependentClauses" ] |
    "alterMaterializedViewSCOandDependentClauses" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertyClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
alterMaterializedViewSCOandDependentClauses = ADD (
    "addColumnClauseForAlter" [
    "alterMaterializedViewSCOandDependentClauses" ] |
    "addViewConstraintClause" { "alterViewConstraintClauses" } |
    "addSCOClause" { "alterMaterializedViewSCOclauses" } |
    "addDataRuleUsageClause" { "alterDataRuleUsageClauses" } |
    "addRelationalDependentClause" [
    "alterMaterializedViewSCOandDependentClauses" ] ) | MODIFY (
    "modifyColumnClause" [ "alterMaterializedViewSCOandDependentClauses" ]
    | "modifyViewConstraintClause" { "alterViewConstraintClauses" } |
    "modifySCOClause" { "alterMaterializedViewSCOclauses" } |
    "modifyDataRuleUsageClause" { "alterDataRuleUsageClauses" } ) | DELETE
    ( "deleteColumnClause" [
    "alterMaterializedViewSCOandDependentClauses" ] |
    "deleteViewConstraintClause" { "alterViewConstraintClauses" } |
    "deleteSCOClause" { "alterMaterializedViewSCOclauses" } |
    "deleteDataRuleUsageClause" { "alterDataRuleUsageClauses" } |
    "deleteRelationalDependentClause" [
    "alterMaterializedViewSCOandDependentClauses" ] )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
addColumnClauseForAlter = COLUMN "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertyClause" ]
addViewConstraintClause = "addUkPkClause" | "addFkClause"
alterViewConstraintClauses = ADD "addViewConstraintClause" | MODIFY
    "modifyViewConstraintClause" | DELETE "deleteViewConstraintClause"
addSCOClause = "addIndexClause" | "addIndexPartitionClause" |
    "addIndexPartitionKeyClause" | "addPartitionClause" |
    "addPartitionKeyClause" | "addSubpartitionClause" |
    "addaddMaterializedViewSCOandDependentClauseClause" |
    "addSubPartitionKeyClause" | "addIndexColumnClause"
alterMaterializedViewSCOclauses = ADD "addSCOClause" | MODIFY
    "modifySCOClause" | DELETE "deleteSCOClause"
```



```

addDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" SET REF DATA_RULE
    "QUOTED_STRING" ( GROUP "QUOTED_STRING" SET REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) "QUOTED_STRING" ( ATTRIBUTE
    "QUOTED_STRING" SET REF COLUMN "QUOTED_STRING" )+ )+ [ SET
    "setPropertiesClause" ]
alterDataRuleUsageClauses = ADD "addDataRuleUsageClause" | MODIFY
    "modifyDataRuleUsageClause" | DELETE "deleteDataRuleUsageClause"
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING"
modifyColumnClause = COLUMN "QUOTED_STRING" ( "renameClause" [
    "moveToClause" ] [ SET "setPropertiesClause" ] | "moveToClause" [ SET
    "setPropertiesClause" ] | SET "setPropertiesClause" )
modifyViewConstraintClause = "modifyUkPkClause" | "modifyFkClause"
modifySCOClauses = "modifyIndexClause" | "modifyIndexPartitionClause" |
    "modifyIndexPartitionKeyClause" | "modifyPartitionClause" |
    "modifyPartitionKeyClause" |
    "modifyaddMaterializedViewSCOandDependentClauseClause" |
    "modifySubPartitionClause" | "modifySubPartitionKeyClause" |
    "modifyIndexColumnClause"
modifyDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" )
deleteColumnClause = COLUMN "QUOTED_STRING"
deleteViewConstraintClause = UNIQUE_KEY "QUOTED_STRING" | PRIMARY_KEY
    "QUOTED_STRING" | FOREIGN_KEY "QUOTED_STRING"
deleteSCOClauses = INDEX "QUOTED_STRING" | PARTITION "QUOTED_STRING" |
    PARTITION_KEY "QUOTED_STRING" | TEMPLATE_SUBPARTITION "QUOTED_STRING"
    | SUBPARTITION_KEY "QUOTED_STRING" | INDEX_COLUMN "QUOTED_STRING" OF
    INDEX "QUOTED_STRING" | INDEX_PARTITION "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" | INDEX_PARTITION_KEY "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" | SUBPARTITION "QUOTED_STRING" OF PARTITION
    "QUOTED_STRING"
deleteDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING"
deleteRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" [ SET
    "setUkPkPropertiesAndReferencesColumnsClauses" ]
addFkClause = FOREIGN_KEY "QUOTED_STRING" [ SET "setFkSubClauses" ]
addIndexClause = INDEX "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addIndexPartitionClause = INDEX_PARTITION "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] (
    "renameSCOConfigurationClause" [ SET
    "setSCOConfigurationPropertiesClauses" ] | [ SET
    "setSCOConfigurationPropertiesClauses" ] )
addIndexPartitionKeyClause = INDEX_PARTITION_KEY "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionClause = PARTITION "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionKeyClause = PARTITION_KEY "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addSubpartitionClause = SUBPARTITION "QUOTED_STRING" OF PARTITION
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
    "setSCOConfigurationPropertiesClauses" ]
addaddMaterializedViewSCOandDependentClauseClause = TEMPLATE_SUBPARTITION
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
    "setSCOConfigurationPropertiesClauses" ]
addSubPartitionKeyClause = SUBPARTITION_KEY "QUOTED_STRING" [ SET

```

```

    "setSCOConfigurationPropertiesClauses" ]
addIndexColumnClause = INDEX_COLUMN "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
modifyUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" (
    "renameClause" [ SET "setUkPkPropertiesAndReferencesColumnsClauses" ]
    | SET "setUkPkPropertiesAndReferencesColumnsClauses" )
modifyFkClause = FOREIGN_KEY "QUOTED_STRING" ( "renameClause" [ SET
    "setFkSubClauses" ] | SET "setFkSubClauses" )
modifyIndexClause = INDEX "QUOTED_STRING" ( "renameSCOConfigurationClause"
    [ SET "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifyIndexPartitionClause = INDEX_PARTITION "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" ( "renameSCOConfigurationClause" [
    "moveToClauseIndexPartition" ] [ SET
    "setSCOConfigurationPropertiesClauses" ] |
    "moveToClauseIndexPartition" [ SET
    "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifyIndexPartitionKeyClause = INDEX_PARTITION_KEY "QUOTED_STRING" OF
    INDEX "QUOTED_STRING" ( SET "setSCOConfigurationPropertiesClauses" )
modifyPartitionClause = PARTITION "QUOTED_STRING" (
    "renameSCOConfigurationClause" [ "moveToClausePartition" ] [ SET
    "setSCOConfigurationPropertiesClauses" ] | "moveToClausePartition" [
    SET "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifyPartitionKeyClause = PARTITION_KEY "QUOTED_STRING" (
    "renameSCOConfigurationClause" [ SET
    "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifyaddMaterializedViewSCOandDependentClauseClause =
    TEMPLATE_SUBPARTITION "QUOTED_STRING" ( "renameSCOConfigurationClause"
    [ "moveToClauseTemplateSubPartition" ] [ SET
    "setSCOConfigurationPropertiesClauses" ] |
    "moveToClauseTemplateSubPartition" [ SET
    "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifySubPartitionClause = SUBPARTITION "QUOTED_STRING" OF PARTITION
    "QUOTED_STRING" ( "renameSCOConfigurationClause" [
    "moveToClauseSubPartition" ] [ SET
    "setSCOConfigurationPropertiesClauses" ] | "moveToClauseSubPartition"
    [ SET "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifySubPartitionKeyClause = SUBPARTITION_KEY "QUOTED_STRING" (
    "renameSCOConfigurationClause" [ SET
    "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
modifyIndexColumnClause = INDEX_COLUMN "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" ( "renameSCOConfigurationClause" [
    "moveToClauseForIndexColumn" ] [ SET
    "setSCOConfigurationPropertiesClauses" ] |
    "moveToClauseForIndexColumn" [ SET
    "setSCOConfigurationPropertiesClauses" ] | SET
    "setSCOConfigurationPropertiesClauses" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setUkPkPropertiesAndReferencesColumnsClauses = "setPropertiesClause" [ SET
    ( REF | REFERENCE ) "constraintColumnReferencesClause" ] | ( REF |
    REFERENCE ) "constraintColumnReferencesClause"
setFkSubClauses = "setPropertiesClause" [ SET ( REF | REFERENCE )

```

```

"setFkReferencesClauses" ] | ( REF | REFERENCE )
"setFkReferencesClauses"
setSCOConfigurationPropertiesClauses = PROPERTIES "(" "propertyNameList"
)" VALUES "(" "propertyValueList" ")"
renameSCOConfigurationClause = RENAME TO "QUOTED_STRING"
moveToClauseIndexPartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClausePartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClauseTemplateSubPartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClauseSubPartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClauseForIndexColumn = MOVE TO POSITION "INTEGER_LITERAL"
constraintColumnReferencesClause = COLUMNS "(" "quotedNameList" ")"
setFkReferencesClauses = "constraintColumnReferencesClause" [ SET ( REF |
REFERENCE ) "constraintUkReferencesClause" ] |
"constraintUkReferencesClause" [ SET ( REF | REFERENCE )
"constraintColumnReferencesClause" ]
quotedNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
constraintUkReferencesClause = ( UNIQUE_KEY | PRIMARY_KEY )
"QUOTED_STRING" [ OF ( TABLE | VIEW ) "QUOTED_STRING" ]

```

Keywords And Parameters

alterMaterializedViewCommand

This clause alters a materialized view.

QUOTED_STRING

name of the materialized view.

renameClause

renames a table with a different name.

setPropertyClause

Used to set properties (core, logical, physical, user-defined) for materialized views (including partitions and subpartitions) and their columns, indexes (including index partitions), unique keys, foreign keys, and primary keys.

Note:

Constraints can be specified but will not be generated for either View or Materialized View in this release.

Basic properties for VIEW, MATERIALIZED_VIEW:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the View, MaterializedView

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the View, MaterializedView

Name: VIEW_QUERY

Type: STRING(4000)

Valid Values: N/A

Default: "

Sets the query definition in View and MaterializedView.

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Default value of the column

Name: NOT_NULL

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify "true" to enforce Not Null restriction on a column.

Basic properties for PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the table rows according to a Hash Algorithm, lists of values, or specified ranges.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH partitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of partitions that is a power of 2. If you have multiple Partition Keys, you only have to specify once.

Basic properties for PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify

MAXVALUE(s) as the value(s) of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Commas can be escaped using "" (for example '1,2,3'). Always specify DEFAULT as the value of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH customized subpartitions the database should create for a particular main RANGE partition (RANGE-HASH BY QUANTITY partitioning). For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once. Set it to 0 to reverse to the use of generic template HASH_QUANTITY specified in SUBPARTITION_KEY.

Basic properties for SUBPARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

For partition-level partitioning according to a Hash Algorithm or lists of values. Each partition is further sorted into subpartitions.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH subpartitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once.

Basic properties for SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST subpartition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any SUBPARTITION. Each LIST subpartition must have at least one value. No value, including NULL, can appear in more than one subpartition.

Basic properties for TEMPLATE_SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

In composite partitioning, template subpartitions are automatically applied to those partitions without their subpartitions specified. Here for LIST subpartitions only, specify a comma-delimited, ordered list of literal values corresponding to the LIST subpartitioning column. Always specify DEFAULT as the value of the last template LIST subpartition, and and make sure you have specified SUBPARTITION_KEY(s) before you specify any TEMPLATE_SUBPARTITION. Each LIST template subpartition must have at least one value. No value, including NULL, can appear in more than one template

subpartition.

Basic properties for INDEX:

Name: INDEX_TYPE

Type: STRING

Valid Values: UNIQUE, NON-UNIQUE, BITMAP, FUNCTION-BASED

Default: (No default, must be one of the preceding choices)

Specify the type of an index. NORMAL can be used in place of NON-UNIQUE.

Name: LOCAL_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify if an index is Global or Local. The default is Global.

Specify Local so that the index is partitioned on the same columns, with the same number of partitions and the same partition bounds as table.

Oracle Database automatically maintains local index partitioning as the underlying table is repartitioned.

Name: COLUMN_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

Specify an expression built from columns of table, constants, SQL functions, and user-defined functions to create a FUNCTION-BASED index.

Basic properties for INDEX_PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the index rows according to a Hash Algorithm, lists of values, or specified ranges. Hash index partitioning is supported

starting with Oracle Database 10g version.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH index partitions the database should create based on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of index partitions that is a power of 2. If you have multiple index Partition Keys, you only have to specify once.

Basic properties for INDEX_PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition. No need to specify VALUES_LESS_THAN for Local index.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last partition. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition. No need to specify VALUES_EQUAL_TO for Local index.

Basic properties for CHECK_CONSTRAINTS:

Name: CHECK_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Specify a condition that each row in the table must satisfy.

Properties for MATERIALIZED_VIEW:

Name: BASE_TABLES

Type: STRING

Valid Values: N/A

Default: "

Specify a comma separated list of base tables for generating materialized view log.

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for the database object. The default is DEFAULT.

Name: BUILD

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE, PREBUILT

Default: "

Specify IMMEDIATE to populate the view when it is created. Specify DEFERRED to delays population until the next refresh operation. IMMEDIATE is the default.

Name: CONSTRAINTS

Type: STRING

Valid Values: , ENFORCED, TRUSTED

Default: "

Specify TRUSTED to let Oracle Database use dimension and constraint information that has been declared trustworthy by the database administrator but that has not been validated by the database. If the dimension and constraint information is valid, then performance may improve. However, if this information is invalid, then the refresh procedure may corrupt the materialized view even though it returns a success status. ENFORCED is the default.

Name: DEFAULTINDEXBUFFERPOOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for the database object. The default is DEFAULT.

Name: DEFAULTINDEXFREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: DEFAULTINDEXFREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the database object. The default is 1.

Name: DEFAULTINDEXINITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: DEFAULTINDEXINITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2.

Name: DEFAULTINDEXMAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the database object.

Name: DEFAULTINDEXMAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: DEFAULTINDEXMINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: DEFAULTINDEXNEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: DEFAULTINDEXPCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: DEFAULT_INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Specify tablespace for default index storage.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FOR_UPDATE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES (FOR UPDATE) to allow a subquery, primary key, object, or rowid materialized view to be updated. When used in conjunction with Advanced Replication, these updates will be propagated to the master. The default is NO.

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the database object. The default is 1.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: HASH_PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

A comma separated list of tablespaces to use for [sub]partition storage.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is

LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the database object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: NEXTDATE

Type: STRING

Valid Values: N/A

Default: "

Specify a datetime expression for calculating the interval between automatic refreshes.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Specify NOPARALLEL for serial execution. This is the default. Specify PARALLEL if you want Oracle to select a degree of parallelism equal to the number of CPUs available on all participating instances times the value of the PARALLEL_THREADS_PER_CPU initialization parameter.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Specify the number of parallel threads used in the parallel operation. Normally Oracle calculates the optimum degree of parallelism, so it is not necessary for you to specify it.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The

default is 40.

Name: QUERY_REWRITE

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE to mark the materialized view eligible for query rewrite or DISABLE to mark the materialized view ineligible for query rewrite. DISABLE is the default.

Name: REFRESH

Type: STRING

Valid Values: , COMPLETE, FAST, FORCE, NEVER

Default: "

Specify FAST to indicate the incremental refresh method. Specify COMPLETE to indicate the complete refresh method, which is implemented by executing the defining query of the materialized view. Specify FORCE to indicate that when a refresh occurs, Oracle Database will perform a fast refresh if one is possible or a complete refresh otherwise. FORCE is the default. Specify NEVER to prevent the materialized view from being refreshed with any Oracle Database refresh mechanism or packaged procedure.

Name: REFRESH_ON

Type: STRING

Valid Values: , COMMIT, DEMAND

Default: "

Specify COMMIT to indicate that a fast refresh is to occur whenever the database commits a transaction that operates on a master table of the materialized view. Specify DEMAND to indicate that the materialized view will be refreshed on demand by calling one of the three DBMS_MVIEW refresh procedures. DEMAND is the default.

Name: ROLLBACK

Type: STRING

Valid Values: , DEFAULT, DEFAULT LOCAL, DEFAULT MASTER, NONE

Default: DEFAULT LOCAL

Specify DEFAULT for Oracle Database to choose automatically which rollback

segment to use. Specify `DEFAULT MASTER` for the remote rollback segment to be used at the remote master site for the individual materialized view. Specify `DEFAULT LOCAL` for the remote rollback segment to be used for the local refresh group that contains the materialized view. `DEFAULT LOCAL` is the default. Specify `NONE` to name both master and local rollback segments.

Name: `ROLLBACKSEGMENTLOCAL`

Type: `STRING`

Valid Values: `N/A`

Default: `"`

Specify a named remote rollback segment to be used for the local refresh group that contains the materialized view. Default is null. Ignore if `DEFAULT` or `DEFAULT LOCAL` is specified for default rollback segment.

Name: `ROLLBACKSEGMENTMASTER`

Type: `STRING`

Valid Values: `N/A`

Default: `"`

Specify a named remote rollback segment to be used at the remote master site for the individual materialized view. Default is null. Ignore if `DEFAULT` or `DEFAULT MASTER` is specified for default rollback segment.

Name: `SHADOW_TABLESPACE`

Type: `STRING(30)`

Valid Values: `N/A`

Default: `"`

Use the `Tablespace` parameter to specify the name of tablespace.

Name: `SHADOW_TABLE_NAME`

Type: `STRING(30)`

Valid Values: `N/A`

Default: `"`

Use the shadow table name to specify the name of Shadow Table.

Name: `STARTWITH`

Type: `STRING`

Valid Values: `N/A`

Default: "

Specify a datetime expression for the first automatic refresh time.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: USING_INDEX_MODE

Type: STRING

Valid Values: , USING_INDEX, USING_NO_INDEX

Default: "

Specify USING_NO_INDEX to suppress the creation of the default index for Materialized View. You can create an alternative index for a Materialized View explicitly. The default is USING_INDEX.

Name: WITH

Type: STRING

Valid Values: , PRIMARY_KEY, ROWID

Default: "

Specify PRIMARY KEY to create a primary key materialized view. Specify ROWID to create a rowid materialized view. Rowid materialized views are useful if the materialized view does not include all primary key columns of the master tables. Rowid materialized views must be based on a single table and meet other restrictions. PRIMARY KEY is the default.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that

in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify `ENABLE` if you want the constraint to be applied to the data in the table. Specify `DISABLE` to disable the integrity constraint. The default is `ENABLE`.

Name: `EXCEPTIONSINTO`

Type: `STRING`

Valid Values: `N/A`

Default: `"`

Specify an exceptions table (`[schema.]table`). The `EXCEPTIONS` table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with `NOVALIDATE` option.

Name: `INITIALLY`

Type: `STRING`

Valid Values: `, DEFERRED, IMMEDIATE`

Default: `"`

Specify `(INITIALLY) IMMEDIATE` to indicate that Oracle should check a `DEFERRABLE` constraint at the end of each subsequent SQL statement. Specify `(INITIALLY) DEFERRED` to indicate that Oracle should check a `DEFERRABLE` constraint at the end of subsequent transactions. The default is `(INITIALLY) IMMEDIATE`.

Name: `ONDELETE`

Type: `STRING`

Valid Values: `, CASCADE, SET NULL`

Default: `"`

Specify `CASCADE` if you want Oracle to remove dependent foreign key values. Specify `SET NULL` if you want Oracle to convert dependent foreign key values to `NULL`.

Name: `RELY`

Type: `STRING`

Valid Values: `, NORELY, RELY`

Default: `"`

Specify `RELY` to activate an existing constraint in `NOVALIDATE` mode for

query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for CHECK_CONSTRAINT:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to

defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for INDEX:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: COMPUTESTATISTICS

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to collect statistics at relatively little cost during the creation of an index.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: INDEXORDER

Type: STRING

Valid Values: , ASC, DESC

Default: "

Use ASC or DESC to indicate whether the index should be created in ascending or descending order. The Oracle default is ASC.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2 for Index.

Name: KEYCOMPRESS

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Specify COMPRESS to enable key compression.

Name: KEYCOMPRESSPREFIXLENGTH

Type: STRING

Valid Values: N/A

Default: "

Specify the prefix length (number of prefix columns to compress). For unique indexes, the valid range of prefix length values is from 1 to the number of key columns minus 1. The default prefix length is the number of key columns minus 1. For nonunique indexes, the valid range of prefix length values is from 1 to the number of key columns. The default prefix length is the number of key columns. Oracle compresses only nonpartitioned indexes that are nonunique or unique indexes of at least two columns. You cannot specify COMPRESS for a bitmap index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (2-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: ONLINE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to indicate that DML operations on the table will be allowed during creation of the index.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of Index tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow Index tablespaces. The number of Index tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of Index tablespaces, then Oracle cycles through the names of the Index tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used

in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for a locally partitioned index. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. If specified, then individual local Hash index partitions specified will be ignored for Local Hash or Range-Hash Index partitioning.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: SORT

Type: STRING

Valid Values: , NOSORT, REVERSE, SORT

Default: "

Specify NOSORT to indicate to Oracle that the rows are already stored in the database in ascending order. Specify REVERSE to store the bytes of the index block in reverse order, excluding the rowid.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PARTITION:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1 for Table and 2 for Index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addColumnClauseForAlter

This clause adds a column at a particular position.

When you alter a table and add columns to it, the position you specify for a new column must be less than or equal to the number of columns added up to that point in the OMBALTER command.

For example, a table TEMP_TAB contains three columns. You use the following

OMBALTER TABLE command to add three more columns:

```
OMBALTER TABLE 'TEMP_TAB' \  
ADD COLUMN 'C4' AT POSITION 4 \  
SET PROPERTIES(DATATYPE,PRECISION) VALUES('NUMBER',7) \  
ADD COLUMN 'C5' AT POSITION 6 \  
SET PROPERTIES(DATATYPE) VALUES('VARCHAR2') \  
ADD COLUMN 'C6' AT POSITION 5 \  
SET PROPERTIES(DATATYPE,PRECISION) VALUES('NUMBER',10);
```

This command does not execute successfully because at the point when you specify the position of the column C5 as 6, the table contains only 5 columns.

QUOTED_STRING

The column name.

addViewConstraintClause

This clause adds the view's configuration clause.

alterViewConstraintClauses

This clause alters the view's constraint clause.

addSCOClause

This clause will add SCOs.

addDataRuleUsageClause

Add a data rule usage to the relation.

alterDataRuleUsageClauses

Add, modify, or delete data rule usages.

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

modifyColumnClause

This clause renames, set properties, and move columns.

modifyViewConstraintClause

This clause modifies the view's constraint clause.

modifySCOClause

This clause will modify SCOs.

modifyDataRuleUsageClause

Rename or modify the properties of a data rule usage.

deleteColumnClause

This clause deletes a column.

deleteViewConstraintClause

This clause deletes the view's constraint.

deleteSCOClause

This clause deletes a SCO.

QUOTED_STRING

Either index, partition, partition_key, or index column name.

deleteDataRuleUsageClause

Delete a data rule usage.

deleteRelationalDependentClause

This clause deletes referential dependencies to other relational objects.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

addUkPkClause

This clause adds the adds unique key and primary keys.

QUOTED_STRING

name of the unique key or primary key.

addFkClause

This clause adds foreign key.

QUOTED_STRING

Name of the foreign key.

addIndexClause

This clause adds an index.

QUOTED_STRING

Name of the index.

addPartitionClause

This clause adds a partition.

QUOTED_STRING

Name of the partition.

addPartitionKeyClause

This clause adds a partition key.

QUOTED_STRING

Name of the partition key. This should be a column identifier.

addIndexColumnClause

This clause will add index column to a specified index.

QUOTED_STRING

This should be a column identifier of owning object (such as a table) of the index.

moveToClause

This clause will move the column to given position.

`modifyUkPkClause`

It modifies unique or primary key.

`modifyFkClause`

This clause modifies the foreign key.

`modifyIndexClause`

This clause modifies the Index.

QUOTED_STRING

Name of the index.

`modifyPartitionClause`

This clause modifies a partition.

QUOTED_STRING

Name of the partition.

`modifyPartitionKeyClause`

This clause modifies a partition key.

QUOTED_STRING

Name of the partition key.

`modifyIndexColumnClause`

Modifies the Index Column. The first `quoted_string` in this clause denotes index column name, and the latter denotes index.

`propertyValue`

This clause adds the property values.

`setUkPkPropertiesAndReferencesColumnsClauses`

This clause adds properties and references to columns.

`setFkSubClauses`

This clause set references to a foreign key.

setSCOConfigurationPropertiesClauses

Set the configuration properties for the following:

- Partition, Subpartition, and Template Subpartition: All refer to configuration properties of Partition.
- Index, and Index Partition: For Index Partition, refer to configuration properties of Partition.

renameSCOConfigurationClause

This clause renames configuration objects.

constraintColumnReferencesClause

This clause provides names of all columns.

setFkReferencesClauses

This clause sets foreign key references.

quotedNameList

This clause gives column names.

constraintUkReferencesClause

The first QUOTED_STRING denotes the UniqueKey or Primary key name, and the latter denotes the table's or view's name.

Examples

```
OMBALTER MATERIALIZED_VIEW 'NEW_MATERIALIZED_VIEW' SET
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('this is an altered desc of new
materialized view', 'Altered New MaterializedView')
```

This will alter a materialized view named "NEW_MATERIALIZED_VIEW", its description is "this is an altered desc of new materialized view", and business name is "Altered New MaterializedView".

See Also

```
OMBALTER, OMBCREATE MATERIALIZED_VIEW, OMBDROP MATERIALIZED_
VIEW, OMBRETRIEVE MATERIALIZED_VIEW
```

OMBALTER MDL_ACTION_PLAN

Purpose

Modify an existing metadata loader action plan.

Prerequisites

Connection must be established to the repository.

Syntax

```
alterMDLActionPlanCommand = ( OMBALTER ( MDL_ACTION_PLAN ) "QUOTED_STRING"
    ( "alterActionPlanClause" { "alterActionPlanClause" } ) )
alterActionPlanClause = "renameActionPlanClause" | "addActionClause" |
    "deleteActionClause" | "modifyActionClause"
renameActionPlanClause = RENAME TO "QUOTED_STRING"
addActionClause = ADD ACTION "QUOTED_STRING" "setReferenceClause"
deleteActionClause = DELETE ACTION "QUOTED_STRING"
modifyActionClause = MODIFY ACTION "QUOTED_STRING" "modifyActionOperation"
    { "modifyActionOperation" }
setReferenceClause = SET ( REF | REFERENCE ) ( "referenceValueClause" )
modifyActionOperation = "renameActionClause" | "setReferenceClause" |
    "deleteReferenceClause"
referenceValueClause = "objectTypeValue" "QUOTED_STRING" [ SET ( REF |
    REFERENCE ) "referenceValueClause" ]
renameActionClause = RENAME TO "QUOTED_STRING"
deleteReferenceClause = DELETE REFERENCE "objectTypeValue" "QUOTED_STRING"
objectTypeValue = ( PROJECT | ORACLE_MODULE | TABLE | VIEW | SEQUENCE |
    MATERIALIZED_VIEW | FUNCTION | PROCEDURE | PACKAGE | DIMENSION | CUBE
    | ADVANCED_QUEUE | STREAMS_QUEUE | MAPPING | REAL_TIME_MAPPING |
    PROCESS_FLOW_MODULE | PROCESS_FLOW_PACKAGE | PROCESS_FLOW | SAP_MODULE
    | CMI_MODULE | GATEWAY_MODULE | EXTERNAL_TABLE | FLAT_FILE_MODULE |
    FLAT_FILE | BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE
    | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL |
    ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
    PRESENTATION_TEMPLATE | LOCATION | CONNECTOR | CONTROL_CENTER |
    CONFIGURATION | COLLECTION | SNAPSHOT | ROLE | USER | ICONSET |
    TRANSFORMATION_MODULE | CALENDAR_MODULE | CALENDAR_FOLDER | CALENDAR |
    EXPERT_MODULE | EXPERT | DATA_RULE_MODULE | DATA_RULE | DATA_AUDITOR
    | STREAMS_CAPTURE_PROCESS | QUEUE_TABLE | QUEUE_PROPAGATION |
    OBJECT_TYPE | NESTED_TABLE | VARYING_ARRAY | DEPLOYMENT | DATA_PROFILE
    | PROFILE_REFERENCE | PLSQL_TABLE_TYPE | PLSQL_RECORD_TYPE |
    PLSQL_REF_CURSOR_TYPE | PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER |
    CMI_DEFINITION | ACTIVITY_TEMPLATE | ACTIVITY_TEMPLATE_FOLDER |
    TRANSPORTABLE_MODULE )
```

Keywords And Parameters

alterMDLActionPlanCommand

Modify an existing metadata loader action plan.

alterActionPlanClause

The type of altering that can be performed on the action plan.

renameActionPlanClause

Rename an action plan.

addActionClause

Add an action to an action plan.

deleteActionClause

Removes an action from an action plan.

modifyActionClause

Modify an action of an action plan.

setReferenceClause

Specify the object type and the absolute path name of an object.

modifyActionOperation

The type of operations that can be performed on an action.

referenceValueClause

Specify a first-class object type and the absolute path name of an object.

QUOTED_STRING

Absolute path name of an object (for example '/MY_PROJECT/MODULE_X/TABLE_Y').

renameActionClause

Rename an action of an action plan.

deleteReferenceClause

Remove a reference object from an action plan.

objectTypeValue

The first-class object type that is allowed to be specified in the referenceValueClause.

Examples

```
OMBALTER MDL_ACTION_PLAN 'MY_PROJECT_ACTION_PLAN'  
RENAME TO 'MY_PROJECT_ALLOBJECTS_ACTION_PLAN'
```

```
OMBALTER MDL_ACTION_PLAN 'MULTIPROJECT_ACTION_PLAN'  
ADD ACTION 'MULTI_PROJECTS'  
SET REFERENCE PROJECT '/MY_PROJECT'  
SET REFERENCE PROJECT '/PUBLIC_PROJECT'
```

```
OMBALTER MDL_ACTION_PLAN 'MULTIPROJECT_ACTION_PLAN'  
MODIFY ACTION 'MULTI_PROJECTS'  
SET REFERENCE PROJECT '/TEST_PROJECT'
```

```
OMBALTER MDL_ACTION_PLAN 'MULTIPROJECT_ACTION_PLAN'  
MODIFY ACTION 'MULTI_PROJECTS'  
RENAME TO 'ONLY_TWO_PROJECTS'
```

```
OMBALTER MDL_ACTION_PLAN 'GRANULAR_OBJS_ACTION_PLAN'  
MODIFY ACTION 'GRANULAR_OBJS'  
DELETE REFERENCE FLAT_FILE '/MY_PROJECT/FLAT_FILE_MODULE/FLAT_  
FILE_1'
```

See Also

```
OMBCREATE MDL_ACTION_PLAN, OMBDROP MDL_ACTION_PLAN,  
OMBRETRIEVE MDL_ACTION_PLAN, OMUEXPORT MDL_FILE
```

OMBALTER MINING_MODEL

Purpose

Alter the content of a mining model.

Prerequisites

The current context of scripting must be an Oracle Module.

Syntax

```

alterMiningModelCommand = OMBALTER MINING_MODEL "miningModelName"
    "alterMiningModelDetailClause"
miningModelName = "QUOTED_STRING"
alterMiningModelDetailClause = "renameClause" [
    "setMiningModelRefsAndMapPropertiesClause" ] {
    "alterOperatorOwnerDescendantsClause" } |
    "setMiningModelRefsAndMapPropertiesClause" {
    "alterOperatorOwnerDescendantsClause" } |
    "alterOperatorOwnerDescendantsClause"+
renameClause = RENAME TO "QUOTED_STRING"
setMiningModelRefsAndMapPropertiesClause = ( SET ( REF | REFERENCE )
    "caseIdOrTargetColumnClause" ) |
    "setMiningModelAndMapPropertiesClause"
alterOperatorOwnerDescendantsClause = ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) | MODIFY ( "modifyOperatorClause" |
    "modifyGroupClause" | "modifyAttributeClause" | "modifyChildClause" )
| DELETE ( "operatorBottomUpLocator" | "groupBottomUpLocator" |
    "attributeBottomUpLocator" | "childBottomUpLocator" |
    "deleteConnectionLocator" )
caseIdOrTargetColumnClause = CASE_ID_COLUMN "QUOTED_STRING" |
    TARGET_COLUMN "QUOTED_STRING"
setMiningModelAndMapPropertiesClause = ( SET "setPropertiesClause" |
    "setMiningMapPropertiesClause" )
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
modifyOperatorClause = "operatorBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyGroupClause = "groupBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyAttributeClause = "attributeBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyChildClause = "childBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )

```

```

operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
childBottomUpLocator = "childType" "childName" { OF "childType"
    "childName" } [ OF "mappableBottomUpLocator" ]
deleteConnectionLocator = CONNECTION ( FROM "mappableBottomUpLocator" [ TO
    "mappableBottomUpLocator" ] | TO "mappableBottomUpLocator" )
setPropertyClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setMiningMapPropertiesClause = SET MINING_BUILD_MAP PROPERTIES
    "propertyKeyList" VALUES "propertyValueList"
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
attributeName = "QUOTED_STRING"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { ", " IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKeyList = "( " "propertyKey" { ", " "propertyKey" } )"
propertyValueList = "( " "propertyValue" { ", " "propertyValue" } )"
pluggableMapName = "QUOTED_STRING"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
attributeNameList = "( " "attributeName" { ", " "attributeName" } )"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
bindableType = PLUGGABLE_MAPPING | MINING_MODEL | OBJECT_TYPE |
    "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

`alterMiningModelCommand`

Alter the content of a data mining model, including the mapping that builds it

`renameClause`

Rename a mapping, mapping operator, mapping group, or mapping attribute.

alterOperatorOwnerDescendantsClause

Alter the desired child objects applicable to a mapping or a pluggable mapping.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating a child object under a mapping (which is not an operator)

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

In the second example, when user forgets to type "OPERATOR" "GROUP" "ATTRIBUTE" key word, instead of complaining the keywords are missing, OMBPlus will complain about error getting child objects. Here is an example:

```
OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'
```

```
OMB02932: Error getting child objects of type TABLE in M1
```

TO A USER: it looks like OMBPlus should complain they forget to type a keyword.

TO OMBPLUS: the syntax is actually for creating a non-operator child object under the mapping. Therefore, it goes and tries to find type definition for non-operator child object "TABLE" and cannot find it. Therefore the exception is thrown.

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

`addChildClause`

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

`addConnectionClause`

Add connections between mapping groups or mapping attributes.

`modifyOperatorClause`

Modify a mapping operator.

`modifyGroupClause`

Modify a mapping group.

`modifyAttributeClause`

Modify a mapping attribute.

`modifyChildClause`

Modify a child that belongs to a mapping, mapping operator, mapping group or mapping attribute.

`operatorBottomUpLocator`

Location of a mapping operator.

`groupBottomUpLocator`

Location of a mapping group.

`attributeBottomUpLocator`

Location of a mapping attribute.

`childBottomUpLocator`

Location of the child that belongs to a map, mapping operator, mapping group or mapping attribute.

`deleteConnectionLocator`

Delete connections between mapping operators, mapping groups or mapping attributes.

setPropertyClause

Describe the keys of properties for the map or objects in the map.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CUBE, DATA_GENERATOR,

DEDUPLICATOR, DIMENSION, EXPRESSION, EXTERNAL_PROCESS, EXTERNAL_TABLE,

FILTER, FLAT_FILE, INPUT_PARAMETER, JOINER, KEY_LOOKUP, LCRCAST,

LCRSPLITTER, MATCHMERGE, MATERIALIZED_VIEW, NAME_AND_ADDRESS,

OUTPUT_PARAMETER, PIVOT, POSTMAPPING_PROCESS, PREMAPPING_PROCESS, SEQUENCE,

SET_OPERATION, SORTER, SPLITTER, TABLE, TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

attributeName

Name of a mapping attribute.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributesBottomUpLocator

Location of a list of mapping attributes.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

pluggableMapName

Name of the pluggable map.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0
Length of the attribute.

Name: PRECISION
Type: NUMBER
Valid Values: N/A
Default: 0
Precision of the attribute.

Name: SCALE
Type: NUMBER
Valid Values: N/A
Default: 0
Scale of the attribute.

Properties for MINING_MODEL:

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Name: SETTINGS_TABLE_NAME
Type: STRING
Valid Values: N/A
Default: "
Name of table which stores the settings for model build.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue
A single property value. It can be a number, float, boolean or single-quoted string.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBALTER MINING_MODEL 'MODEL1' RENAME TO 'MODEL2'
```

```
OMBALTER MINING_MODEL 'MODEL1' SET REF CASE_ID_COLUMN 'C1'
```

```
OMBALTER MINING_MODEL 'MODEL1' SET REF TARGET_COLUMN 'C2'
```

```
OMBALTER MINING_MODEL 'MODEL1'
```

See Also

OMBALTER, OMBCREATE MINING_MODEL, OMBRETRIEVE MINING_MODEL, OMBDROP MINING_MODEL

OMBALTER NESTED_TABLE

Purpose

Alter the Nested Table by resetting its properties.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
alterNestedTableCommand = OMBALTER ( NESTED_TABLE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

renameClause

renames a table with a different name.

setPropertiesClause

Basic properties for NESTED_TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Nested Table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Nested Table

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Base Element of the Nested Table

Properties for NESTED_TABLE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER NESTED_TABLE 'SOME_NESTED_TABLE' SET PROPERTIES  
(DESCRIPTION)
```

```
VALUES ('This is a new description.')
```

This will set its description to "This is a new description."

See Also

OMBALTER, OMBCREATE NESTED_TABLE, OMBDROP NESTED_TABLE

OMBALTER OBJECT_TYPE

Purpose

Alter the Object Type by resetting its properties or adding/removing its attributes.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
alterObjectTypeCommand = OMBALTER ( OBJECT_TYPE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] [
        "alterAttributesClause" ] | "alterPropertiesOrIconSetClause" [
            "alterAttributesClause" ] | "alterAttributesClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertyClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
alterAttributesClause = ( "addAttributeClause" | "deleteAttributeClause" |
    "modifyAttributeClause" )+
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
addAttributeClause = ADD OBJECT_TYPE_ATTRIBUTE "QUOTED_STRING" [ AT
    POSITION "INTEGER_LITERAL" ] [ SET "setPropertyClause" ]
deleteAttributeClause = DELETE OBJECT_TYPE_ATTRIBUTE "QUOTED_STRING"
modifyAttributeClause = MODIFY OBJECT_TYPE_ATTRIBUTE "QUOTED_STRING" (
    "renameClause" [ "moveToAttributeClause" ] [ SET "setPropertyClause"
        ] | "moveToAttributeClause" [ SET "setPropertyClause" ] | SET
        "setPropertyClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
moveToAttributeClause = MOVE TO POSITION "INTEGER_LITERAL"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterObjectTypeCommand

Alters an Object Type of the given name by either renaming it, or by setting it's properties or by modifying one or more of its Attributes or a combination of these.

renameClause

renames a table with a different name.

alterAttributesClause

Adds, deletes or modifies one or more Attributes of this Object Type.

setPropertyClause

Sets properties (core, logical, physical, user-defined) for Object Type or its Attributes.

Basic properties for OBJECT_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Object Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Object Type

Basic properties for OBJECT_TYPE_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Properties for OBJECT_TYPE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addAttributeClause

Adds an Attribute with the given name and properties.

deleteAttributeClause

Deletes an Attribute with the given name.

modifyAttributeClause

Modifies an Attribute with the given name by either renaming it or changing

its properties or both.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER OBJECT_TYPE 'SOME_OBJECT_TYPE' SET PROPERTIES  
(DESCRIPTION) VALUES
```

```
('This is will be used as Payload type for an AQ.') DELETE
```

```
OBJECT_TYPE_ATTRIBUTE 'ATTR' ADD OBJECT_TYPE_ATTRIBUTE 'ATTR1' SET  
PROPERTIES (DATATYPE) VALUES ('NUMBER')
```

This will set its description to "This is will be used as Payload type for an AQ.", remove attribute 'ATTR' and add an attribute "ATTR1" of Number type.

See Also

```
OMBALTER, OMBRETRIEVE OBJECT_TYPE, OMBCREATE OBJECT_TYPE,  
OMBDROP OBJECT_TYPE
```

OMBALTER ORACLE_MODULE

Purpose

Alter the Oracle module by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of project.

Syntax

```
alterOracleModuleCommand = OMBALTER ( ORACLE_MODULE "QUOTED_STRING" (
    "renameClause" [
        "alterPropertiesOrReferenceClauseForDataMetadataModule" ] |
        "alterPropertiesOrReferenceClauseForDataMetadataModule" |
        "addOrRemoveOrModifyModuleReferenceLocationClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataMetadataModule = ( ( SET ( (
    "alterPropertiesClause" [ ( SET
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) ) | (
        "setReferenceClauseForDataMetadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadataModule" ] ) ) ) | ( UNSET
        "unsetReferenceClauseForDataMetadataModule" [ SET
        "setReferenceClauseForDataMetadataModule" ] ) ) [
        "addOrRemoveOrModifyModuleReferenceLocationClause" ]
addOrRemoveOrModifyModuleReferenceLocationClause = (
    "addReferenceLocationClause" | "removeReferenceLocationClause" |
    "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
    "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
alterPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
unsetReferenceClauseForDataMetadataModule = (
    "unsetReferenceLocationClause" [ UNSET
    "unsetReferenceMetadataLocationOrIconSetClause" ] |
    "unsetReferenceMetadataLocationOrIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" SET AS DEFAULT
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceMetadataLocationOrIconSetClause = (
    "unsetReferenceMetadataLocationClause" [ UNSET
    "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
```

```

propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
  "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceMetadataLocationClause = ( REFERENCE | REF )
  METADATA_LOCATION "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET

```

Keywords And Parameters

alterOracleModuleCommand

This command modifies an existing Oracle module.

QUOTED_STRING

Name of the existing Oracle module in single quotes.

renameClause

Rename an Oracle module.

alterPropertiesOrReferenceClauseForDataMetadataModule

Alter existing Oracle module's properties and/or locations and/or icon sets.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the Oracle module.

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the Oracle module.

unsetReferenceClauseForDataMetadataModule

Unset location and/or icon set for the Oracle module.

addReferenceLocationClause

Add a runtime location to the Oracle module.

removeReferenceLocationClause

Remove a runtime location from the Oracle module.

modifyReferenceLocationClause

Modify a runtime location of the Oracle module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a location to the existing Oracle module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the Oracle module.

unsetReferenceLocationClause

Unset a location to the existing Oracle module.

unsetReferenceMetadataLocationOrIconSetClause

Unset metadata location and/or icon set for the Oracle module.

propertyValue

Value of a property.

setReferenceMetadataLocationClause

Set metadata location for the Oracle module.

setReferenceIconSetClause

Set icon set for the Oracle module.

unsetReferenceMetadataLocationClause

Unset metadata location for the Oracle module.

unsetReferenceIconSetClause

Unset icon set for the Oracle module.

Examples

```
OMBALTER ORACLE_MODULE 'src_module' RENAME TO 'tgt_module' SET
PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a target module.',
```


'target module')

This will rename the Oracle module "src_module" to "tgt_module", and set its description to "This becomes a target module.", set its business name to "target module".

See Also

OMBALTER, OMBCREATE ORACLE_MODULE, OMBDROP ORACLE_MODULE

OMBALTER PACKAGE

Purpose

Alter the Package by renaming it, and/or reset its properties, and/or adding/deleting/modifying the user types.

Prerequisites

Should be in the context of a Oracle Module or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
alterPackageCommand = OMBALTER ( PACKAGE "QUOTED_STRING" ( "renameClause"
    [ "alterPropertiesOrIconSetClause" ] [
    "alterRelationalDependentsClause" ] | "alterPropertiesOrIconSetClause"
    [ "alterRelationalDependentsClause" ] |
    "alterRelationalDependentsClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterRelationalDependentsClause = ( ADD "addRelationalDependentClause" |
    DELETE "deleteRelationalDependentClause" ) [
    "alterRelationalDependentsClause" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
deleteRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`alterPackageCommand`

This command modifies an existing Package.

`QUOTED_STRING`

Name of the existing Package in single quotes.

`renameClause`

Rename a Package.

`alterRelationalDependentsClause`

This clause adds or deletes referential dependencies to other relational objects.

`setPropertyClause`

Used to set properties (core, user-defined) for packages. Valid properties are as shown:

Basic properties for PACKAGE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the package

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the package

Name: PACKAGE_BODY

Type: STRING

Valid Values: N/A

Default: "

Sets the Package Body for a Imported Package

Properties for PACKAGE:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the package with selected AUTHID option. Function will be executed

with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

deleteRelationalDependentClause

This clause deletes referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBALTER PACKAGE 'pkg' RENAME TO 'package_1' SET PROPERTIES  
(DESCRIPTION,
```

```
BUSINESS_NAME) VALUES ('This becomes a package_1', 'package_1')
```

This will rename the Package "pkg" to "package_1", and set its description to "This becomes a package_1", set its business name to "package_1"

See Also

OMBALTER, OMBCREATE PACKAGE, OMBDROP PACKAGE

OMBALTER PLSQL_RECORD_TYPE

Purpose

Alter the PLSQL Record Type by resetting its properties or adding/removing its attributes.

Prerequisites

Should be in the context of a Package

Syntax

```
alterPlSqlRecordTypeCommand = OMBALTER ( PLSQL_RECORD_TYPE "QUOTED_STRING"
    ( "renameClause" [ "alterPropertiesOrIconSetClause" ] [
        "alterRecordTypeAttributeClause" ] | "alterPropertiesOrIconSetClause"
    [ "alterRecordTypeAttributeClause" ] |
        "alterRecordTypeAttributeClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
alterRecordTypeAttributeClause = ( "addRecordTypeAttributeClause" |
    "deleteRecordTypeAttributeClause" | "modifyRecordTypeAttributeClause"
    )+
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
addRecordTypeAttributeClause = ADD ATTRIBUTE "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
deleteRecordTypeAttributeClause = DELETE ATTRIBUTE "QUOTED_STRING"
modifyRecordTypeAttributeClause = MODIFY ATTRIBUTE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterPlSqlRecordTypeCommand

Alters an PLSQL Record Type of the given name by either renaming it, or by setting it's properties or by modifying one or more of its Attributes or a combination of these.

renameClause

renames a table with a different name.

alterRecordTypeAttributeClause

Adds, deletes or modifies one or more Attributes of this PLSQL Record Type.

setPropertyClause

Sets properties (core, logical, physical, user-defined) for PLSQL Record Type or its Attributes.

Basic properties for PLSQL_RECORD_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the PLSQL Record Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the PLSQL Record Type

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,
INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,
NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,
TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,
VARHCHAR, VARCHAR2, XMLTYPE
Default: "
Datatype of the Attribute

Properties for PLSQL_RECORD_TYPE:

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE
Type: STRING
Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL
Default: NA_ADDRTYPE_NORMAL
You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE
Type: STRING
Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,
PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,
START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address

component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name

of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be

used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15, NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2, NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,

NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,

NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,

NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2, NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,

NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,

NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA, NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,

NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,

NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,

NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
 NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
 POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
 PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
 ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
 STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
 STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =||, || =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addRecordTypeAttributeClause

Adds an Attribute with the given name and properties.

deleteRecordTypeAttributeClause

Deletes an Attribute with the given name.

modifyRecordTypeAttributeClause

Modifies an Attribute with the given name by either renaming it or changing its properties or both.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER PLSQL_RECORD_TYPE 'SOME_PLSQL_RECORD_TYPE' SET  
PROPERTIES
```

```
(DESCRIPTION) VALUES ('This is will be used as a return type of ref-cursor  
type.') DELETE ATTRIBUTE 'ATTR' ADD ATTRIBUTE 'ATTR1' SET PROPERTIES  
(DATATYPE) VALUES ('NUMBER')
```

This will set its description to "This is will be used as a return type of ref-cursor type.", remove attribute 'ATTR' and add an attribute "ATTR1" of Number type.

See Also

ALTER, OMBRETRIEVE PLSQL_RECORD_TYPE, OMBCREATE PLSQL_RECORD_TYPE, OMBDROP PLSQL_RECORD_TYPE

OMBALTER PLSQL_REF_CURSOR_TYPE

Purpose

Alter the Ref-Cursor Type by resetting its properties.

Prerequisites

Should be in the context of a Package.

Syntax

```
alterPlSqlRefCursorTypeCommand = OMBALTER ( PLSQL_REF_CURSOR_TYPE
      "QUOTED_STRING" ( "renameClause" [ SET "setPropertiesClause" ] | SET
        "setPropertiesClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
  "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterPlSqlRefCursorTypeCommand

Alters a PL/SQL Ref-cursor Type of the given name by setting it's properties.

renameClause

renames a table with a different name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Ref-cursor Type.

Basic properties for PLSQL_REF_CURSOR_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Ref-Cursor Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Ref-Cursor Type

Name: RETURN_TYPE

Type: STRING(20)

Valid Values: N/A

Default: "

Return type of the Ref-Cursor Type. This should be a PLSQL Record Type.

Properties for PLSQL_REF_CURSOR_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for

skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this

attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAM, NA_COUNTYNAM, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15, NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2, NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,

NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,

NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,

NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2, NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,

NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,

NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA, NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,

NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,

NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,

NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,

NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | | =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER PLSQL_REF_CURSOR_TYPE 'SOME_REF_CURSOR_TYPE' SET  
PROPERTIES
```

```
(DESCRIPTION,RETURN_TYPE) VALUES ('This is a new  
description','MY_MODULE.MY_PACKAGE.MY_RECORDTYPE')
```

This will alter the description and the return type of the Ref-cursor type
"SOME_PLSQL_REF_CURSOR_TYPE".

See Also

```
ALTER, OMBRETRIEVE PLSQL_REF_CURSOR_TYPE, OMBCREATE PLSQL_REF_  
CURSOR_TYPE, OMBDROP PLSQL_REF_CURSOR_TYPE
```

OMBALTER PLSQL_TABLE_TYPE

Purpose

Alter the Table Type by resetting its properties.

Prerequisites

Should be in the context of a Package.

Syntax

```
alterPlSqlTableTypeCommand = OMBALTER ( PLSQL_TABLE_TYPE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterPlSqlTableTypeCommand

Alters a PL/SQL Table Type of the given name by setting its properties.

renameClause

renames a table with a different name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Table Type.

Basic properties for PLSQL_TABLE_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Table Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Table Type

Name: RETURN_TYPE

Type: STRING(20)

Valid Values: NUMBER, VARCHAR2, VARCHAR, DATE, FLOAT

Default: "

Return type of the Table Type. This can be a scalar type or a PLSQL Record Type.

Properties for PLSQL_TABLE_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history

logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,

NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier

indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,
NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,
NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED,
NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,
NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,
NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,
NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,
NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15, NA_EXTRA_16,
NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2, NA_EXTRA_20, NA_EXTRA_3,
NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,
NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,
NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,
NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,
NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2, NA_LASTNAME,
NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,
NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA, NA_NAMEDSIGNATOR,
NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD, NA_NONAMBIGUOUSMATCH,
NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG, NA_NP_UNIT_NBR, NA_OTHERPOSTNAME,
NA_PARSESTATUS, NA_PARSESTATUSDESC, NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON,
NA_PERSONCOUNT, NA_PHONE,

NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | | =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
aalterPLSqlTableTypeCommandExampleTag??
```

See Also

ALTER, OMBRETRIEVE PLSQL_TABLE_TYPE, OMBCREATE PLSQL_TABLE_TYPE, OMBDROP PLSQL_TABLE_TYPE

OMBALTER PLUGGABLE_MAPPING

Purpose

Alter the content of a pluggable mapping.

Prerequisites

1. The current context of scripting must be a project or pluggable map folder.
2. No concurrent user should be modifying the pluggable mapping.

Syntax

```

alterPluggableMappingCommand = OMBALTER PLUGGABLE_MAPPING
    "pluggableMapName" "alterOperatorOwnerDetailClause"
pluggableMapName = "QUOTED_STRING"
alterOperatorOwnerDetailClause = "renameClause" [
    "alterPropertiesOrIconSetClause" ] [
    "alterOperatorOwnerDescendantsClause"+ ] |
    "alterPropertiesOrIconSetClause" [
    "alterOperatorOwnerDescendantsClause"+ ] |
    "alterOperatorOwnerDescendantsClause"+
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] )
    | "setReferenceIconSetClause" ) ) | UNSET
    "unsetReferenceIconSetClause"
alterOperatorOwnerDescendantsClause = ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) | MODIFY ( "modifyOperatorClause" |
    "modifyGroupClause" | "modifyAttributeClause" | "modifyChildClause" )
    | DELETE ( "operatorBottomUpLocator" | "groupBottomUpLocator" |
    "attributeBottomUpLocator" | "childBottomUpLocator" |
    "deleteConnectionLocator" )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
modifyOperatorClause = "operatorBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyGroupClause = "groupBottomUpLocator" ( "renameClause" | SET

```

```

    "setPropertyClause" )
modifyAttributeClause = "attributeBottomUpLocator" ( "renameClause" | SET
    "setPropertyClause" )
modifyChildClause = "childBottomUpLocator" ( "renameClause" | SET
    "setPropertyClause" )
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
childBottomUpLocator = "childType" "childName" { OF "childType"
    "childName" } [ OF "mappableBottomUpLocator" ]
deleteConnectionLocator = CONNECTION ( FROM "mappableBottomUpLocator" [ TO
    "mappableBottomUpLocator" ] | TO "mappableBottomUpLocator" )
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
attributeName = "QUOTED_STRING"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { "," IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
attributeNameList = "(" "attributeName" { "," "attributeName" } ")"
bindableType = PLUGGABLE_MAPPING | OBJECT_TYPE | "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

alterPluggableMappingCommand

Alter the content of a pluggable mapping.

pluggableMapName

Name of the pluggable map.

alterOperatorOwnerDetailClause

Alter the detail of the pluggable mapping.

renameClause

Rename a mapping, mapping operator, mapping group, or mapping attribute.

alterOperatorOwnerDescendantsClause

Alter the desired child objects applicable to a mapping or a pluggable mapping.

setPropertyClause

Describe the keys of properties for the map or objects in the map.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

addChildClause

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

The following is an example for creating a child object under a mapping

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

Note: Key word "OPERATOR" "GROUP" "ATTRIBUTE" are important for their respective ADD clauses. Without the key words, OMBPlus will interpret the ADD clause as an addChildClause. Here is an example:

OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'

OMB02932: Error getting child objects of type TABLE in M1

OMBPLUS interprets the ADD clause as one for creating a non-operator child object under the mapping object. Therefore, it tries to find type definition for non-operator child object "TABLE" and cannot find it.

addConnectionClause

Add connections between mapping groups or mapping attributes.

modifyOperatorClause

Modify a mapping operator.

modifyGroupClause

Modify a mapping group.

modifyAttributeClause

Modify a mapping attribute.

modifyChildClause

Modify a child that belongs to a mapping, mapping operator, mapping group or mapping attribute.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

childBottomUpLocator

Location of the child that belongs to a map, mapping operator, mapping group or mapping attribute.

deleteConnectionLocator

Delete connections between mapping operators, mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCAST, LCRSPLITTER, MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION, TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

attributeName

Name of a mapping attribute.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributesBottomUpLocator

Location of a list of mapping attributes.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Length of the attribute.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Precision of the attribute.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for PLUGGABLE_MAPPING:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref

cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC,
VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero,
based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,

NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1,
NA_EXTRA_10,
NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,
NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_
EXTRA_7,
NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_
CODE,
NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_
FIRSTNAMESTD,
NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_
INSTALLATIONTYPE,
NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_
ISGOODGROUP,
NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME,
NA_LOCALITY_2,
NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_
ORDER, NA_MCD,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_
MIDDLENAME3,
NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_
NEIGHBORHOOD,
NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_
DESIG,
NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_
PARSESTATUSDESC,
NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
NA_PHONE,
NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
POSTALCODEFORMATTED,
NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
PRENAME,
NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
ROUTENAME,
NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
STREETCORRECTED,
NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
STREETNUMBERMATCH,
NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,

NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | |=

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder

determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a

staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING
Valid Values: N/A
Default: "
Exceptions Table Name

Name: EXTRACTION_HINT
Type: STRING
Valid Values: N/A
Default: "
Hint used when extracting from this table using SQL

Name: LOADING_HINT
Type: STRING
Valid Values: N/A
Default: "
Hint used when loading into this table using SQL

Name: PARTITION_NAME
Type: STRING
Valid Values: N/A
Default: "
Partition Name

Name: PEL_ENABLED
Type: BOOLEAN
Valid Values: true, false
Default: false
PEL Enabled

Name: PRIMARY_SOURCE
Type: STRING
Valid Values: NO, YES
Default: NO
A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER
Valid Values: >= 0
Default: 0
Number of records to skip

Name: REPLACE_DATA
Type: BOOLEAN
Valid Values: true, false
Default: false
Replace existing data in target partition if there is any.

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: SINGLEROW
Type: BOOLEAN
Valid Values: true, false
Default: false
Singlerow

Name: SORTED_INDEXES_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
Sorted Indexes Clause

Name: SUBPARTITION_NAME
Type: STRING
Valid Values: N/A
Default: "
Subpartition Name

Name: TARGET_LOAD_ORDER
Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to
lookup a value in the bound table. If the condition is not met, the
default value expression will be returned. If a default expression is not
defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORD_CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TARGET_FILTER_FOR_DELETE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER
Type: STRING(65535)
Valid Values: N/A
Default: "
The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE
Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN,
NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP,
NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL,
NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_
KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD,
NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS,
NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK
Type: STRING(128)
Valid Values: N/A
Default: "
The database link used to access this entity during mapping.

Name: DB_LOCATION
Type: STRING
Valid Values: N/A
Default: "
The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a
staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this
operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name,
so all errors generated by a particular input record will be rolled up into
a single record with the error names concatenated in the error name
attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into

a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW
Type: BOOLEAN
Valid Values: true, false
Default: false
Singlerow

Name: SORTED_INDEXES_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
Sorted Indexes Clause

Name: SUBPARTITION_NAME
Type: STRING
Valid Values: N/A
Default: "
Subpartition Name

Name: TARGET_FILTER_FOR_DELETE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER
Type: STRING(65535)
Valid Values: N/A
Default: "
The Target Load Order property enables you to determine the order in which

multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBALTER PLUGGABLE_MAP 'PLUGGABLE_MAP1' RENAME TO 'PLUGGABLE_MAP2'
```

```
OMBALTER PLUGGABLE_MAP 'PLUGGABLE_MAP1'  
ADD CONNECTION FROM GROUP 'INOUTGRP1' OF OPERATOR 'CUST_SRC'  
TO GROUP 'INOUTGRP1' OF OPERATOR 'CUST_LOOK_UP'
```

```
OMBALTER PLUGGABLE_MAP 'MAP1' DELETE OPERATOR 'CUST_SRC'
```

See Also

OMBALTER, OMBCREATE PLUGGABLE_MAPPING, OMBRETRIEVE PLUGGABLE_MAPPING, OMBDROP PLUGGABLE_MAPPING

OMBALTER PLUGGABLE_MAPPING_FOLDER

Purpose

Alter the content of a pluggable map folder.

Prerequisites

1. The current context of scripting must be a project.
2. No concurrent user should be modifying the pluggable map folder.

Syntax

```
alterPluggableMappingFolderCommand = ( OMBALTER PLUGGABLE_MAPPING_FOLDER
    "pluggableMapFolderName" ( ( "renameClause" [
        "alterPropertiesOrIconSetClause" ] ) |
    "alterPropertiesOrIconSetClause" ) )
pluggableMapFolderName = "QUOTED_STRING"
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] )
    | "setReferenceIconSetClause" ) ) | UNSET
    "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterPluggableMappingFolderCommand

Alter the content of a pluggable map folder.

pluggableMapFolderName

Name of the pluggable map folder.

renameClause

Rename a mapping, mapping operator, mapping group, or mapping attribute.

setPropertiesClause

Describe the keys of properties for the map or objects in the map.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

Examples

```
OMBALTER PLUGGABLE_MAP_FOLDER 'PLUGGABLE_MAP_FOLDER1'
RENAME TO
'PLUGGABLE_MAP_FOLDER2'
```

```
OMBALTER PLUGGABLE_MAP_FOLDER 'PLUGGABLE_MAP_FOLDER1'
SET PROPERTIES (BUSINESS_NAME) VALUES ('Pluggable Map Folder')
```

See Also

OMBALTER

OMBALTER PRESENTATION_TEMPLATE

Purpose

Alters a presentation template.

Prerequisites

Should be in the context of a business presentation module or use the full path.

Syntax

```

alterReportCommand = ( OMBALTER PRESENTATION_TEMPLATE "QUOTED_STRING" ( (
    "renameClause" [ SET "setpropertiesClauseDelayed" ] [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterReportClauses" } ) | ( SET "setpropertiesClauseDelayed" [ SET
    "setReferenceIconSetClause" ] [ UNSET "unsetReferenceIconSetClause" ]
    { "alterReportClauses" } ) | ( SET "setReferenceIconSetClause" [ UNSET
    "unsetReferenceIconSetClause" ] { "alterReportClauses" } ) | ( UNSET
    "unsetReferenceIconSetClause" { "alterReportClauses" } ) | (
    "alterReportClauses" { "alterReportClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setpropertiesClauseDelayed = PROPERTIES "(" "propertyNameListVector" ")"
VALUES "(" "propertyValueListVector" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterReportClauses = ADD "addReportItemClauseForAlter" | MODIFY
    "modifyReportItemClause" | DELETE "deleteReportItemClause"
propertyNameListVector = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueListVector = "propertyValue" { "," "propertyValue" }
addReportItemClauseForAlter = ( DATA_ITEM "QUOTED_STRING" [ SET
    "setPropertiesClause" ] [ SET ( REF | REFERENCE )
    "ReportMeasureReferencesClause" ] ) | ( EDGE_ITEM "QUOTED_STRING" [
    SET "setPropertiesClause" ] [ SET ( REF | REFERENCE )
    "ReportEdgeReferencesClause" ] )
modifyReportItemClause = ( DATA_ITEM "QUOTED_STRING" [ "renameClause" ] [
    SET "setPropertiesClause" ] [ SET ( REF | REFERENCE )
    "ReportMeasureReferencesClause" ] [ UNSET ( REF | REFERENCE )
    "unsetReportMeasureReferencesClause" ] ) | ( EDGE_ITEM "QUOTED_STRING"
    [ "renameClause" ] [ SET "setPropertiesClause" ] [ SET ( REF |
    REFERENCE ) "ReportEdgeReferencesClause" ] [ UNSET ( REF | REFERENCE )
    "unsetReportEdgeReferencesClause" ] )
deleteReportItemClause = DATA_ITEM "QUOTED_STRING" | EDGE_ITEM
    "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
ReportMeasureReferencesClause = MEASURE "QUOTED_STRING" OF CUBE
    "QUOTED_STRING"
ReportEdgeReferencesClause = [ ROLE "QUOTED_STRING" OF ] DIMENSION
    "QUOTED_STRING"
unsetReportMeasureReferencesClause = MEASURE
unsetReportEdgeReferencesClause = ROLE | DIMENSION
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }

```

Keywords And Parameters

alterReportCommand

This clause alters a presentation template.

QUOTED_STRING

name of the presentation template.

renameClause

Renames a presentation template with a different name.

setPropertyClauseDelayed

This clause sets the properties.

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterReportClauses

This clause modifies a presentation template.

propertyNameListVector

This clause holds the names of the properties.

propertyValueListVector

This clause holds the values of the properties.

addReportItemClauseForAlter

This adds an item reference to a presentation template.

QUOTED_STRING

name of the item.

modifyReportItemClause

This modifies an item reference in a presentation template.

DATA_ITEM

This modifies a data item in a presentation template.

QUOTED_STRING

name of the item.

EDGE_ITEM

This modifies an edge item in a presentation template.

deleteReportItemClause

This removes an item reference from a presentation template.

DATA_ITEM

This deletes a data item from a presentation template.

QUOTED_STRING

name of the item.

EDGE_ITEM

This deletes an edge item from a presentation template.

propertyValue

This is a property value.

setPropertyClause

This clause sets the properties of the object.

Basic properties for presentation template:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the presentation template

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the presentation template

Name: PRESENTATION_TYPE

Type: STRING(40)

Valid Values: CROSSTAB, TABLE or a subtype of graph

Default: "

The type of the presentation template

Basic properties for EDGE_ITEM:

Name: PLACEMENT

Type: STRING(40)

Valid Values: TOP OR SIDE, TOP, SIDE, PAGE

Default: "

The placement of the edge item in the presentation template

Properties for PRESENTATION_TEMPLATE:

Name: CATALOG_FOLDER

Type: STRING

Valid Values: N/A

Default: "

Catalog Folder for deployed BI Beans presentation

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for referenced database objects

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to

create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

ReportMeasureReferencesClause

This clause references a measure from the item.

ReportEdgeReferencesClause

This clause references dimension or roles from the item.

unsetReportMeasureReferencesClause

This clause will remove a reference to a measure.

unsetReportEdgeReferencesClause

This clause will remove a reference to an edge item.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

Examples

```
OMBALTER PRESENTATION_TEMPLATE 'COST' SET PROPERTIES  
(DESCRIPTION) VALUES  
( 'COST' )
```

See Also

```
OMBCREATE PRESENTATION_TEMPLATE, OMBRETRIEVE PRESENTATION_  
TEMPLATE
```

OMBALTER PROCEDURE

Purpose

Alter the Procedure by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
alterProcedureCommand = OMBALTER ( PROCEDURE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] [
    "alterFuncProcParameterSCOClause" ] | "alterPropertiesOrIconSetClause"
    [ "alterFuncProcParameterSCOClause" ] |
    "alterFuncProcParameterSCOClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterFuncProcParameterSCOClause = ( ADD ( "alterFuncProcParameterClause" |
    "addRelationalDependentClause" ) | MODIFY
    "modifyFuncProcParameterClause" | DELETE (
    "deleteFuncProcParameterClause" | "deleteRelationalDependentClause" )
    ) [ "alterFuncProcParameterSCOClause" ]
setPropertiesClause = PROPERTIES (" "propertyNameList" )" VALUES ("
    "propertyValueList" )"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterFuncProcParameterClause = PARAMETER "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ]
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
modifyFuncProcParameterClause = ( PARAMETER "QUOTED_STRING" (
    "renameClause" | "moveToClause" | [ SET "setPropertiesClause" ] ) )
deleteFuncProcParameterClause = ( PARAMETER "QUOTED_STRING" )
deleteRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterProcedureCommand

This command modifies an existing Procedure.

QUOTED_STRING

Name of the existing Procedure in single quotes.

renameClause

Rename a Procedure.

alterFuncProcParameterSCOClause

Modify, delete or add a Parameter for Function/Procedure, or add or delete dependencies to some other relational objects.

setPropertyClause

Used to set properties (core, user-defined) for procedure. Valid properties are as shown:

Basic properties for PROCEDURE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Procedure

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Procedure

Name: IMPLEMENTATION

Type: STRING

Valid Values: N/A

Default: "

Set the code for Procedure which is included global variable declaration and code between BEGIN and END.

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB,
BOOLEAN,

CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL
YEAR TO

MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW,
TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,
VARHRCAR,

VARCHAR2, XMLTYPE

Default: NUMBER

Set the data type for Parameter

Name: IN_OUT

Type: STRING

Valid Values: IN, OUT, INOUT

Default: 'IN'

Set the parameter mode for Parameter

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Set the default value for Parameter

Properties for PROCEDURE:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

`alterFuncProcParameterClause`

This clause alters Parameter of a Procedure.

`addRelationalDependentClause`

This clause adds referential dependencies to other relational objects.

`modifyFuncProcParameterClause`

Modify one or more Parameters to this Function/Procedure.

`deleteFuncProcParameterClause`

Delete one or more Parameters to this Function/Procedure.

`deleteRelationalDependentClause`

This clause deletes referential dependencies to other relational objects.

`propertyNameList`

Comma separated list of property names. Property names are unquoted.

`propertyValueList`

Comma separated list of property values.

`moveToClause`

Move a Parameters of this Function/Procedure.

`propertyValue`

Value of a property.

Examples

```
OMBALTER PROCEDURE 'proc' RENAME TO 'proc_1' SET PROPERTIES
(DESCRIPTION,
```

```
BUSINESS_NAME) VALUES ('This becomes a proc_1', 'proc')
```

This will rename the Procedure "proc" to "proc_1", and set its description to "This becomes a proc_1", set its business name to "proc".

If Packaged Function is overloaded, first find the Signature by using OMBLIST command, and then use OMBALTER command using appropriate signature.

Example, if OMBLIST PROCEDURES gives following two signatures,

```
PROC_1 (NUMBER)
```

```
PROC_1 (VARCHAR2, NUMBER)
```

The OMBALTER Syntax to modify the first one will be as follows

```
OMBALTER PROCEDURE 'PROC_1 \ (NUMBER \)' SET PROPERTIES
(DESCRIPTION,
```

```
BUSINESS_NAME) VALUES ('descri_PROC_1', 'PROC_1')
```

See Also

OMBALTER, OMBCREATE PROCEDURE, OMBDROP PROCEDURE

OMBALTER PROCESS_FLOW

Purpose

Alter the Process Flow by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of a Process Flow Package.

Syntax

```
alterProcessFlowCommand = OMBALTER PROCESS_FLOW "QUOTED_STRING" ( (
    renameClause [ "alterPropertiesOrIconSetClause" ] [
        "alterProcessFlowSCOClauses" ] ) | ( "alterPropertiesOrIconSetClause"
    [ "alterProcessFlowSCOClauses" ] ) | "alterProcessFlowSCOClauses" )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterProcessFlowSCOClauses = ADD ( "addProcessParameterClause" [
    "alterProcessFlowSCOClauses" ] | "addProcessVariableClause" [
    "alterProcessFlowSCOClauses" ] | "addActivityClause" [
    "alterProcessFlowSCOClauses" ] | "addTransitionClause" [
    "alterProcessFlowSCOClauses" ] ) | MODIFY (
    "modifyProcessParameterClause" [ "alterProcessFlowSCOClauses" ] |
    "modifyProcessVariableClause" [ "alterProcessFlowSCOClauses" ] |
    "modifyActivityClause" [ "alterProcessFlowSCOClauses" ] |
    "modifyUserDefinedActivityClause" [ "alterProcessFlowSCOClauses" ] |
    "modifyTransitionClause" [ "alterProcessFlowSCOClauses" ] ) | DELETE (
    "deleteProcessParameterClause" [ "alterProcessFlowSCOClauses" ] |
    "deleteProcessVariableClause" [ "alterProcessFlowSCOClauses" ] |
    "deleteActivityClause" [ "alterProcessFlowSCOClauses" ] |
    "deleteTransitionClause" [ "alterProcessFlowSCOClauses" ] |
    "deleteUserDefinedParameterClause" [ "alterProcessFlowSCOClauses" ] )
setPropertiesClause = PROPERTIES (" "propertyNameList" ") VALUES ("
    "propertyValueList" ")
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addProcessParameterClause = ( PARAMETER "QUOTED_STRING" ) [ SET
    "setPropertiesClause" ]
addProcessVariableClause = ( VARIABLE "QUOTED_STRING" ) [ SET
    "setPropertiesClause" ]
addActivityClause = ( "addStandardActivityClause" | "addMapActivityClause"
    | "addTemplateActivityClause" | "addDataAuditorActivityClause" |
    "addFunctionActivityClause" | "addSubProcessActivityClause" )
addTransitionClause = ( TRANSITION "QUOTED_STRING" ( FROM ACTIVITY
    "QUOTED_STRING" ) ( TO "QUOTED_STRING" ) ) [ SET "setPropertiesClause"
    ]
modifyProcessParameterClause = PARAMETER "QUOTED_STRING" ( "renameClause"
    [ SET "setPropertiesClause" ] | SET "setPropertiesClause" )
modifyProcessVariableClause = VARIABLE "QUOTED_STRING" ( "renameClause" [
    SET "setPropertiesClause" ] | SET "setPropertiesClause" )
modifyActivityClause = ACTIVITY "QUOTED_STRING" ( ( "renameClause" [
    "alterPropertiesOrIconSetClause" ] [ "modifyActivityParameterClause" ]
    ) | ( "alterPropertiesOrIconSetClause" [
    "modifyActivityParameterClause" ] ) | "modifyActivityParameterClause"
    )
modifyUserDefinedActivityClause = ( USER_DEFINED | TRANSFORMATION |
```

```

MAPPING ) ACTIVITY "QUOTED_STRING" ( "renameClause" [
  "alterPropertiesOrIconSetClause" ] [ "modifyUDActivityParameterClause"
  ] | "alterPropertiesOrIconSetClause" [
  "modifyUDActivityParameterClause" ] |
  "modifyUDActivityParameterClause" )
modifyTransitionClause = TRANSITION "QUOTED_STRING" ( "renameClause" [ SET
  "setPropertiesClause" ] | SET "setPropertiesClause" )
deleteProcessParameterClause = PARAMETER "QUOTED_STRING"
deleteProcessVariableClause = VARIABLE "QUOTED_STRING"
deleteActivityClause = ACTIVITY "QUOTED_STRING"
deleteTransitionClause = TRANSITION "QUOTED_STRING"
deleteUserDefinedParameterClause = PARAMETER "QUOTED_STRING" OF
  USER_DEFINED ACTIVITY "QUOTED_STRING"
propertyNameList = ( "UNQUOTED_STRING" | BINDING ) { "," (
  "UNQUOTED_STRING" | BINDING ) }
propertyValueList = "propertyValue" { "," "propertyValue" }
addStandardActivityClause = ( "UNQUOTED_STRING" | USER_DEFINED ) ACTIVITY
  "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
  "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ]
addMapActivityClause = ( MAPPING ACTIVITY "QUOTED_STRING" [ SET (
  PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF | REFERENCE )
  "setPropertiesAndReferencesMapClauses" ] [ "setReferenceIconSetClause"
  ] ) ] | ( REF | REFERENCE ) "setPropertiesAndReferencesMapClauses" [
  SET "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addTemplateActivityClause = ( ACTIVITY_TEMPLATE ACTIVITY "QUOTED_STRING" [
  SET ( PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF |
  REFERENCE ) "setPropertiesAndReferencesTemplateClause" ] [
  "setReferenceIconSetClause" ] ) ] | ( REF | REFERENCE )
  "setPropertiesAndReferencesTemplateClause" [ SET
  "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addDataAuditorActivityClause = ( DATA_AUDITOR ACTIVITY "QUOTED_STRING" [
  SET ( PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF | REFERENCE
  ) "setPropertiesAndReferencesDataAuditorClauses" ] [
  "setReferenceIconSetClause" ] ) ] | ( REF | REFERENCE )
  "setPropertiesAndReferencesDataAuditorClauses" [ SET
  "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addFunctionActivityClause = ( TRANSFORMATION ACTIVITY "QUOTED_STRING" [
  SET ( PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF | REFERENCE
  ) "setPropertiesAndReferencesFunctionClauses" ] [
  "setReferenceIconSetClause" ] ) ] | ( REF | REFERENCE )
  "setPropertiesAndReferencesFunctionClauses" [ SET
  "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addSubProcessActivityClause = ( SUBPROCESS ACTIVITY "QUOTED_STRING" SET (
  PROPERTIES "collectPropertiesClause" SET ( REF | REFERENCE )
  "setPropertiesAndReferencesSubProcessClauses" [ SET
  "setReferenceIconSetClause" ] | ( REF | REFERENCE )
  "setPropertiesAndReferencesSubProcessClauses" [ SET
  "setReferenceIconSetClause" ] ) )
modifyActivityParameterClause = MODIFY "modifyActivityParameter" [
  "modifyActivityParameterClause" ]
modifyUDActivityParameterClause = ADD ( "addUDActivityParameterClause" [
  "modifyUDActivityParameterClause" ] ) | MODIFY (
  "alterUDActivityParameterClause" [ "modifyUDActivityParameterClause" ]
  ) | DELETE ( "deleteUDActivityParameterClause" [
  "modifyUDActivityParameterClause" ] )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
collectPropertiesClause = "(" "propertyNameList" ")" VALUES "("
  "propertyValueList" ")"
setPropertiesAndReferencesMapClauses = ( MAPPING "QUOTED_STRING" )

```

```

setPropertyAndReferencesTemplateClause = ( ACTIVITY_TEMPLATE
    "QUOTED_STRING" )
setPropertyAndReferencesDataAuditorClauses = ( DATA_AUDITOR
    "QUOTED_STRING" )
setPropertyAndReferencesFunctionClauses = ( TRANSFORMATION
    "QUOTED_STRING" )
setPropertyAndReferencesSubProcessClauses = ( PROCESS_FLOW
    "QUOTED_STRING" )
modifyActivityParameter = PARAMETER "QUOTED_STRING" SET
    "setPropertyClause"
addUDActivityParameterClause = PARAMETER "QUOTED_STRING" [ SET
    "setPropertyClause" ]
alterUDActivityParameterClause = PARAMETER "QUOTED_STRING" ( (
    "renameClause" [ SET "setPropertyClause" ] ) | ( SET
    "setPropertyClause" ) )
deleteUDActivityParameterClause = PARAMETER "QUOTED_STRING"

```

Keywords And Parameters

alterProcessFlowCommand

Alter process flow.

renameClause

Rename process flow/ activity, depending on current context.

alterProcessFlowSCOClauses

This clause is wrapper clause to add, modify or delete Process Flow's second class objects.

setPropertyClause

Used to set properties (core, user-defined) for process flow. Note: For MAPPING, TRANSFORMATION and SUBPROCESS activities the setPropertyAndReferencesMapClauses, setPropertyAndReferencesFunctionClauses and setPropertyAndReferencesSubProcessClauses respectively, are mandatory.

For MAPPING or TRANSFORMATION activities and the REFERENCE property has to

be set to a

valid MAP or TRANSFORMATION within the current project.

For SUBPROCESS activities the REFERENCE property has to be set to a SUBPROCESS within the same PROCESS_FLOW_PACKAGE.

Valid properties are as shown:

Base properties for PROCESS_FLOW:

Basic properties for Process Flow, Activity, Transition and Parameter:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Process Flow

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Process Flow Core properties for Transition :

Name: TRANSITION_CONDITION

Type: STRING

Valid Values: ", SUCCESS, ERROR, WARNING

Default: ", that is, Unconditional

Sets the Transition Condition of a Transition

Description of the Process Flow Core properties for Activity Parameter :

Name: DATATYPE

Type: STRING

Valid Values: INTEGER, FLOAT, DATE, STRING, BOOLEAN

Default: STRING

Sets the datatype of a Activity Parameter

Name: DIRECTION

Type: STRING

Valid Values: IN

Default: IN

Sets the direction of a Activity Parameter

Name: VALUE

Type: STRING

Valid Values: Examples '123', '123.456', 'Jan-08-2003', 'I am String',
'true'

Default: "

For Mapping activities representing PLSQL maps, the allowed value for the parameters:

OPERATING_MODE:'SET_BASED' 'ROW_BASED' 'ROW_BASED_TARGET_ONLY'

'SET_BASED_FAIL_OVER_TO_ROW_BASED'

'SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY'

AUDIT_LEVEL:'NONE' 'STATISTICS' 'ERROR_DETAILS' 'COMPLETE'

Sets the value of a Activity Parameter

Name: BINDING

Type: STRING

Valid Values: Examples 'PARAM_1', 'PARAM_2'

Default: "

Represents the parameter on the process flow that this parameter is bound to.

When setting users can specify the name of any PROCESS PARAMETER of same datatype.

This feature allows for parameterizing the process flow. If the parameter is bound

the VALUE property is ignored when generating the process flow.

To unbind a parameter, use an empty quoted string, that is "", and the parameter will be unbound.

Properties for PROCESS_FLOW:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

`addProcessParameterClause`

This clause adds the Parameters for a Process Flow.

`addActivityClause`

This clause adds the Activities for a Process Flow.

`addTransitionClause`

This clause adds the Transitions for a Process Flow.

`modifyProcessParameterClause`

This clause modifies the Parameters of a Process Flow.

`modifyActivityClause`

This clause modifies the Activities of a Process Flow.

`modifyUserDefinedActivityClause`

Alter the properties of the specified user defined activity.

`modifyTransitionClause`

This clause modifies the Transitions of a Process Flow.

`deleteProcessParameterClause`

This clause deletes the Parameters of a Process Flow.

`deleteActivityClause`

This clause deletes the Activities of a Process Flow.

`deleteTransitionClause`

This clause deletes the Transitions of a Process Flow.

`deleteUserDefinedParameterClause`

This clause deletes the Parameters of a User Defined Activity of a Process Flow.

`propertyNameList`

A comma delimited set of property names to set.

`propertyValueList`

A comma delimited set of property values to set.

`addStandardActivityClause`

This clause adds standard activity types AND, EMAIL, END_ERROR, END_WARNING, END_SUCCESS, FILE_EXISTS, FORK, FTP, OR, ASSIGN, END_LOOP, FOR_LOOP, MANUAL, NOTIFICATION, ROUTE, SET_STATUS, SQLPLUS, WAIT, WHILE_LOOP OR USER_DEFINED to a Process Flow.

`addMapActivityClause`

This clause adds the MAP activity to a Process Flow.

`addTemplateActivityClause`

This clause adds an ACTIVITY_TEMPLATE as an activity to a Process Flow.

`addDataAuditorActivityClause`

This clause adds a DATA_AUDITOR activity to a Process Flow.

`addFunctionActivityClause`

This clause adds the Function or Procedure activity to a Process Flow.

`addSubProcessActivityClause`

This clause adds a Process as an activity to a Process Flow.

`modifyActivityParameterClause`

Alter the properties of activity parameters.

modifyUDActivityParameterClause

Alter user defined activity by adding new parameters or modifying existing parameter.

propertyValue

Integer value, float value or quoted string literal.

collectPropertiesClause

This clause collects core properties of Map, Function/Procedure and Subprocess activity.

setPropertiesAndReferencesMapClauses

This clause sets reference to the existing Map.

setPropertiesAndReferencesFunctionClauses

This clause sets a reference to existing Function or Procedure.

setPropertiesAndReferencesSubProcessClauses

This clause sets a reference to existing Process Flow.

modifyActivityParameter

Alter the properties of the parameter.

addUDActivityParameterClause

Add more activity parameters to the user defined activity.

alterUDActivityParameterClause

In the current user defined activity, alter the properties of the activity parameter like rename, set DATATYPE or set VALUE.

deleteUDActivityParameterClause

For the current user defined activity, delete the activity parameter.

Examples

```
OMBALTER PROCESS_FLOW 'process_flow' RENAME TO 'p_flow' SET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a process flow.',  
'process flow')
```


This will rename the Process Flow "process_flow" to "p_flow", and set its description to "This becomes a process flow", set its business name to "process flow".

See Also

OMBALTER, OMBCREATE PROCESS_FLOW, OMBDROP PROCESS_FLOW

OMBALTER PROCESS_FLOW_MODULE

Purpose

Alter the Process Flow Module by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of a project.

Syntax

```
alterProcessFlowModuleCommand = OMBALTER ( PROCESS_FLOW_MODULE
    "QUOTED_STRING" ( "renameClause" [ "alterPropertiesOrReferenceClause"
    ] | "alterPropertiesOrReferenceClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClause = SET ( "setPropertiesClause" [ SET
    "setReferenceClause" [ UNSET "unsetReferenceClause" ] | UNSET
    "unsetReferenceClause" [ SET "setReferenceClause" ] ] |
    "setReferenceClause" [ UNSET "unsetReferenceClause" ] ) | UNSET
    "unsetReferenceClause" [ SET "setReferenceClause" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
unsetReferenceClause = ( "unsetReferenceLocationClause" [ UNSET
    "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterProcessFlowModuleCommand

This command modifies an existing process flow module.

renameClause

Rename an existing process flow module.

setPropertiesClause

Set values of properties of a process flow module.

Base properties for PROCESS_FLOW_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Process Flow Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Process Flow Module

propertyNameList

Comma-delimited list of property names. Property names are not in quotation marks.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a location to a supported workflow engine.

unsetReferenceLocationClause

Unset the location of the process flow module.

propertyValue

Value of a property.

Examples

```
OMBALTER PROCESS_FLOW_MODULE 'process_module' RENAME TO 'p_module'
SET
```

```
PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a process
flow module.', 'process module')
```

This will rename the Process Flow Module "process_module" to "p_module", and set its description to "This becomes a process flow module", set its business name to "process module".

See Also

OMBALTER, OMBCREATE PROCESS_FLOW_MODULE, OMBDROP PROCESS_FLOW_MODULE

OMBALTER PROCESS_FLOW_PACKAGE

Purpose

Alter the Process Flow Package by renaming it, and/or reset its properties.

Prerequisites

Should be in the context of a Process Flow Module.

Syntax

```
alterProcessFlowPackageCommand = OMBALTER ( PROCESS_FLOW_PACKAGE
    "QUOTED_STRING" ( "renameClause" [ "alterPropertiesOrIconSetClause" ]
    | "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = ( "UNQUOTED_STRING" | BINDING ) { "," (
    "UNQUOTED_STRING" | BINDING ) }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterProcessFlowPackageCommand

Alter process flow package.

renameClause

Rename the process flow package.

setPropertiesClause

Set the properties of the process flow package.

Basic properties for PROCESS_FLOW_PACKAGE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Process Flow Package

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Process Flow Package

Properties for PROCESS_FLOW_PACKAGE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "OWF.PACKAGES.DEPLOYABLE:DESCRIPTION"

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

A comma delimited set of property names to set.

propertyValueList

A comma delimited set of property values to set.

propertyValue

Integer value, float value or quoted string literal.

Examples

```
OMBALTER PROCESS_FLOW_PACKAGE 'process_package' RENAME TO 'p_
package' SET
PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a process
flow package.', 'process package')
```

This will rename the Process Flow Package "process_package" to "p_package", and set its description to "This becomes a process flow package", set its business name to "process package".

See Also

OMBALTER, OMBCREATE PROCESS_FLOW_PACKAGE, OMBDROP PROCESS_FLOW_PACKAGE

OMBALTER PROFILE_REFERENCE

Purpose

To alter a profile reference, such as to add or delete a data rule usage, or set properties on a profile attribute.

Prerequisites

Must be done in the context of a Data Profile.

Syntax

```
alterProfileReferenceCommand = OMBALTER ( PROFILE_REFERENCE
    "QUOTED_STRING" ( [ SET "setPropertiesClause" ] { ADD
    "addDataRuleUsageClause" | DELETE "deleteDataRuleUsageClause" | MODIFY
    ( "modifyDataRuleUsageClause" | "modifyProfileAttributeClause" ) } )
)
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
addDataRuleUsageClause = DATA_RULE_USAGE ( [ "QUOTED_STRING" ] [ SET REF
    DATA_RULE "QUOTED_STRING" ] ( DERIVE FROM ( (
    "getProfileAttributeDerivationClause" |
    "getProfileForeignKeyDerivationClause" |
    "getProfileUniqueKeyDerivationClause" |
    "getFunctionalDependencyDerivationClause" |
    "getRowRelationshipDerivationClause" ) [ CREATE IN DATA_RULE_MODULE
    "QUOTED_STRING" ] ) | ( GROUP "QUOTED_STRING" SET REF TABLE
    "QUOTED_STRING" ( ATTRIBUTE "QUOTED_STRING" SET REF COLUMN
    "QUOTED_STRING" )+ )+ [ SET "setPropertiesClause" ] ) )
deleteDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING"
modifyDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | SET
    "setPropertiesClause" )
modifyProfileAttributeClause = PROFILE_ATTRIBUTE "QUOTED_STRING" SET
    "setPropertiesClause"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
getProfileAttributeDerivationClause = PROFILE_ATTRIBUTE "QUOTED_STRING"
    PROFILE_TYPE "QUOTED_STRING"
getProfileForeignKeyDerivationClause = PROFILE_FOREIGN_KEY "QUOTED_STRING"
getProfileUniqueKeyDerivationClause = PROFILE_UNIQUE_KEY "QUOTED_STRING"
getFunctionalDependencyDerivationClause = FUNCTIONAL_DEPENDENCY
    "QUOTED_STRING"
getRowRelationshipDerivationClause = ROW_RELATIONSHIP "QUOTED_STRING"
renameClause = RENAME TO "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterProfileReferenceCommand

This clause alters the profile reference.

QUOTED_STRING

This is the name of the source pointed to by the profile reference.

setPropertiesClause

Configuration properties for PROFILE_REFERENCE that affect loading:

Name: COPY_DATA

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable copying of data from source to profile workspace.

Name: CALCULATE_DATATYPES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data type discovery for the selected table.

Name: CALCULATE_COMMON_FORMATS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable common format discovery for the selected table.

Name: NULL_VALUE

Type: STRING

Valid Values: any string value

Default: null

This value will be considered as the null value when profiling. Please enclose the value in single quotes. An unquoted null (the current default value) will be considered a database null.

Name: SAMPLE_RATE

Type: 100

Valid Values: 1-100

Default: 100

This value will be the percent of total rows that will be randomly selected during loading.

Configuration properties for PROFILE_REFERENCE that affect profiling:

Name: CALCULATE_DOMAINS

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable domain discovery.

Name: DOMAIN_MAX_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

The maximum number of distinct values in a column in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MAX_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

The maximum number of distinct values in a column, expressed as a percentage of the total number of rows in the table, in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MIN_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

The minimum number of rows for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: DOMAIN_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

The minimum number of rows, expressed as a percentage of the total number of rows, for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: CALCULATE_UK

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable unique key discovery.

Name: UK_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a unique key relationship.

Name: CALCULATE_FD

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable functional dependency discovery.

Name: FD_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a functional dependency relationship.

Name: FD_UK_LHS_COUNT

Type: NUMBER

Valid Values: 1-number of attributes of source less 1

Default: 1

This is the maximum number of attributes for unique key and functional dependency profiling.

Name: CALCULATE_FK

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable foreign key discovery.

Name: FK_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a foreign key relationship.

Name: CALCULATE_REDUNDANT_COLUMNS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable redundant column discovery with respect to a foreign key-unique key pair.

Name: REDUNDANT_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that are redundant.

Name: CALCULATE_DATA_RULES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data rule profiling for the selected table.

Configuration properties for PROFILE_ATTRIBUTE that affect loading:

Name: USE_IN_LOADING

Type: BOOLEAN

Valid Values: true | false

Default: true

This tells the profiler if the data for this column is to be copied from the source schema to the profile workspace schema.

Name: CALCULATE_PATTERNS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable pattern discovery.

Name: MAX_NUM_PATTERNS

Type: NUMBER

Valid Values: any number less than the number of rows of the source

Default: 10

This tells the profiler to get the top-N patterns for the attribute.

Configuration properties for PROFILE_ATTRIBUTE that affect loading:

Name: USE_IN_FK

Type: BOOLEAN

Valid Values: true | false

Default: true

This tells the profiler if this column is to be part of the determinant in functional dependency discovery. By default, all non-numeric columns set this property to false

Name: USE_IN_LOADING

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will include this column in profiling.

Name: CALCULATE_DATATYPES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data type discovery for the selected table.

Name: CALCULATE_COMMON_FORMATS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable common format discovery for the selected table.

Name: NULL_VALUE

Type: STRING

Valid Values: any string value

Default: null

This value will be considered as the null value when profiling. Please enclose the value in single quotes. An unquoted null (the current default

value) will be considered a database null.

Configuration properties for PROFILE_ATTRIBUTE that affect profiling:

Name: USE_IN_DETERMINANT

Type: BOOLEAN

Valid Values: true | false

Default: true

This tells the profiler if this column is to be part of the determinant in functional dependency discovery. By default, all non-numeric columns set this property to false

Name: CALCULATE_DOMAINS

Type: BOOLEAN

Valid Values: true | false

Default: true

This tells the profiler if domain values are to be discovered for this column.

Name: DOMAIN_MAX_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

This tells the profiler the maximum number of distinct values this column can have to be considered as domain attributes.

Name: DOMAIN_MAX_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

This tells the profiler the maximum number of distinct values as a percentage this column can have to be considered as domain attributes.

Name: DOMAIN_MIN_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

This tells the profiler the minimum number of rows a particular value should have in order to be considered a domain value.

Name: DOMAIN_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

This tells the profiler the minimum number of rows as a percentage a particular value should have in order to be considered a domain value.

addDataRuleUsageClause

Add a data rule usage to the profile table.

QUOTED_STRING

name of data rule usage.

deleteDataRuleUsageClause

Delete a data rule usage.

QUOTED_STRING

name of data rule usage.

modifyDataRuleUsageClause

Rename or modify the properties of a data rule usage.

QUOTED_STRING

name of data rule usage.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

renameClause

Renames an object.

propertyValue

Value of a property.

Examples

```
OMBALTER PROFILE_REFERENCE 'EMP' ADD DATA_RULE_USAGE SET REF
DATA_RULE
'MIN_EMPNO_RULE' DERIVE FROM PROFILE_ATTRIBUTE 'EMPNO' PROFILE_
TYPE 'MIN'
CREATE IN DATA_RULE_MODULE 'DR_MODULE'
```

getProfileAttributeDerivationClause

PROFILE_TYPE is one of {'MIN','MAX','RANGE','DOMAIN','NULL'}.

See Also

OMBALTER

OMBALTER PROJECT

Purpose

Alter the project by renaming it, and/or reset its properties.

Prerequisites

Should be in the top level context.

Syntax

```
alterProjectCommand = OMBALTER ( PROJECT "QUOTED_STRING" ( "renameClause"
    [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterProjectCommand

Modify an existing project.

QUOTED_STRING

Name of the existing project in quotes.

renameClause

Rename a project.

alterPropertiesOrIconSetClause

Alter existing Project properties and/or Icon Set.

setPropertiesClause

Associate a set of properties with a project.

Basic properties for PROJECT:

Name: BUSINESS_NAME

Type: STRING(200)
Valid Values: N/A
Default: NAME
Business name of a Project

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of a Project

propertyNameList
Comma separated list of property names. Property names are unquoted.

propertyValueList
Comma separated list of property values.

propertyValue
Value of a property.

Examples

```
OMBALTER PROJECT 'New Project' RENAME TO 'Old Project' SET PROPERTIES  
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes an old project.', 'old  
payroll project')
```

This will rename project "New Project" to "Old Project", and set its description to "This becomes an old project", set its business name to "old payroll project".

See Also

OMBALTER, OMBCREATE PROJECT, OMBDROP PROJECT

OMBALTER QUEUE_PROPAGATION

Purpose

Alter the Queue Propagation by resetting its properties.

Prerequisites

Should be in the context of an Advanced Queue. The target Queue should exist in any Oracle Module.

Syntax

```
alterQPCommand = OMBALTER ( QUEUE_PROPAGATION "QUOTED_STRING" (
    "renameClause" [ SET "setQPPPropertiesClause" ] | SET
    "setQPPPropertiesClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setQPPPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterQPCommand

Alters the Queue Propagation with the given name by either renaming it or by setting it's properties or both.

renameClause

Renames the Queue Propagation to the given name.

setQPPPropertiesClause

Sets properties (core, logical, physical, user-defined) for Queue Propagation. Valid properties are as shown:

Basic properties for QUEUE_PROPAGATION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Queue Propagation

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Queue Propagation

Name: TARGET_QUEUE

Type: STRING(4000)

Valid Values: N/A

Default: "

Target Queue for the Queue Propagation. This has to be the name of a Queue existing in any Oracle Module.

Properties for QUEUE_PROPAGATION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DURATION

Type: STRING

Valid Values: N/A

Default: "

The duration of propagation to be done. The default value is null. Applicable only for non-streams queue propagation.

Name: GENERATE_DBLINK

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Database Link which is used for propagation

Name: GENERATE_QUEUE_PROPAGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Queue Propagation

Name: GENERATE_REPLICATION_RULE

Type: BOOLEAN

Valid Values: true, false

Default: false

Generate Ruleset and Rule for Replication purpose in Streams queue propagation

Name: GENERATE_SCHEDULE_PROPAGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Scheduling propagation. Applicable only for non-streams queue propagation.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LATENCY

Type: STRING

Valid Values: N/A

Default: 60

The latency for the queue propagation. By default the value is 60. Applicable only for non-streams queue propagation.

Name: NEXT_TIME

Type: STRING

Valid Values: N/A

Default: "

Next time when the propagation to be done. The default value is null. Applicable only for non-streams queue propagation.

Name: NOT_PERMITTED_TAG_VALUES

Type: STRING

Valid Values: N/A

Default: "

List of comma separated Tag values (in Hex numbers) which are not allowed for propagation

Name: PERMITTED_TAG_VALUES

Type: STRING

Valid Values: N/A

Default: "

List of comma separated Tag values (in Hex numbers) which are allowed for propagation

Name: START_TIME

Type: STRING

Valid Values: N/A

Default: SYSDATE

The start time for the propagation to happen. The default value is SYSDATE. Applicable only for non-streams queue propagation.

Name: SUBSCRIBER_TRANSFORMATION

Type: STRING

Valid Values: N/A

Default: "

A Transformation that will be applied before propagation to the target queue. Applicable only for non-streams queue propagation.

Name: SUBSCRIBER_RULE_CONDITION

Type: STRING

Valid Values: N/A

Default: "

A Rule condition to check whether the message can be propagated to the subscriber. Applicable only for non-streams queue propagation.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER QUEUE_PROPAGATION 'SOME_QUEUE_PROPAGATION' SET
PROPERTIES
(SUBSCRIBER_RULE_CONDITION, SUBSCRIBER_TRANSFORMATION, START_
TIME, DURATION,
NEXT_TIME, LATENCY, PERMITTED_TAG_VALUES, NOT_PERMITTED_TAG_
VALUES,
GENERATE_DBLINK, GENERATE_SCHEDULE_PROPAGATION) VALUES ('x>2',
'my_transform', 'sysdate+2', 40, 'sysdate+4', 4, '5,ee,ff', 'aa,33',
'true', 'false')
```

This will set its properties as specified.

See Also

OMBALTER, OMBCREATE QUEUE_PROPAGATION, OMBRETRIEVE QUEUE_
PROPAGATION, OMBDROP QUEUE_PROPAGATION

OMBALTER QUEUE_TABLE

Purpose

Alter the Queue Table by resetting its properties.

Prerequisites

Should be in the context of an Oracle Module. The target Queue should exist in any Oracle Module.

Syntax

```
alterQTCommand = OMBALTER ( QUEUE_TABLE "QUOTED_STRING" ( "renameClause" [
    SET "setQTPropertiesClause" ] | SET "setQTPropertiesClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setQTPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterQTCommand

Alters the Queue Table with the given name by either renaming it or by setting it's properties or both.

renameClause

Renames the Queue Table to the given name.

setQTPropertiesClause

Sets properties (core, logical, physical, user-defined) for Queue Table.

Valid properties are as shown:

Basic properties for QUEUE_TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Queue Table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Queue Table

Name: PAYLOAD_TYPE

Type: STRING(4000)

Valid Values: N/A

Default: "

Object Type for the Queue Table. This has to be the name of an Object Type (OBJECT_TYPE) existing in any Oracle Module.

Properties for QUEUE_TABLE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: GENERATE_QUEUE_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate code to create the queue table that will persist the messages of this Advanced Queue.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER QUEUE_TABLE 'SOME_QUEUE_TABLE' SET PROPERTIES  
(TABLESPACE,  
GENERATE_QUEUE_TABLE) VALUES ('users', 'true')
```

This will set its properties as specified.

See Also

OMBALTER, OMBCREATE QUEUE_TABLE, OMBRETRIEVE QUEUE_TABLE,
OMBDROP QUEUE_TABLE

OMBALTER REAL_TIME_MAPPING

Purpose

Alter the content of a Real Time mapping.

Prerequisites

1. The current context of scripting must be an Oracle Module
2. No concurrent user should be modifying the mapping

Syntax

```

alterRealTimeMappingCommand = OMBALTER REAL_TIME_MAPPING "mappingName"
    "alterMapDetailClause"
mappingName = "QUOTED_STRING"
alterMapDetailClause = "renameClause" [ "alterPropertiesOrIconSetClause" ]
    [ "alterOperatorOwnerDescendantsClause"+ ] |
    "alterPropertiesOrIconSetClause" [
    "alterOperatorOwnerDescendantsClause"+ ] |
    "alterOperatorOwnerDescendantsClause"+
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] )
    | "setReferenceIconSetClause" ) ) | UNSET
    "unsetReferenceIconSetClause"
alterOperatorOwnerDescendantsClause = ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) | MODIFY ( "modifyOperatorClause" |
    "modifyGroupClause" | "modifyAttributeClause" | "modifyChildClause" )
    | DELETE ( "operatorBottomUpLocator" | "groupBottomUpLocator" |
    "attributeBottomUpLocator" | "childBottomUpLocator" |
    "deleteConnectionLocator" )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
modifyOperatorClause = "operatorBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyGroupClause = "groupBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )
modifyAttributeClause = "attributeBottomUpLocator" ( "renameClause" | SET
    "setPropertiesClause" )

```

```

modifyChildClause = "childBottomUpLocator" ( "renameClause" | SET
    "setPropertyClause" )
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
childBottomUpLocator = "childType" "childName" { OF "childType"
    "childName" } [ OF "mappableBottomUpLocator" ]
deleteConnectionLocator = CONNECTION ( FROM "mappableBottomUpLocator" [ TO
    "mappableBottomUpLocator" ] | TO "mappableBottomUpLocator" )
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
attributeName = "QUOTED_STRING"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { "," IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
pluggableMapName = "QUOTED_STRING"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
attributeNameList = "( " "attributeName" { "," "attributeName" } )"
bindableType = PLUGGABLE_MAPPING | OBJECT_TYPE | "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

mappingName

Name of the mapping.

alterMapDetailClause

Alter the detail of the mapping.

renameClause

Rename a mapping, mapping operator, mapping group, or mapping attribute.

alterOperatorOwnerDescendantsClause

Alter the desired child objects applicable to a mapping or a pluggable mapping.

setPropertiesClause

Describe the keys of properties for the map or objects in the map.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

addChildClause

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

The following is an example for creating a child object under a mapping

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

Note: Key word "OPERATOR" "GROUP" "ATTRIBUTE" are important for their respective ADD clauses. Without the key words, OMBPlus will interpret the ADD clause as an addChildClause. Here is an example:

```
OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'
```

```
OMB02932: Error getting child objects of type TABLE in M1
```

OMBPLUS interprets the ADD clause as one for creating a non-operator child object under the mapping object. Therefore, it tries to find type definition for non-operator child object "TABLE" and cannot find it.

addConnectionClause

Add connections between mapping groups or mapping attributes.

modifyOperatorClause

Modify a mapping operator.

modifyGroupClause

Modify a mapping group.

modifyAttributeClause

Modify a mapping attribute.

modifyChildClause

Modify a child that belongs to a mapping, mapping operator, mapping group or mapping attribute.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

childBottomUpLocator

Location of the child that belongs to a map, mapping operator, mapping group or mapping attribute.

deleteConnectionLocator

Delete connections between mapping operators, mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT,
CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPAND_OBJECT,
EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_
SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCast, LCRSPplitter,
MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER,
OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_
PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION,
TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

attributeName

Name of a mapping attribute.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributesBottomUpLocator

Location of a list of mapping attributes.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the operator

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the group

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the attribute

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the attribute

Name: DATATYPE
Type: STRING(20)
Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE
Default: "
Datatype of the Attribute

Name: LENGTH
Type: NUMBER
Valid Values: N/A
Default: 0
Length of the attribute.

Name: PRECISION
Type: NUMBER
Valid Values: N/A
Default: 0
Precision of the attribute.

Name: SCALE
Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for MAPPING:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_LANGUAGE

Type: STRING

Valid Values: ABAP, PLSQL, SQLLOADER, UNDEFINED

Default: UNDEFINED

The language used when generating code for the mapping.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing

data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,

NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,

NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,

NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,

NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,

NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,

NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,

NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,

NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_
NEIGHBORHOOD,
NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_
DESIG,
NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_
PARSESTATUSDESC,
NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
NA_PHONE,
NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
POSTALCODEFORMATTED,
NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
PRENAME,
NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
ROUTENAME,
NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
STREETCORRECTED,
NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
STREETNUMBERMATCH,
NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, || =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a
staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this
operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to

lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"MATCHMERGE.GENERAL.MATCH_NEW_RECORD_
CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name

attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY
Type: STRING(16)
Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR
Default: UNKNOWN
New Data Granularity

Name: DB_LOCATION
Type: STRING
Valid Values: N/A
Default: "
The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME
Type: STRING
Valid Values: N/A
Default: "
Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION
Type: STRING(30)
Valid Values: N/A
Default: "
oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT
Type: BOOLEAN
Valid Values: true, false
Default: false
Directly swap source into target as a partition without first creating a
staging table.

Name: ENABLE_CONSTRAINTS
Type: BOOLEAN
Valid Values: true, false
Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All

constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

pluggableMapName

Name of the pluggable map.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBALTER REAL_TIME_MAPPING 'MAP1' RENAME TO 'MAP2'
```

```
OMBALTER REAL_TIME_MAPPING 'MAP1'  
ADD CONNECTION FROM GROUP 'INOUTGRP1' OF OPERATOR 'CUST_SRC'  
TO GROUP 'INOUTGRP1' OF OPERATOR 'CUST_LOOK_UP'
```

```
OMBALTER REAL_TIME_MAPPING 'MAP1' DELETE OPERATOR 'OP1'
```

```
OMBALTER REAL_TIME_MAPPING 'MAP1' DELETE VARIABLE 'LAST_CUST'
```

```
OMBALTER REAL_TIME_MAPPING 'MAP1'  
MODIFY VARIABLE 'LAST_CUST'  
SET PROPERTIES (DATATYPE, LENGTH) VALUES ('VARCHAR2', 100)
```

See Also

OMBALTER, OMBCREATE REAL_TIME_MAPPING, OMBRETRIEVE REAL_TIME_MAPPING, OMBDROP REAL_TIME_MAPPING

OMBALTER REGISTERED_FUNCTION

Purpose

Alters a function that can be used in a query.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
alterRegisteredFunctionCommand = ( OMBALTER REGISTERED_FUNCTION
    "QUOTED_STRING" ( ( "renameClause" [ SET
        "setPropertyClauseforRegFun" ] [ SET "setReferenceIconSetClause" ] [
        UNSET "unsetReferenceIconSetClause" ] { "alterFunctionClauses" } ) |
    ( SET "setPropertyClauseforRegFun" [ SET "setReferenceIconSetClause"
        ] [ UNSET "unsetReferenceIconSetClause" ] { "alterFunctionClauses" }
    ) | ( SET "setReferenceIconSetClause" [ UNSET
        "unsetReferenceIconSetClause" ] { "alterFunctionClauses" } ) | ( UNSET
        "unsetReferenceIconSetClause" { "alterFunctionClauses" } ) ) | (
        "alterFunctionClauses" { "alterFunctionClauses" } ) ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertyClauseforRegFun = PROPERTIES "(" "propertyNameListforRegFun"
    ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterFunctionClauses = ADD "addFunctionArgClausesForAlter" | MODIFY
    "modifyFunctionArgClause" | DELETE "deleteFunctionArgClause"
propertyNameListforRegFun = ( "UNQUOTED_STRING" | ( PACKAGE | SIGNATURE )
    ) { "," ( "UNQUOTED_STRING" | ( PACKAGE | SIGNATURE ) ) }
propertyValueList = "propertyValue" { "," "propertyValue" }
addFunctionArgClausesForAlter = PARAMETER "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertyClause" ]
modifyFunctionArgClause = PARAMETER "QUOTED_STRING" [ "renameClause" ] [
    "moveParamToClause" ] [ SET "setPropertyClause" ]
deleteFunctionArgClause = PARAMETER "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
moveParamToClause = MOVE TO POSITION "INTEGER_LITERAL"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

alterRegisteredFunctionCommand

This clause alters a function.

QUOTED_STRING

name of the function.

renameClause

Renames a function with a different name.

setPropertyClauseforRegFun

This clause sets the properties of the object.

setReferenceIconSetClause

Set specified Icon Set.

unsetReferenceIconSetClause

Unset specified Icon Set.

alterFunctionClauses

This clause alters the function parameters.

propertyNameListforRegFun

This is the list of property names.

propertyValueList

This is the list of property values.

addFunctionArgClausesForAlter

This clause adds a function parameter.

QUOTED_STRING

name of the parameter.

modifyFunctionArgClause

This clause modifies a function parameter.

QUOTED_STRING

name of the parameter.

deleteFunctionArgClause

This clause deletes a function parameter.

QUOTED_STRING

name of the parameter.

propertyValue

This is a property value.

setPropertyClause

This clause sets the properties of the object.

Basic properties for REGISTERED_FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the function

Name: AVAILABLE

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the Function is available for the user to use in calculations

Name: RETURN_TYPE

Type: STRING()

Valid Values: BINARY_INTEGER, BLOB, BOOLEAN, CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, SYS.ANYDATA, SYS.ROW_LCR, SYS_REFCURSOR, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE, SYS.XMLFORMAT,

BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
SYS.XMLSEQUENCETYPE,

BLAST_MATCH_PLSQLRECORDTYPE

Default: 'NUMBER'

Return type of the function

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: ''

Business name of the parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: ''

Description of the parameter

Name: DATATYPE

Type: STRING()

Valid Values: BINARY_INTEGER, BLOB, BOOLEAN, CHAR, CLOB, DATE, FLOAT,
INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH
NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, SYS.ANYDATA,
SYS.ROW_LCR, SYS_REFCURSOR, TIMESTAMP, TIMESTAMP WITH LOCAL TIME
ZONE

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE,
SYS.XMLFORMAT,

BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
SYS.XMLSEQUENCETYPE,

BLAST_MATCH_PLSQLRECORDTYPE

Default: 'NUMBER'

Datatype of the parameter

Properties for REGISTERED_FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for the referenced Function

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: PACKAGE

Type: STRING

Valid Values: N/A

Default: "

May be used to identify the name of a Package that contains the Function

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

moveParamToClause

This clause moves parameters within registered functions.

propertyNameList

This is the list of property names.

Examples

```
OMBALTER REGISTERED_FUNCTION 'My_Sum' SET PROPERTIES  
(DESCRIPTION) VALUES  
( 'My summation' )
```

See Also

OMBCREATE REGISTERED_FUNCTION, OMBRETRIEVE REGISTERED_FUNCTION

OMBALTER ROLE

Purpose

To alter properties of a Warehouse Builder role.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
alterRoleCommand = OMBALTER ( ROLE "QUOTED_STRING" ( "renameClause" [ SET
    "setPropertyClause" ] | SET "setPropertyClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterRoleCommand

This clause alters a Warehouse Builder role.

renameClause

renames a role with a different name.

setPropertyClause

Used to set properties of a Warehouse Builder role. Valid properties are as shown:

Basic properties for ROLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the role

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the role

Examples

```
OMBALTER ROLE 'DEVELOPER_ROLE' RENAME TO 'DEVELOPMENT_ROLE' SET  
PROPERTIES(BUSINESS_NAME, DESCRIPTION) VALUES('Warehouse development  
role',  
'only be granted to developers')
```

See Also

OMBCREATE ROLE, OMBDROP ROLE

OMBALTER SAP_MODULE

Purpose

Alter the SAP module by renaming it, and/or reset its properties.

It is not supported in the current release.

Prerequisites

You must open a project to alter a SAP module.

Syntax

```
alterSAPModuleCommand = OMBALTER ( SAP_MODULE "QUOTED_STRING" (
    "renameClause" [
        "alterPropertiesOrReferenceClauseForDataMetadadataModule" ] |
        "alterPropertiesOrReferenceClauseForDataMetadadataModule" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrReferenceClauseForDataMetadadataModule = ( ( SET ( (
    "alterPropertiesClause" [ ( SET
        "setReferenceClauseForDataMetadadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadadataModule" ] ) | ( UNSET
        "unsetReferenceClauseForDataMetadadataModule" [ SET
        "setReferenceClauseForDataMetadadataModule" ] ) ) | (
        "setReferenceClauseForDataMetadadataModule" [ UNSET
        "unsetReferenceClauseForDataMetadadataModule" ] ) ) ) | ( UNSET
        "unsetReferenceClauseForDataMetadadataModule" [ SET
        "setReferenceClauseForDataMetadadataModule" ] ) ) [
        "addOrRemoveOrModifyModuleReferenceLocationClause" ]
    alterPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
        "propertyValueList" ")"
    setReferenceClauseForDataMetadadataModule = ( "setReferenceLocationClause" [
        SET "setReferenceMetadataLocationOrIconSetClause" ] |
        "setReferenceMetadataLocationOrIconSetClause" )
    unsetReferenceClauseForDataMetadadataModule = (
        "unsetReferenceLocationClause" [ UNSET
        "unsetReferenceMetadataLocationOrIconSetClause" ] |
        "unsetReferenceMetadataLocationOrIconSetClause" )
    addOrRemoveOrModifyModuleReferenceLocationClause = (
        "addReferenceLocationClause" | "removeReferenceLocationClause" |
        "modifyReferenceLocationClause" ) { "addReferenceLocationClause" |
        "removeReferenceLocationClause" | "modifyReferenceLocationClause" }
    propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
    propertyValueList = "propertyValue" { "," "propertyValue" }
    setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
    setReferenceMetadataLocationOrIconSetClause = (
        "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
        ] | "setReferenceIconSetClause" )
    unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
        "QUOTED_STRING"
    unsetReferenceMetadataLocationOrIconSetClause = (
        "unsetReferenceMetadataLocationClause" [ UNSET
        "unsetReferenceIconSetClause" ] | "unsetReferenceIconSetClause" )
    addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
        "QUOTED_STRING" [ SET AS DEFAULT ]
    removeReferenceLocationClause = REMOVE ( REFERENCE | REF ) LOCATION
        "QUOTED_STRING"
    modifyReferenceLocationClause = MODIFY ( REFERENCE | REF ) LOCATION
```

```
"QUOTED_STRING" SET AS DEFAULT
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
  "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceMetadataLocationClause = ( REFERENCE | REF )
  METADATA_LOCATION "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
```

Keywords And Parameters

alterSAPModuleCommand

Modify an existing SAP module.

renameClause

Rename a SAP module.

alterPropertiesOrReferenceClauseForDataMetadataModule

Alter existing SAP module's properties and/or locations and/or icon sets.

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the SAP module.

unsetReferenceClauseForDataMetadataModule

Unset location and/or icon set for the SAP module.

addOrRemoveOrModifyModuleReferenceLocationClause

Add/remove/modify runtime location for the SAP module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a location to the existing SAP module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the SAP module.

unsetReferenceLocationClause

Unset a location to the existing SAP module.

unsetReferenceMetadataLocationOrIconSetClause

Unset metadata location and/or icon set for the SAP module.

addReferenceLocationClause

Add a runtime location to the SAP module.

removeReferenceLocationClause

Remove a runtime location from the SAP module.

modifyReferenceLocationClause

Modify a runtime location of the SAP module.

propertyValue

Value of a property.

setReferenceMetadataLocationClause

Set metadata location for the SAP module.

setReferenceIconSetClause

Set icon set for the SAP module.

unsetReferenceMetadataLocationClause

Unset metadata location for the SAP module.

unsetReferenceIconSetClause

Unset icon set for the SAP module.

Examples

```
OMBALTER SAP_MODULE 'src_module' RENAME TO 'tgt_module' SET
PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('This becomes a target module.',
'target module')
```

This will rename the SAP module "src_module" to "tgt_module", and set its description to "This becomes a target module", set its business name to "target module".

See Also

OMBALTER, OMBCREATE SAP_MODULE, OMBDROP SAP_MODULE

OMBALTER SEQUENCE

Purpose

To alter properties and definition of a sequence.

Prerequisites

In the context of an Oracle Module.

Syntax

```
alterSequenceCommand = OMBALTER ( SEQUENCE "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] { MODIFY
    "modifySequenceColumnNameClause" } | "alterPropertiesOrIconSetClause" {
    MODIFY "modifySequenceColumnNameClause" } | ( MODIFY
    "modifySequenceColumnNameClause" )+ ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
modifySequenceColumnNameClause = COLUMN "QUOTED_STRING" SET
    "setPropertiesClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterSequenceCommand

This clause is for sequence alter command.

renameClause

renames a table with a different name.

modifySequenceColumnNameClause

This clause modifies the sequence's column.

setPropertiesClause

set sequences properties.

Basic properties for SEQUENCE:

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the sequence.

Name: CURRVAL
Type: NUMBER
Valid Values: N/A
Default: 1
current increment value.

Name: NEXTVAL
Type: NUMBER
Valid Values: N/A
Default: 1
next increment value. next increment value.

Properties for SEQUENCE:

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true
Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Name: INCREMENT_BY
Type: NUMBER
Valid Values: -2147483648 - 2147483647
Default: 1

Sequence Incremented By

Name: START_WITH

Type: NUMBER

Valid Values: -2147483648 - 2147483647

Default: 1

Sequence Starts With

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER SEQUENCE 'NEW_SEQUENCE' SET PROPERTIES (DESCRIPTION)
VALUES ('this
```

is an altered desc of new sequence')

This will alter a sequence named "NEW_SEQUENCE", its description is "this

is an altered desc of new sequence."

See Also

OMBALTER, OMBCREATE SEQUENCE, OMBDROP SEQUENCE, OMBRETRIEVE SEQUENCE

OMBALTER SNAPSHOT

Purpose

A snapshot can be altered to remove, add or update components.

Prerequisites

The snapshot to be altered should already exist. This command can be executed for any component regardless of current context.

Syntax

```
parseAlterCommand = OMBALTER "alterSnapshotCommand"
alterSnapshotCommand = ( SNAPSHOT "QUOTED_STRING" ( SET
    "setPropertyClause" | ( ( ( ADD | MODIFY ) "objectClause" ) | (
    DELETE "UNQUOTED_STRING" "QUOTED_STRING" )+ ) )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
objectClause = "UNQUOTED_STRING" "QUOTED_STRING" [ CASCADE | NO CASCADE ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parseAlterCommand

Root production for OMBALTER SNAPSHOT.

alterSnapshotCommand

To alter already existing snapshot.

QUOTED_STRING

Name of snapshot to be altered.

ADD

Add components to snapshot.

MODIFY

Replace an already existing component with the latest definition of the component from repository.

DELETE

Remove component from snapshot.

setPropertyClause

Properties of snapshot can be altered through this optional clause.

Properties of Snapshot is DESCRIPTION and TYPE. Altering TYPE property is an irreversible operation. Altering snapshot from FULL to SIGNATURE will only keep information required for comparing. SIGNATURE snapshots cannot be exported.

Basic properties for SNAPSHOT:

Name: TYPE

Type: STRING(200)

Valid Values: FULL,SIGNATURE

Default: FULL

This is the type of snapshot

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the snapshot

propertyNameList

Property names for SNAPSHOT that can be altered.

propertyValueList

List of property values for SNAPSHOT.

propertyValue

Allowable value types for a snapshot property.

Examples

```
OMBALTER SNAPSHOT 'S1' ADD TABLE '/Project1/WH2/T3'
```

This command adds table T3 into snapshot S1.

```
OMBALTER SNAPSHOT 'S1' DELETE TABLE '/Project1/WH2/T1'
```

This command removes T1 table from snapshot S1, if system can find that table in snapshot.

OMBALTER SNAPSHOT 'S1' MODIFY TABLE '/Project1/WH2/T1'

This command updates definition of component T1 in snapshot from repository.

OMBALTER SNAPSHOT 'S1' SET PROPERTIES (DESCRIPTION) VALUES ('this is new description');

This command sets new description for snapshot.

OMBALTER SNAPSHOT 'S1' SET PROPERTIES (TYPE) VALUES('SIGNATURE')

This command transforms a FULL snapshot into a SIGNATURE snapshot, which is only useful for the compare service.

See Also

OMBCREATE SNAPSHOT, OMBDROP SNAPSHOT, OMBRESTORE SNAPSHOT, OMBCOMPARE SNAPSHOT, OMBLIST SNAPSHOT, OMBRETRIEVE SNAPSHOT

OMBALTER STREAMS_CAPTURE_PROCESS

Purpose

Alter the Streams Capture Process by resetting its properties and adding/removing tables to capture.

Prerequisites

Should be in the context of Streams Queue.

Syntax

```
alterCaptureCommand = OMBALTER STREAMS_CAPTURE_PROCESS "QUOTED_STRING" (
    "renameClause" [ SET "setCapturePropertiesClause" ] [ (
        "addTableClause" | "deleteTableClause" )+ ] | SET
        "setCapturePropertiesClause" [ ( "addTableClause" |
        "deleteTableClause" )+ ] | ( "addTableClause" | "deleteTableClause" )+
    )
renameClause = RENAME TO "QUOTED_STRING"
setCapturePropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES
    "(" "propertyValueList" ")"
addTableClause = ADD TABLE "QUOTED_STRING"
deleteTableClause = DELETE TABLE "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterCaptureCommand

Alters the Streams Capture Process with the given name by either renaming it or by setting it's properties or both.

addTableClause

Add a table to the set of tables whose changes are to be captured by this Streams Capture Process

deleteTableClause

Remove a table from the set of tables whose changes are to be captured by this Streams Capture Process

propertyNameList

The list of properties.

Basic properties for STREAMS_CAPTURE_PROCESS:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Streams Capture Process

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Streams Capture

Properties for STREAMS_CAPTURE_PROCESS:

Name: CAPTURE_START_PARAMETER

Type: STRING

Valid Values: START_DATE, START_SCN

Default: START_SCN

This specifies whether the Streams Capture Process should start capturing changes based on the Start Date or the Start SCN.

Name: CAPTURE_TAGGED_LCR

Type: BOOLEAN

Valid Values: true, false

Default: false

If TRUE, then a redo entry is always considered for capture and an LCR is always considered for apply, regardless of whether redo entry or LCR has a non-NULL tag. If FALSE, then a redo entry is considered for capture and an LCR is considered for apply only when the redo entry or the LCR contains a NULL tag.

Name: CAPTURE_TIMEOUT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The maximum number of seconds to wait for another instance of the same capture process to finish.

Name: DBA_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location of the DBA user who should create the supplemental logs.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether the Object is deployable or not.

Name: DISABLE_ON_LIMIT

Type: BOOLEAN

Valid Values: true, false

Default: true

If set to true, the capture process will be disabled once the message/time limit is reached.

Name: MAXIMUM_SCN

Type: NUMBER

Valid Values: 0 - 1000000000

Default: 0

This is the Maximum SCN value whose corresponding changes will be captured by the Streams Capture Process.

Name: MESSAGE_COUNT_LIMIT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

If Disable On Limit is set to true, then the Streams Capture Process is disabled once the specified number of messages have been captured.

Name: PARALLELISM_DEGREE

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The number of parallel server process that will mine the redo logs.

Name: START_DATE

Type: STRING

Valid Values: N/A

Default: 1970-01-01

The user specified date from which the Streams Capture Process should start capturing changes.

Name: START_SCN

Type: NUMBER

Valid Values: N/A

Default: 0

The user specified SCN from which the Streams Capture Process should start capturing changes.

Name: TIME_LIMIT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

If Disable On Limit is set to true, then the Streams Capture Process is disabled once the specified number of seconds elapse.

Name: WRITE_ALERT_LOG

Type: BOOLEAN

Valid Values: true, false

Default: true

If set to true, then the Streams Capture Process writes a message to the alert log on exit.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER STREAMS_CAPTURE_PROCESS 'SOME_CAPTURE_PROCESS' SET  
PROPERTIES
```

```
(DISABLE_ON_LIMIT,MAXIMUM_SCN, MESSAGE_COUNT_LIMIT,  
PARALLELISM_DEGREE,
```

```
CAPTURE_TIMEOUT, TIME_LIMIT, WRITE_ALERT_LOG, CAPTURE_START_  
PARAMETER,
```

```
START_SCN, CAPTURE_TAGGED_LCR) VALUES
```

```
('true','999','100','10','100','100','false','START_SCN','100')
```

This will set its properties as specified.

See Also

OMBALTER, OMBCREATE STREAMS_CAPTURE_PROCESS, OMBRETRIEVE
STREAMS_CAPTURE_PROCESS, OMBDROP STREAMS_CAPTURE_PROCESS

OMBALTER STREAMS_QUEUE

Purpose

Alter the Streams Queue by resetting its properties.

Prerequisites

Should be in the context of an Oracle Module. The Queue Table should exist in the same Oracle Module.

Syntax

```
alterANYQCommand = OMBALTER ( STREAMS_QUEUE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertyClause" ] | SET
    "setPropertyClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterANYQCommand

Alters the Streams Queue with the given name by either renaming it or by setting it's properties or both.

renameClause

Renames the Streams Queue to the given name.

setPropertyClause

Sets properties (core, logical, physical, user-defined) for Advanced Queue.

Valid properties are as shown:

Basic properties for `ADVANCED_QUEUE`:

Name: `BUSINESS_NAME`

Type: `STRING(200)`

Valid Values: `N/A`

Default: `"`

Business name of the Advanced Queue

Name: `DESCRIPTION`

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Advanced Queue

Name: QTABLE

Type: STRING(4000)

Valid Values: N/A

Default: "

Queue Table for the Advanced Queue. This has to be the name of a Queue Table(Queue_Table) existing in the same Oracle Module.

Properties for STREAMS_QUEUE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DEQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Dequeue Enabled for AQ

Name: ENQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enqueue enabled for AQ

Name: GENERATE_ADVANCED_QUEUE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the Advanced Queue.

Name: GENERATE_TEMPORARY_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the temporary table.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MAX_RETRIES

Type: NUMBER

Valid Values: N/A

Default: 5

Max. number of Retries

Name: RETENTION_TIME

Type: NUMBER

Valid Values: N/A

Default: 0

Message retention time

Name: RETRY_DELAY

Type: NUMBER

Valid Values: N/A

Default: 0

Delay period before retry

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER STREAMS_QUEUE 'SOME_STREAMS_QUEUE' SET PROPERTIES  
(MAX_RETRIES,  
RETRY_DELAY, RETENTION_TIME, ENQUEUE_ENABLED, DEQUEUE_ENABLED)  
VALUES  
(10,20,60,'true','true','true','false')
```

This will set its properties as specified.

See Also

OMBALTER, OMBCREATE STREAMS_QUEUE, OMBRETRIEVE STREAMS_QUEUE,
OMBDROP STREAMS_QUEUE

OMBALTER TABLE to OMBALTER VIEW

This chapter lists commands associated with OMBALTER in alphabetical order, starting with the command OMBALTER TABLE.

OMBALTER TABLE

Purpose

To alter properties and definition of a table.

Prerequisites

In the context of an Oracle Module.

Syntax

```
alterTableCommand = OMBALTER ( TABLE "QUOTED_STRING" ( "renameClause" [
    "alterPropertiesOrIconSetClause" ] [ "alterTableSCOClauses" ] |
    "alterPropertiesOrIconSetClause" [ "alterTableSCOClauses" ] |
    "alterTableSCOClauses" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
alterTableSCOClauses = ADD ( "addColumnClauseForAlter" |
    "addConstraintClause" | "addSCOClause" | "addDataRuleUsageClause" ) [
    "alterTableSCOClauses" ] | MODIFY ( "modifyColumnClause" |
    "modifyConstraintClause" | "modifySCOClause" |
    "modifyDataRuleUsageClause" ) [ "alterTableSCOClauses" ] | DELETE (
    "deleteColumnClause" | "deleteConstraintClause" | "deleteSCOClause" |
    "deleteDataRuleUsageClause" ) [ "alterTableSCOClauses" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
addColumnClauseForAlter = COLUMN "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ]
addConstraintClause = "addUkPkClause" | "addFkClause" |
    "addCheckConstraintClause"
addSCOClause = "addIndexClause" | "addIndexPartitionClause" |
    "addIndexPartitionKeyClause" | "addPartitionClause" |
    "addPartitionKeyClause" | "addSubpartitionClause" |
    "addaddMaterializedViewSCOandDependentClauseClause" |
    "addSubPartitionKeyClause" | "addIndexColumnClause"
addDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" SET REF DATA_RULE
    "QUOTED_STRING" ( GROUP "QUOTED_STRING" SET REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) "QUOTED_STRING" ( ATTRIBUTE
    "QUOTED_STRING" SET REF COLUMN "QUOTED_STRING" )+ )+ [ SET
    "setPropertiesClause" ]
modifyColumnClause = COLUMN "QUOTED_STRING" ( "renameClause" [
    "moveToClause" ] [ SET "setPropertiesClause" ] | "moveToClause" [ SET
    "setPropertiesClause" ] | SET "setPropertiesClause" )
modifyConstraintClause = "modifyUkPkClause" | "modifyFkClause" |
    "modifyCheckConstraintClause"
modifySCOClause = "modifyIndexClause" | "modifyIndexPartitionClause" |
    "modifyIndexPartitionKeyClause" | "modifyPartitionClause" |
    "modifyPartitionKeyClause" |
    "modifyaddMaterializedViewSCOandDependentClauseClause" |
    "modifySubPartitionClause" | "modifySubPartitionKeyClause" |
    "modifyIndexColumnClause"
modifyDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" (
```

```

        "renameClause" [ SET "setPropertiesClause" ] | SET
        "setPropertiesClause" )
deleteColumnClause = COLUMN "QUOTED_STRING"
deleteConstraintClause = UNIQUE_KEY "QUOTED_STRING" | PRIMARY_KEY
        "QUOTED_STRING" | FOREIGN_KEY "QUOTED_STRING" | CHECK_CONSTRAINT
        "QUOTED_STRING"
deleteSCOClauses = INDEX "QUOTED_STRING" | PARTITION "QUOTED_STRING" |
        PARTITION_KEY "QUOTED_STRING" | TEMPLATE_SUBPARTITION "QUOTED_STRING"
        | SUBPARTITION_KEY "QUOTED_STRING" | INDEX_COLUMN "QUOTED_STRING" OF
        INDEX "QUOTED_STRING" | INDEX_PARTITION "QUOTED_STRING" OF INDEX
        "QUOTED_STRING" | INDEX_PARTITION_KEY "QUOTED_STRING" OF INDEX
        "QUOTED_STRING" | SUBPARTITION "QUOTED_STRING" OF PARTITION
        "QUOTED_STRING"
deletedataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" [ SET
        "setUkPkPropertiesAndReferencesColumnsClauses" ]
addFkClause = FOREIGN_KEY "QUOTED_STRING" [ SET "setFkSubClauses" ]
addCheckConstraintClause = CHECK_CONSTRAINT "QUOTED_STRING" [ SET
        "setPropertiesClause" ]
addIndexClause = INDEX "QUOTED_STRING" [ SET
        "setSCOConfigurationPropertiesClauses" ]
addIndexPartitionClause = INDEX_PARTITION "QUOTED_STRING" OF INDEX
        "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] (
        "renameSCOConfigurationClause" [ SET
        "setSCOConfigurationPropertiesClauses" ] | [ SET
        "setSCOConfigurationPropertiesClauses" ] )
addIndexPartitionKeyClause = INDEX_PARTITION_KEY "QUOTED_STRING" OF INDEX
        "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionClause = PARTITION "QUOTED_STRING" [ AT POSITION
        "INTEGER_LITERAL" ] [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionKeyClause = PARTITION_KEY "QUOTED_STRING" [ SET
        "setSCOConfigurationPropertiesClauses" ]
addSubpartitionClause = SUBPARTITION "QUOTED_STRING" OF PARTITION
        "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
        "setSCOConfigurationPropertiesClauses" ]
addMaterializedViewSCOandDependentClauseClause = TEMPLATE_SUBPARTITION
        "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
        "setSCOConfigurationPropertiesClauses" ]
addSubPartitionKeyClause = SUBPARTITION_KEY "QUOTED_STRING" [ SET
        "setSCOConfigurationPropertiesClauses" ]
addIndexColumnClause = INDEX_COLUMN "QUOTED_STRING" OF INDEX
        "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
modifyUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" (
        "renameClause" [ SET "setUkPkPropertiesAndReferencesColumnsClauses" ]
        | SET "setUkPkPropertiesAndReferencesColumnsClauses" )
modifyFkClause = FOREIGN_KEY "QUOTED_STRING" ( "renameClause" [ SET
        "setFkSubClauses" ] | SET "setFkSubClauses" )
modifyCheckConstraintClause = CHECK_CONSTRAINT "QUOTED_STRING" (
        "renameClause" [ SET "setPropertiesClause" ] | SET
        "setPropertiesClause" )
modifyIndexClause = INDEX "QUOTED_STRING" ( "renameSCOConfigurationClause"
        [ SET "setSCOConfigurationPropertiesClauses" ] | SET
        "setSCOConfigurationPropertiesClauses" )
modifyIndexPartitionClause = INDEX_PARTITION "QUOTED_STRING" OF INDEX
        "QUOTED_STRING" ( "renameSCOConfigurationClause" [
        "moveToClauseIndexPartition" ] [ SET
        "setSCOConfigurationPropertiesClauses" ] |

```

```

"moveToClauseIndexPartition" [ SET
"setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
modifyIndexPartitionKeyClause = INDEX_PARTITION_KEY "QUOTED_STRING" OF
INDEX "QUOTED_STRING" ( SET "setSCOConfigurationPropertiesClauses" )
modifyPartitionClause = PARTITION "QUOTED_STRING" (
"renameSCOConfigurationClause" [ "moveToClausePartition" ] [ SET
"setSCOConfigurationPropertiesClauses" ] | "moveToClausePartition" [
SET "setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
modifyPartitionKeyClause = PARTITION_KEY "QUOTED_STRING" (
"renameSCOConfigurationClause" [ SET
"setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
modifyaddMaterializedViewSCOandDependentClauseClause =
TEMPLATE_SUBPARTITION "QUOTED_STRING" ( "renameSCOConfigurationClause"
[ "moveToClauseTemplateSubPartition" ] [ SET
"setSCOConfigurationPropertiesClauses" ] |
"moveToClauseTemplateSubPartition" [ SET
"setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
modifySubPartitionClause = SUBPARTITION "QUOTED_STRING" OF PARTITION
"QUOTED_STRING" ( "renameSCOConfigurationClause" [
"moveToClauseSubPartition" ] [ SET
"setSCOConfigurationPropertiesClauses" ] | "moveToClauseSubPartition"
[ SET "setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
modifySubPartitionKeyClause = SUBPARTITION_KEY "QUOTED_STRING" (
"renameSCOConfigurationClause" [ SET
"setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
modifyIndexColumnClause = INDEX_COLUMN "QUOTED_STRING" OF INDEX
"QUOTED_STRING" ( "renameSCOConfigurationClause" [
"moveToClauseForIndexColumn" ] [ SET
"setSCOConfigurationPropertiesClauses" ] |
"moveToClauseForIndexColumn" [ SET
"setSCOConfigurationPropertiesClauses" ] | SET
"setSCOConfigurationPropertiesClauses" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
"FLOATING_POINT_LITERAL" )
setUkPkPropertiesAndReferencesColumnsClauses = "setPropertiesClause" [ SET
( REF | REFERENCE ) "constraintColumnReferencesClause" ] | ( REF |
REFERENCE ) "constraintColumnReferencesClause"
setFkSubClauses = "setPropertiesClause" [ SET ( REF | REFERENCE )
"setFkReferencesClauses" ] | ( REF | REFERENCE )
"setFkReferencesClauses"
setSCOConfigurationPropertiesClauses = PROPERTIES "(" "propertyNameList"
)" VALUES "(" "propertyValueList" ")"
renameSCOConfigurationClause = RENAME TO "QUOTED_STRING"
moveToClauseIndexPartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClausePartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClauseTemplateSubPartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClauseSubPartition = MOVE TO POSITION "INTEGER_LITERAL"
moveToClauseForIndexColumn = MOVE TO POSITION "INTEGER_LITERAL"
constraintColumnReferencesClause = COLUMNS "(" "quotedNameList" ")"
setFkReferencesClauses = "constraintColumnReferencesClause" [ SET ( REF |
REFERENCE ) "constraintUkReferencesClause" ] |
"constraintUkReferencesClause" [ SET ( REF | REFERENCE )
"constraintColumnReferencesClause" ]
quotedNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }

```

```
constraintUkReferencesClause = ( UNIQUE_KEY | PRIMARY_KEY )
    "QUOTED_STRING" [ OF ( TABLE | VIEW ) "QUOTED_STRING" ]
```

Keywords And Parameters

alterTableCommand

This clause alters a table.

QUOTED_STRING

name of the table.

renameClause

renames a table with a different name.

alterTableSCOClauses

This clause will add, modify, delete, columns, configuration, and keys.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for tables (including partitions and subpartitions) and their columns, indexes (including index partitions), check constraints, unique keys, foreign keys, and primary keys.

Basic properties for TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the table

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Default value of the column

Name: NOT_NULL

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify "true" to enforce Not Null restriction on a column.

Basic properties for PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the table rows according to a Hash Algorithm, lists of values, or specified ranges.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH partitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of partitions that is a power of 2. If you have multiple Partition Keys, you only have to specify once.

Basic properties for PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Commas can be escaped using "" (for example '1,2,3'). Always specify DEFAULT as the value of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH customized subpartitions the database should create for a particular main RANGE partition (RANGE-HASH BY QUANTITY partitioning). For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once. Set it to 0 to reverse to the use of generic template HASH_QUANTITY specified in SUBPARTITION_KEY.

Basic properties for SUBPARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

For partition-level partitioning according to a Hash Algorithm or lists of values. Each partition is further sorted into subpartitions.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH subpartitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once.

Basic properties for SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST subpartition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last subpartition, and make sure you have specified

SUBPARTITION_KEY(s) before you specify any SUBPARTITION. Each LIST subpartition must have at least one value. No value, including NULL, can appear in more than one subpartition.

Basic properties for TEMPLATE_SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

In composite partitioning, template subpartitions are automatically applied to those partitions without their subpartitions specified. Here for LIST subpartitions only, specify a comma-delimited, ordered list of literal values corresponding to the LIST subpartitioning column. Always specify DEFAULT as the value of the last template LIST subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any TEMPLATE_SUBPARTITION. Each LIST template subpartition must have at least one value. No value, including NULL, can appear in more than one template subpartition.

Basic properties for INDEX:

Name: INDEX_TYPE

Type: STRING

Valid Values: UNIQUE, NON-UNIQUE, BITMAP, FUNCTION-BASED

Default: (No default, must be one of the preceding choices)

Specify the type of an index. NORMAL can be used in place of NON-UNIQUE.

Name: LOCAL_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify if an index is Global or Local. The default is Global.

Specify Local so that the index is partitioned on the same columns, with the same number of partitions and the same partition bounds as table.

Oracle Database automatically maintains local index partitioning as the underlying table is repartitioned.

Name: COLUMN_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

Specify an expression built from columns of table, constants, SQL functions, and user-defined functions to create a FUNCTION-BASED index.

Basic properties for INDEX_PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the index rows according to a Hash Algorithm, lists of values, or specified ranges. Hash index partitioning is supported starting with Oracle Database 10g version.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH index partitions the database should create based on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of index partitions that is a power of 2. If you have multiple index Partition Keys, you only have to specify once.

Basic properties for INDEX_PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition in a global index. The value list is a comma-delimited, ordered list of literal

values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition. No need to specify VALUES_LESS_THAN for Local index.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last partition. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition. No need to specify VALUES_EQUAL_TO for Local index.

Basic properties for CHECK_CONSTRAINTS:

Name: CHECK_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Specify a condition that each row in the table must satisfy.

Properties for TABLE:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: CACHE_MODE

Type: STRING

Valid Values: , CACHE, NOCACHE

Default: "

Indicate how Oracle should store blocks in the buffer cache.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: MONITORING_MODE

Type: STRING

Valid Values: , MONITORING, NOMONITORING

Default: "

Specify MONITORING if you want modification statistics to be collected on this table.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow tablespaces. The number of tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of tablespaces, then Oracle cycles through the names of the tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. For composite-partitioned tables, it is used for subpartition template to store a list of tablespaces.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: ROWDEPENDENCIES_MODE

Type: STRING

Valid Values: , NOROWDEPENDENCIES, ROWDEPENDENCIES

Default: "

Specify ROWDEPENDENCIES to use row-level dependency tracking.

Name: ROW_MOVEMENT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify whether Oracle can move a table row.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PRIMARY_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify

(INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE

disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for CHECK_CONSTRAINT:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for

query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for PARTITION:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1 for Table and 2 for Index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for INDEX:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: COMPUTESTATISTICS

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to collect statistics at relatively little cost during the creation of an index.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: INDEXORDER

Type: STRING

Valid Values: , ASC, DESC

Default: "

Use ASC or DESC to indicate whether the index should be created in ascending or descending order. The Oracle default is ASC.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2 for Index.

Name: KEYCOMPRESS

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Specify COMPRESS to enable key compression.

Name: KEYCOMPRESSPREFIXLENGTH

Type: STRING

Valid Values: N/A

Default: "

Specify the prefix length (number of prefix columns to compress). For unique indexes, the valid range of prefix length values is from 1 to the number of key columns minus 1. The default prefix length is the number of key columns minus 1. For nonunique indexes, the valid range of prefix length values is from 1 to the number of key columns. The default prefix length is the number of key columns. Oracle compresses only nonpartitioned indexes that are nonunique or unique indexes of at least two columns. You cannot specify COMPRESS for a bitmap index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (2-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: ONLINE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to indicate that DML operations on the table will be allowed during creation of the index.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of Index tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow Index tablespaces. The number of Index tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of Index tablespaces, then Oracle cycles through the names of the Index tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for a locally partitioned index. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. If specified, then individual local Hash index partitions specified will be ignored for Local Hash or Range-Hash Index partitioning.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: SORT

Type: STRING

Valid Values: , NOSORT, REVERSE, SORT

Default: "

Specify NOSORT to indicate to Oracle that the rows are already stored in the database in ascending order. Specify REVERSE to store the bytes of the index block in reverse order, excluding the rowid.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

`addColumnClauseForAlter`

This clause adds a column at a particular position.

When you alter a table and add columns to it, the position you specify for a new column must be less than or equal to the number of columns added up to that point in the OMBALTER command.

For example, a table TEMP_TAB contains three columns. You use the following

OMBALTER TABLE command to add three more columns:

```
OMBALTER TABLE 'TEMP_TAB' \
```

```
ADD COLUMN 'C4' AT POSITION 4 \
```

```
SET PROPERTIES(DATATYPE,PRECISION) VALUES('NUMBER',7) \
```

```
ADD COLUMN 'C5' AT POSITION 6 \
SET PROPERTIES(DATATYPE) VALUES('VARCHAR2') \
ADD COLUMN 'C6' AT POSITION 5 \
SET PROPERTIES(DATATYPE,PRECISION) VALUES('NUMBER',10);
```

This command does not execute successfully because at the point when you specify the position of the column C5 as 6, the table contains only 5 columns.

QUOTED_STRING

The column name.

addConstraintClause

Adds primary and unique key, and add check constraints.

addSCOClause

This clause will add SCOs.

addDataRuleUsageClause

Add a data rule usage to the relation.

modifyColumnClause

This clause renames, set properties, and move columns.

modifyConstraintClause

This clause modifies keys and check constraints

modifySCOClause

This clause will modify SCOs.

modifyDataRuleUsageClause

Rename or modify the properties of a data rule usage.

deleteColumnClause

This clause deletes a column.

deleteConstraintClause

This clause deletes a key or check constraint.

deleteSCOClause

This clause deletes a SCO.

QUOTED_STRING

Either index, partition, partition_key, or index column name.

deleteDataRuleUsageClause

Delete a data rule usage.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

addUkPkClause

This clause adds the adds unique key and primary keys.

QUOTED_STRING

name of the unique key or primary key.

addFkClause

This clause adds foreign key.

QUOTED_STRING

Name of the foreign key.

addCheckConstraintClause

add a check constraint.

QUOTED_STRING

Name of the CheckConstraint.

addIndexClause

This clause adds an index.

QUOTED_STRING

Name of the index.

addPartitionClause

This clause adds a partition.

QUOTED_STRING

Name of the partition.

addPartitionKeyClause

This clause adds a partition key.

QUOTED_STRING

Name of the partition key. This should be a column identifier.

addIndexColumnClause

This clause will add index column to a specified index.

QUOTED_STRING

This should be a column identifier of owning object (such as a table) of the index.

moveToClause

This clause will move the column to given position.

modifyUkPkClause

It modifies unique or primary key.

modifyFkClause

This clause modifies the foreign key.

modifyCheckConstraintClause

This clause modifies the check constraint.

modifyIndexClause

This clause modifies the Index.

QUOTED_STRING

Name of the index.

modifyPartitionClause

This clause modifies a partition.

QUOTED_STRING

Name of the partition.

modifyPartitionKeyClause

This clause modifies a partition key.

QUOTED_STRING

Name of the partition key.

modifyIndexColumnClause

Modifies the Index Column. The first quoted_string in this clause denotes index column name, and the latter denotes index.

propertyValue

This clause adds the property values.

setUkPkPropertiesAndReferencesColumnsClauses

This clause adds properties and references to columns.

setFkSubClauses

This clause set references to a foreign key.

setSCOConfigurationPropertiesClauses

Set the configuration properties for the following:

- Partition, Subpartition, and Template Subpartition: All refer to configuration properties of Partition.
- Index, and Index Partition: For Index Partition, refer to configuration properties of Partition.

renameSCOConfigurationClause

This clause renames configuration objects.

constraintColumnReferencesClause

This clause provides names of all columns.

setFkReferencesClauses

This clause sets foreign key references.

quotedNameList

This clause gives column names.

constraintUkReferencesClause

The first QUOTED_STRING denotes the UniqueKey or Primary key name, and the latter denotes the table's or view's name.

Examples

```
OMBALTER TABLE 'new_table' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME)
```

```
VALUES ('this is an altered desc of new table', 'Altered New Table')
```

This will alter a table named "NEW_TABLE", its description is "this is an altered desc of new table", and business name is "Altered New Table".

See Also

OMBALTER, OMBCREATE TABLE, OMBDROP TABLE, OMBRETRIEVE TABLE

OMBALTER TABLE_FUNCTION

Purpose

Alter the Table Function by renaming it, and/or reset its properties, and/or resetting its Ordered/Partitioned fields,
and/or adding Parameters.

Prerequisites

Should be in the context of Oracle Module or Package. The REFCursorType and PLSQLTableType which are set as Datatype for parameters should preexist in corresponding Package.

Syntax

```
alterFunctionCommand = OMBALTER ( FUNCTION "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] [
    "alterFuncProcParameterSCOClaue" ] | "alterPropertiesOrIconSetClause"
    [ "alterFuncProcParameterSCOClaue" ] |
    "alterFuncProcParameterSCOClaue" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" | UNSET "unsetReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) | UNSET "unsetReferenceIconSetClause"
alterFuncProcParameterSCOClaue = ( ADD ( "alterFuncProcParameterClause" |
    "addRelationalDependentClause" ) | MODIFY
    "modifyFuncProcParameterClause" | DELETE (
    "deleteFuncProcParameterClause" | "deleteRelationalDependentClause" )
    ) [ "alterFuncProcParameterSCOClaue" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
alterFuncProcParameterClause = PARAMETER "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ]
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
modifyFuncProcParameterClause = ( PARAMETER "QUOTED_STRING" (
    "renameClause" | "moveToClause" | [ SET "setPropertiesClause" ] ) )
deleteFuncProcParameterClause = ( PARAMETER "QUOTED_STRING" )
deleteRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterFunctionCommand

Alters a Table Function

renameClause

Rename a Table Function

alterFuncProcParameterSCOClause

Modify, delete or add a Parameter for Function/Procedure, or add or delete dependencies to some other relational objects.

setPropertiesClause

Associate a set of properties with a Table Function.

Properties for TABLE_FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Table Function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Table Function

Name: PARALLEL_EXECUTION

Type: STRING(5)

Valid Values: TRUE, FALSE

Default: "

Enables Parallel Execution of the Table Function

Name: PIPELINED_EXECUTION

Type: STRING(5)

Valid Values: TRUE, FALSE

Default: "

Enables Partitioned Execution of the Table Function

Name: ORDER_METHOD

Type: STRING(9)

Valid Values: ORDERBY, CLUSTERBY

Default: ORDERBY

Order Method for the Table Function

Name: PARTITION_METHOD

Type: STRING(5)

Valid Values: NONE, ANY, HASH, RANGE

Default: NONE

Partition Method for the Table Function

Name: RETURN_TYPE

Type: STRING(4000)

Valid Values: N/A

Default: "

Name of the Return Type of this Table Function. For this release, this has to be a PLSQL Table Type whose datatype has to be PLSQLRecordType. Also, it should already be defined in USER_TYPES Package in this Module.

Name: IMPLEMENTATION

Type: STRING(4000)

Valid Values: N/A

Default: "

Implementation code for this Table Function.

Name: IS_DETERMINISTIC

Type: STRING(5)

Valid Values: TRUE, FALSE

Default: "

Whether this Table Function is Deterministic.

Properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the Parameter

Name: DATATYPE
Type: STRING(4000)
Valid Values: Any valid REF cursor type
Default: "
Datatype of the Parameter. Parameter will always be IN type for Table Function. For this release, it has to be a REF Cursor type. Also, this REF Cursor should already be defined in USER_TYPES Package in this Module.

Properties for TABLE_FUNCTION:

Name: AUTHID
Type: STRING
Valid Values: Current_User, Definer, None
Default: None
Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true
Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

alterFuncProcParameterClause

Alter the parameters of a table function, including adding, dropping parameters, changing their direction, datatype and default values.

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

modifyFuncProcParameterClause

Modify one or more Parameters to this Function/Procedure.

deleteFuncProcParameterClause

Delete one or more Parameters to this Function/Procedure.

deleteRelationalDependentClause

This clause deletes referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

moveToClause

Move a Parameters of this Function/Procedure.

propertyValue

Value of a property.

Examples

```
OMBALTER TABLE_FUNCTION 'table_function' RENAME TO 'table_function2' SET
PROPERTIES (PARTITION_METHOD, ORDER_METHOD) VALUES ('RANGE',
'CLUSTERBY')
```


This will rename the Table Function "table_function" to "table_function2", and set its property OrderMethod to "CLUSTERBY" and property PartitionMethod to "RANGE".

See Also

OMBALTER, OMBCREATE TABLE_FUNCTION, OMBDROP TABLE_FUNCTION

OMBALTER TIME_DIMENSION

Purpose

This command alter a time dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```

alterTimeDimensionCommand = OMBALTER TIME_DIMENSION "TimeDimensionName" (
  ( "renameDimensionClause" [ "setPropertiesClause" ] |
    "setPropertiesClause" ) | "setFiscalPropertyClause" |
  "renameMapClause" | "useSequenceClause" | ( "addDimensionRoleClause" |
  DELETE ( "deleteDimensionRoleClause" | "deleteLevelClause" ) | MODIFY
  ( "modifyDimensionRoleClause" | "modifyLevelClause" ) )+ | ( {
  "addCalendarHierarchyClause" | "modifyCalendarHierarchyClause" |
  "addFiscalCalendarHierarchyClause" |
  "modifyFiscalCalendarHierarchyClause" | "deleteHierarchyClause" } [
  "implementationClause" ] [ "populationClause" ] ) )
TimeDimensionName = "QUOTED_STRING"
renameDimensionClause = RENAME DIMENSION TO "QUOTED_STRING"
setPropertiesClause = SET PROPERTIES "propertyKeyList" VALUES
  "propertyValueList"
setFiscalPropertyClause = "setFiscalPropertiesClause"
renameMapClause = RENAME MAPPING TO "QUOTED_STRING"
useSequenceClause = SET REF SEQUENCE "QUOTED_STRING"
addDimensionRoleClause = ADD DIMENSION_ROLE "roleName" [
  "setPropertiesClause" ]
deleteDimensionRoleClause = DIMENSION_ROLE "roleName"
deleteLevelClause = "levelLocator"
modifyDimensionRoleClause = DIMENSION_ROLE "roleName" RENAME TO "roleName"
  [ "setPropertiesClause" ]
modifyLevelClause = "levelLocator" ( "renameClause" [
  "setPropertiesClause" ] )
addCalendarHierarchyClause = ADD ( ( NORMAL_CALENDAR HIERARCHY
  "hierarchyName" [ "setPropertiesClause" ] SET ( REF | REFERENCE )
  "normalCalendarLevelList" ) | ( WEEK_CALENDAR HIERARCHY
  "hierarchyName" [ "setPropertiesClause" ] SET ( REF | REFERENCE )
  "weekCalendarLevelList" ) )
modifyCalendarHierarchyClause = MODIFY ( ( NORMAL_CALENDAR
  "hierarchyLocator" ( [ "renameClause" ] [ "setPropertiesClause" ] SET
  ( REF | REFERENCE ) "normalCalendarLevelList" ) ) | ( WEEK_CALENDAR
  "hierarchyLocator" [ "renameClause" ] [ "setPropertiesClause" ] SET (
  REF | REFERENCE ) "weekCalendarLevelList" ) )
addFiscalCalendarHierarchyClause = ADD FISCAL_CALENDAR HIERARCHY
  "hierarchyName" [ "setPropertiesClause" ] SET ( REF | REFERENCE )
  "fiscalCalendarLevelList"
modifyFiscalCalendarHierarchyClause = MODIFY FISCAL_CALENDAR
  "hierarchyLocator" [ "renameClause" ] [ "setPropertiesClause" ] SET (
  REF | REFERENCE ) "fiscalCalendarLevelList"
deleteHierarchyClause = DELETE "hierarchyLocator"
implementationClause = IMPLEMENTED BY ( STAR | SNOWFLAKE ) [ USING
  COMPOSITE_UNIQUE_KEY ]
populationClause = POPULATE DATA FROM "calendarYear" FOR "yearCount" YEARS
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"

```

```

setFiscalPropertiesClause = SET FISCAL_CALENDAR PROPERTIES
    "propertyKeyList" VALUES "propertyValueList"
roleName = "QUOTED_STRING"
levelLocator = LEVEL "levelName"
renameClause = RENAME TO "QUOTED_STRING"
hierarchyName = "QUOTED_STRING"
normalCalendarLevelList = "(" ( "normalCalendarLevelType" LEVEL [
    "levelName" ] [ "setPropertiesClause" ] ) { ","
    "normalCalendarLevelType" LEVEL [ "levelName" ] [
    "setPropertiesClause" ] } ")"
weekCalendarLevelList = "(" ( "weekCalendarLevelType" LEVEL [ "levelName"
    ] [ "setPropertiesClause" ] ) { "," "weekCalendarLevelType" LEVEL [
    "levelName" ] [ "setPropertiesClause" ] } ")"
hierarchyLocator = HIERARCHY "hierarchyName"
fiscalCalendarLevelList = "(" ( "fiscalCalendarLevelType" LEVEL [
    "levelName" ] [ "setPropertiesClause" ] ) { ","
    "fiscalCalendarLevelType" LEVEL [ "levelName" ] [
    "setPropertiesClause" ] } ")"
calendarYear = "INTEGER_LITERAL"
yearCount = "INTEGER_LITERAL"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
levelName = "QUOTED_STRING"
normalCalendarLevelType = ( DAY | CALENDAR_MONTH | CALENDAR_QUARTER |
    CALENDAR_YEAR )
weekCalendarLevelType = ( DAY | CALENDAR_WEEK )
fiscalCalendarLevelType = ( DAY | FISCAL_WEEK | FISCAL_MONTH |
    FISCAL_QUARTER | FISCAL_YEAR )

```

Keywords And Parameters

TimeDimensionName

The name of the time dimension.

renameDimensionClause

This clause renames the time dimension.

setPropertiesClause

This clause sets the following properties:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

setFiscalPropertyClause
set the fiscal properties.

renameMapClause
This clause renames the map that is created by the OMBCREATE TIME_DIMENSION command.

useSequenceClause
This clause sets the Sequence.

addDimensionRoleClause
This clause adds a dimension role.

deleteDimensionRoleClause
This clause deletes a dimension role.

deleteLevelClause
This clause finds the level to be deleted.

modifyDimensionRoleClause
This clause allows to rename the dimension role, or it change the dimension role's properties.

modifyLevelClause
This clause modifies level by either renaming it, or setting level properties.

addCalendarHierarchyClause
This clause adds a new hierarchy to the time dimension by: renaming the hierarchy, setting of hierarchy properties, or setting level references.

modifyCalendarHierarchyClause
This clause modifies a hierarchy of the time dimension by: renaming the hierarchy, setting of hierarchy properties, or setting level references.

addFiscalCalendarHierarchyClause

This clause adds an fiscal hierarchy to the time dimension.

modifyFiscalCalendarHierarchyClause

This clause modifies a fiscal hierarchy of the time dimension by: renaming the fiscal hierarchy, setting of fiscal hierarchy properties, or setting fiscal level references.

implementationClause

Time Dimension is implemented as STAR or as SNOWFLAKE.

populationClause

This clause specifies the starting year and the number of years for which data will be populated.

propertyKeyList

A list of time dimension properties.

propertyValueList

A list of time dimension property values.

setFiscalPropertiesClause

This clause sets the following properties:

Fiscal types allowed in OWB time dimension. Name: FISCAL_TYPE

Type: STRING

Valid Values: '544', '445'

Default: '544'

Fiscal calendar year start date, it could be any date of a year.

Name: FISCAL_CALENDAR_START_YEAR

Type: STRING

Valid Values: Dates in these format 'DD-MON-YYYY' or 'DD-MM-YYYY'

Default: '01-JAN-2000'

The day of the week when the fiscal year begins.

Name: FISCAL_CALENDAR_START_DAY_OF_WEEK

Type: STRING

Valid Values: 'MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY',

'SATURDAY', 'SUNDAY'

Default: 'false'

roleName

A role name.

levelLocator

This clause gets the level.

hierarchyName

The name of a hierarchy.

hierarchyLocator

This clause gets the hierarchy.

fiscalCalendarLevelList

This clause creates a fiscal hierarchy and sets reference fiscal levels.

propertyKey

Basic properties for TIME DIMENSION, TIME DIMENSION MAP,
DIMENSION_ATTRIBUTE, LEVEL, LEVEL_ATTRIBUTE and HIERARCHY:

Basic properties for TIME DIMENSION :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Time Dimension

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Time Dimension

Name: STORAGE

Type: STRING

Valid Values: 'RELATIONAL', 'AW'

Default: 'RELATIONAL'

The storage of a dimension can be AW or relational

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the analytical workspace name where the dimension is implemented

Name: AW_DIMENSION_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace dimension physical object name

Basic properties for TIME MAP :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Time Dimension Map

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Time Dimension Map

Basic properties for DIMENSION_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension_Attribute

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "

Properties for DIMENSION:

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS
Type: STRING
Valid Values: DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY, DEPLOY_TO_CATALOG_ONLY
Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for Dimension, they are DDL Scripts for Relational Dimensional or Scripts for ROLAP or or Scripts for AW.

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "

Enter additional comments for the generated code.

Name: VIEW_NAME
Type: STRING(30)
Valid Values: N/A
Default: "

Name of the view that is generated to hide the control rows on the dimension implementation table of a star schema. If this field is left

blank, the view name will default to '<Name of Dimension>_v'

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Dimension is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

Examples

We assume that the time dimension 'FYR2005' already exists ...

```
OMBALTER TIME_DIMENSION 'FYR2005'
```

```
SET FISCAL_CALENDAR PROPERTIES ( FISCAL_TYPE, FISCAL_CALENDAR_
START_YEAR,
```

```
FISCAL_CALENDAR_START_DAY_OF_WEEK )
```

```
VALUES ( '544', '01-01-2000', 'MONDAY' )
```

```
ADD FISCAL_CALENDAR HIERARCHY 'FCALH1'
```

```
SET PROPERTIES ( DESCRIPTION, BUSINESS_NAME )
```

```
VALUES ('TimeSeries Hierarchy Description 1', 'TIME DIMENSION HIERARCHY
FSTAR1 FCALH1')
```

```
DELETE FISCAL_CALENDAR HIERARCHY 'FCALH0'
```

```
MODIFY FISCAL_CALENDAR HIERARCHY 'FCALH3'
```

```
SET PROPERTIES ( DESCRIPTION, BUSINESS_NAME )
```

```
VALUES ('TimeSeries Hierarchy Description 1', 'TIME DIMENSION HIERARCHY
FSTAR1 FCALH1')
```

```
SET REF ( DAY LEVEL 'MY_DAY' , FISCAL_YEAR LEVEL 'MY_FISCAL_YEAR' )
```

```
ADD DIMENSION_ROLE 'FISCAL_2005_SHIPMENTS'
```

```
IMPLEMENTED BY STAR
```

```
ADD REF SEQUENCE 'YR2005_1'
```

```
POPULATE DATA FROM 2000 FOR 2 YEARS.
```

```
OMBALTER TIME_DIMENSION 'FYR2005'
```

```
SET PROPERTIES ( DESCRIPTION, BUSINESS_NAME )
```

```
VALUES ('TimeSeries Description 2', 'TIME SERIES 2')
```

See Also

OMBCREATE TIME_DIMENSION, OMBDROP TIME_DIMENSION, OMBRETRIEVE TIME_DIMENSION

OMBALTER TRANSPORTABLE_MODULE

Purpose

To alter the definition of a transportable module.

Prerequisites

In the context of a project.

Syntax

```
alterTMCommand = OMBALTER TRANSPORTABLE_MODULE "QUOTED_STRING" (
    "alterTMClause" { "alterTMClause" } )
alterTMClause = "renameClause" | "setPropertiesAndLocationsAndIconSet" |
    "unsetReferenceIconSetClause" | "modifyClause"
renameClause = RENAME TO "QUOTED_STRING"
setPropertiesAndLocationsAndIconSet = SET ( "setPropertiesClause" |
    "setSourceLocationClause" | "setTargetLocationClause" |
    "setReferenceIconSetClause" )
unsetReferenceIconSetClause = UNSET ( REF | REFERENCE ) ICONSET
modifyClause = MODIFY ( "modifyTablespaceClause" | "modifyDatafileClause"
    | "modifySchemaClause" )
setPropertiesClause = ( PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")" )
setSourceLocationClause = SOURCE_LOCATION "QUOTED_STRING"
setTargetLocationClause = TARGET_LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REF | REFERENCE ) ICONSET "QUOTED_STRING"
modifyTablespaceClause = TRANSPORTABLE_MODULE_TABLESPACE "QUOTED_STRING"
modifyDatafileClause = DATAFILE "QUOTED_STRING"
modifySchemaClause = TRANSPORTABLE_MODULE_SCHEMA "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterTMCommand

This command is for altering a transportable module.

QUOTED_STRING

The name of the transportable module to be altered.

alterTMClause

Multiple altering actions can be specified with one OMBALTER TRANSPORTABLE_MODULE command.

renameClause

Change the name of the transportable module

QUOTED_STRING

The new name for the transportable module.

setPropertyAndLocationsAndIconSet

Set properties for the transportable module, and/or specify source and target locations, and/or specify icon set for the newly create transportable module.

unsetReferenceIconSetClause

Remove the reference to the icon set.

modifyClause

Change contents within a transportable module.

setPropertyClause

Set properties for the transportable module.

Basic properties for TRANSPORTABLE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of the transportable module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description for the transportable module

Properties for TRANSPORTABLE_MODULE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: TARGET_OS_TYPE

Type: STRING

Valid Values: Linux, Unix, Windows

Default: Unix

The operating system type of the target machine. This is needed for generating shell scripts in correct style required by the operating system.

Name: TRANSPORT_TABLESPACE

Type: BOOLEAN

Valid Values: true, false

Default: true

Specifies whether transportable tablespace (TTS) feature is to be used for deploying tables in the transportable module. If set to true, tablespaces are copied from source to target using the server TTS mechanism. If set to false, tables are individually extracted and deployed using Oracle Data Pump available in Oracle Database 10g or later; but tablespaces are not transported. Since Oracle Data Pump is new in Oracle Database 10g, setting this parameter to false is only allowed if both source and target databases are with Oracle 10g or a higher versions.

Name: WHAT_TO_DEPLOY

Type: STRING

Valid Values: ALL_OBJECTS, TABLES_ONLY

Default: ALL_OBJECTS

Specifies whether only tables in the transportable module are deployed or everything in it is deployed.

Name: WORK_DIRECTORY

Type: STRING

Valid Values: N/A

Default: "

The full path of work directory on target machine, where temporary files, logs and tablespace datafiles may be stored. If left unspecified, OWB's runtime home directory is used as the work directory. It is highly recommended that users specify dedicated directory for transportable module

deployment.

Properties for TRANSPORTABLE_MODULE_TABLESPACE:

Name: DROP_EXISTING_TABLESPACE

Type: BOOLEAN

Valid Values: true, false

Default: false

If this parameter is set to true, existing tablespace in target is dropped and re-created.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: TARGET_TABLESPACE_NAME

Type: STRING

Valid Values: N/A

Default: DEFAULT

Tablespace name in the target database. Target tablespace name must be same with source tablespace name prior to Oracle 10.2.

Properties for TRANSPORTABLE_MODULE_DATAFILE:

Name: DIRECTORY

Type: STRING

Valid Values: N/A

Default: "

The directory on target machine where the datafile will be created. If left unspecified, the target work directory is used for storing the datafile that is transported from source machine.

Name: FILENAME

Type: STRING

Valid Values: N/A

Default: DEFAULT

The name of the new file on target. Please check to see if there is already a file with same name in the same directory. Transportable tablespace deployment may overwrite any existing files.

Name: REUSE

Type: BOOLEAN

Valid Values: true, false

Default: false

If this parameter is set to true, existing datafile is overwritten. If set to false, overwriting existing file is not allowed. In this case, if an file exists, deployment will terminate.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setSourceLocationClause

Specify the source location name.

QUOTED_STRING

The name of an already created transportable module source location.

setTargetLocationClause

Specify the target location name.

QUOTED_STRING

The name of an already created transportable module target location.

setReferenceIconSetClause

Set the icon set for the new transportable module.

QUOTED_STRING

The name of the icon set.

modifyTablespaceClause

Modify a tablespace within a transportable module.

QUOTED_STRING

The tablespace name. Note that the tablespace name is its name in the source database.

modifyDatafileClause

Modify a datafile within a transportable module.

QUOTED_STRING

The name of the datafile. Note that the datafile name is its full path name in the source database. The name must be exactly same with what is in DBA_DATA_FILES view in the source database.

modifySchemaClause

Modify a schema within a transportable module.

QUOTED_STRING

The name of the schema. Note that the schema name is its name in the source database.

propertyNameList

The list of unquoted property names.

propertyValueList

The list of property values.

propertyValue

A property value can be a single-quoted string, an integer, or a floating point number.

Examples

```
OMBALTER TRANSPORTABLE_MODULE 'TM101'  
SET SOURCE_LOCATION 'TM_SRC_LOC'  
SET TARGET_LOCATION 'TM_TGT_LOC2'
```



```
SET PROPERTIES (WORKING_DIRECTORY, TARGET_OS_TYPE, WHAT_TO_
DEPLOY,
TRANSPORT_TABLESPACE)
```

```
VALUES ('d:mydir', 'Windows', 'TABLES_ONLY', 'true')
```

This example changes the source and target locations and the four properties associated with transportable module.

```
OMBALTER TRANSPORTABLE_MODULE 'TM101'
MODIFY TRANSPORTABLE_MODULE_TABLESPACE 'src_tablespace_1'
SET PROPERTIES (TARGET_TABLESPACE_NAME, DROP_EXISTING_
TABLESPACE)
VALUES ('tgt_tablespace_1', 'true');
```

This example changes properties of a tablespace within the transportable module. The properties specify the target tablespace name, and whether to drop and re-create target tablespace if it already exists.

```
OMBALTER TRANSPORTABLE_MODULE 'TM101'
MODIFY DATAFILE 'D:\TTSFILES\TTS1B.DBF'
SET PROPERTIES (DIRECTORY, FILENAME, REUSE)
VALUES ('d:\tmdir', 'TM1B.DBF', 'false')
```

This example changes properties of a datafile within a tablespace. The properties specify the target datafile's directory and name, plus a flag to indicate whether to overwrite existing file. Notice that the source datafile name is the full path of the file.

```
OMBALTER TRANSPORTABLE_MODULE 'TM101'
MODIFY TRANSPORTABLE_MODULE_SCHEMA 'src_schema_1'
SET PROPERTIES (SCHEMA_NAME, PASSWORD, DEFAULT_TABLESPACE,
SCHEMA_EXISTS_ACTION, SCHEMA_DOESNT_EXIST_ACTION,
TABLE_EXISTS_ACTION, COPY_SOURCE_SCHEMA, PARALLEL)
VALUES ('TM1_U', 'TM1_U', 'USERS',
'Replace', 'Create',
'Append', 'false', 2)
```

This example changes the properties of a schema within a transportable module. The properties specify target schema name, password and default tablespace, plus some important choices. In this example, target schema will be replaced if it already exists. If it does not exist, it will be created. If a table already exists in target, new data will be appended to

the existing table. The transportable module will not copy the entire schema from source to target. And finally, the degree of parallelism is 2.

See Also

OMBALTER, OMBCREATE TRANSPORTABLE_MODULE, OMBRETRIEVE TRANSPORTABLE_MODULE, OMBDROP TRANSPORTABLE_MODULE

OMBALTER USER

Purpose

To alter properties of a Warehouse Builder user.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
alterUserCommand = OMBALTER ( USER "QUOTED_STRING" ( SET
    "setPropertyClause" ) )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

alterUserCommand

This clause alters a Warehouse Builder user.

setPropertyClause

Used to set properties of a Warehouse Builder user. Valid properties are as shown:

Basic properties for USER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the User

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the User

Name: ISTARGETSCHEMA

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the user will be set up as target schema for deployment; and also the property TARGETSCHEMAPWD must be provided when you are setting the ISTARGETSCHEMA as true.

Name: TARGETSCHEMAPWD

Type: STRING(30)

Valid Values: N/A

Default: N/A

This properties will be provided only when you are seting ISTARGETSCHEMA as true, so that the necessary target schema objects can be installed into the potential target schema. And this property cannot be retrieved due to security consideration.

User preferences:

Name: LOCALE

Type: STRING

Valid Values: Albanian, Arabic, Bulgarian, Byelorussian, Catalan, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, French, German, Greek, Hebrew, Hungarian, Icelandic, Italian, Japanese, Korean, Lithuanian, Macedonian, Norwegian, Polish, Portuguese, Romanian, Russian, Serbian, Serbo_Croatian, Slovak, Slovenian, Spanish, Swedish, Thai, Turkish, Ukranian

Default: "

Name: SHOW_PROJECT

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_MODULE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_LOCATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_ACTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_TYPE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: DEFAULT_PROFILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Name: ALLOW_UNDO_REDO

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PAUSE_AFTER_COMPILE

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_COMMIT

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_JOB_NAME

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_EXECUTION_PARAMS

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_MONITOR

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DEPLOYMENT_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DEPLOYMENT_DEPENDENCIES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_MONITOR_RESULTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_MONITOR_LOGFILE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: PERSONALITY

Type: STRING

Valid Values: N/A

Default: Default

Name: SHOW_GUIDED_ASSISTANCE

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: HIDE_WIZARD_WELCOME_PAGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DELETE_CONFIRMATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: RECYCLE_DELETED_OBJECTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: EMPTY_RECYCLE_BIN

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: CLEAR_CLIPBOARD

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_PROJECT

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_MODULE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_LOCATION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_GENERATION_ACTION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_GENERATION_TYPE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_FILE_PATH

Type: STRING(1000)

Valid Values: N/A

Default: "

Name: LOG_FILE_NAME

Type: STRING(1000)

Valid Values: N/A

Default: log

Name: LOG_FILE_MAX_SIZE

Type: STRING

Valid Values: 1-10000000

Default: 100

Name: LOG_ERROR_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_WARNING_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_INFORMATION_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: NAMING_MODE

Type: STRING

Valid Values: PHYSICAL_NAMING_MODE, BUSINESS_NAMING_MODE

Default: PHYSICAL_NAMING_MODE

Name: PROPAGATE_NAME_CHANGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: DESIGNREPOS_PWD_PERSIST

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: RUNTIMEREPOS_PWD_SHARE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: DEFAULT_SEC_POLICY

Type: STRING

Valid Values: MINIMUM_SECURITY, MAXIMUM_SECURITY

Default: MINIMUM_SECURITY

Examples

```
OMBALTER USER 'USER1' SET PROPERTIES(BUSINESS_NAME, DESCRIPTION,  
ISTARGETSCHEMA,TARGETSCHEMA) VALUES('developer user1 changed', 'one  
user  
from developer group', 'true','dbPwdOfUser1')
```

See Also

OMBREGISTER USER, OMBUNREGISTER USER, OMBRETRIEVE USER

OMBALTER VARYING_ARRAY

Purpose

Alter the Varying Array by resetting its properties.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
alterVaryingArrayCommand = OMBALTER ( VARYING_ARRAY "QUOTED_STRING" (
    "renameClause" [ "alterPropertiesOrIconSetClause" ] |
    "alterPropertiesOrIconSetClause" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

renameClause

renames a table with a different name.

setPropertiesClause

Basic properties for VARYING_ARRAY:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Varying Array

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Varying Array

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Base Element of the Varying Array

Properties for VARYING_ARRAY:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBALTER VARYING_ARRAY 'SOME_VARRAY' SET PROPERTIES  
(DESCRIPTION,ARRAY_LENGTH) VALUES ('This is a new description.',10)
```

This will set its description to "This is a new description." and its array length to 10.

See Also

OMBALTER, OMBCREATE VARYING_ARRAY, OMBDROP VARYING_ARRAY

OMBALTER VIEW

Purpose

To alter properties and definition of a view.

Prerequisites

In the context of an Oracle Module.

Syntax

```
alterViewCommand = OMBALTER ( VIEW "QUOTED_STRING" ( "renameClause" [
    "alterPropertiesOrIconSetClause" ] [ "alterViewSCOandDependentClauses"
    ] | "alterPropertiesOrIconSetClause" [
    "alterViewSCOandDependentClauses" ] |
    "alterViewSCOandDependentClauses" ) )
renameClause = RENAME TO "QUOTED_STRING"
alterPropertiesOrIconSetClause = SET ( "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferenceIconSetClause" | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) | UNSET ( REF | REFERENCE )
    "unsetReferenceIconSetClause"
alterViewSCOandDependentClauses = ADD ( "addColumnClauseForAlter" [
    "alterViewSCOandDependentClauses" ] | "addViewConstraintClause" {
    "alterViewConstraintClauses" } | "addDataRuleUsageClause" {
    "alterDataRuleUsageClauses" } | "addRelationalDependentClause" [
    "alterViewSCOandDependentClauses" ] ) | MODIFY ( "modifyColumnClause"
    [ "alterViewSCOandDependentClauses" ] | "modifyViewConstraintClause" {
    "alterViewConstraintClauses" } | "modifyDataRuleUsageClause" {
    "alterDataRuleUsageClauses" } ) | DELETE ( "deleteColumnClause" [
    "alterViewSCOandDependentClauses" ] | "deleteViewConstraintClause" {
    "alterViewConstraintClauses" } | "deleteDataRuleUsageClause" {
    "alterDataRuleUsageClauses" } | "deleteRelationalDependentClause" [
    "alterViewSCOandDependentClauses" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ICONSET
addColumnClauseForAlter = COLUMN "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setPropertiesClause" ]
addViewConstraintClause = "addUkPkClause" | "addFkClause"
alterViewConstraintClauses = ADD "addViewConstraintClause" | MODIFY
    "modifyViewConstraintClause" | DELETE "deleteViewConstraintClause"
addDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" SET REF DATA_RULE
    "QUOTED_STRING" ( GROUP "QUOTED_STRING" SET REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) "QUOTED_STRING" ( ATTRIBUTE
    "QUOTED_STRING" SET REF COLUMN "QUOTED_STRING" )+ )+ [ SET
    "setPropertiesClause" ]
alterDataRuleUsageClauses = ADD "addDataRuleUsageClause" | MODIFY
    "modifyDataRuleUsageClause" | DELETE "deleteDataRuleUsageClause"
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING"
modifyColumnClause = COLUMN "QUOTED_STRING" ( "renameClause" [
    "moveToClause" ] [ SET "setPropertiesClause" ] | "moveToClause" [ SET
    "setPropertiesClause" ] | SET "setPropertiesClause" )
modifyViewConstraintClause = "modifyUkPkClause" | "modifyFkClause"
modifyDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" (
    "renameClause" [ SET "setPropertiesClause" ] | SET
```

```

"setPropertiesClause" )
deleteColumnClause = COLUMN "QUOTED_STRING"
deleteViewConstraintClause = UNIQUE_KEY "QUOTED_STRING" | PRIMARY_KEY
"QUOTED_STRING" | FOREIGN_KEY "QUOTED_STRING"
deleteDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING"
deleteRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
MATERIALIZED_VIEW ) "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" [ SET
"setUkPkPropertiesAndReferencesColumnsClauses" ]
addFkClause = FOREIGN_KEY "QUOTED_STRING" [ SET "setFkSubClauses" ]
moveToClause = MOVE TO POSITION "INTEGER_LITERAL"
modifyUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" (
"renameClause" [ SET "setUkPkPropertiesAndReferencesColumnsClauses" ]
| SET "setUkPkPropertiesAndReferencesColumnsClauses" )
modifyFkClause = FOREIGN_KEY "QUOTED_STRING" ( "renameClause" [ SET
"setFkSubClauses" ] | SET "setFkSubClauses" )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
"FLOATING_POINT_LITERAL" )
setUkPkPropertiesAndReferencesColumnsClauses = "setPropertiesClause" [ SET
( REF | REFERENCE ) "constraintColumnReferencesClause" ] | ( REF |
REFERENCE ) "constraintColumnReferencesClause"
setFkSubClauses = "setPropertiesClause" [ SET ( REF | REFERENCE )
"setFkReferencesClauses" ] | ( REF | REFERENCE )
"setFkReferencesClauses"
constraintColumnReferencesClause = COLUMNS "(" "quotedNameList" ")"
setFkReferencesClauses = "constraintColumnReferencesClause" [ SET ( REF |
REFERENCE ) "constraintUkReferencesClause" ] |
"constraintUkReferencesClause" [ SET ( REF | REFERENCE )
"constraintColumnReferencesClause" ]
quotedNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
constraintUkReferencesClause = ( UNIQUE_KEY | PRIMARY_KEY )
"QUOTED_STRING" [ OF ( TABLE | VIEW ) "QUOTED_STRING" ]

```

Keywords And Parameters

alterViewCommand

This clause alters a view.

QUOTED_STRING

name of the view.

renameClause

renames a table with a different name.

alterViewSCOandDependentClauses

This clause alters the view clause.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for views

and their columns, unique keys, foreign keys, and primary keys.

Note:

Constraints can be specified but will not be generated for either View or Materialized View in this release.

Basic properties for VIEW, MATERIALIZED_VIEW:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the View, MaterializedView

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the View, MaterializedView

Name: VIEW_QUERY

Type: STRING(4000)

Valid Values: N/A

Default: "

Sets the query definition in View and MaterializedView.

Properties for VIEW:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for

those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE

constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps

the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addColumnClauseForAlter

This clause adds a column at a particular position.

When you alter a table and add columns to it, the position you specify for a new column must be less than or equal to the number of columns added up to that point in the OMBALTER command.

For example, a table TEMP_TAB contains three columns. You use the following

OMBALTER TABLE command to add three more columns:

```
OMBALTER TABLE 'TEMP_TAB' \  
ADD COLUMN 'C4' AT POSITION 4 \  
SET PROPERTIES(DATATYPE,PRECISION) VALUES('NUMBER',7) \  
ADD COLUMN 'C5' AT POSITION 6 \  
SET PROPERTIES(DATATYPE) VALUES('VARCHAR2') \  

```

```
ADD COLUMN 'C6' AT POSITION 5 \
SET PROPERTIES(DATATYPE,PRECISION) VALUES('NUMBER',10);
```

This command does not execute successfully because at the point when you specify the position of the column C5 as 6, the table contains only 5 columns.

QUOTED_STRING

The column name.

addViewConstraintClause

This clause adds the view's configuration clause.

alterViewConstraintClauses

This clause alters the view's constraint clause.

addDataRuleUsageClause

Add a data rule usage to the relation.

alterDataRuleUsageClauses

Add, modify, or delete data rule usages.

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

modifyColumnClause

This clause renames, set properties, and move columns.

modifyViewConstraintClause

This clause modifies the view's constraint clause.

modifyDataRuleUsageClause

Rename or modify the properties of a data rule usage.

deleteColumnClause

This clause deletes a column.

`deleteViewConstraintClause`

This clause deletes the view's constraint.

`deleteDataRuleUsageClause`

Delete a data rule usage.

`deleteRelationalDependentClause`

This clause deletes referential dependencies to other relational objects.

`propertyNameList`

The list of properties.

`propertyValueList`

The list of property values.

`addUkPkClause`

This clause adds the adds unique key and primary keys.

`QUOTED_STRING`

name of the unique key or primary key.

`addFkClause`

This clause adds foreign key.

`QUOTED_STRING`

Name of the foreign key.

`moveToClause`

This clause will move the column to given position.

`modifyUkPkClause`

It modifies unique or primary key.

`modifyFkClause`

This clause modifies the foreign key.

propertyValue

This clause adds the property values.

setUkPkPropertiesAndReferencesColumnsClauses

This clause adds properties and references to columns.

setFkSubClauses

This clause set references to a foreign key.

constraintColumnReferencesClause

This clause provides names of all columns.

setFkReferencesClauses

This clause sets foreign key references.

quotedNameList

This clause gives column names.

constraintUkReferencesClause

The first QUOTED_STRING denotes the UniqueKey or Primary key name, and the latter denotes the table's or view's name.

Examples

```
OMBALTER VIEW 'NEW_VIEW' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME) VALUES
```

```
('this is an altered desc of new view', 'Altered New View')
```

This will alter a view named "NEW_VIEW", its description is "this is an altered desc of new view", and business name is "Altered New View".

See Also

OMBALTER, OMBCREATE VIEW, OMBDROP VIEW, OMBRETRIEVE VIEW

OMBCREATE to OMBCREATE PLSQL_ TABLE_TYPE

This chapter lists commands associated with OMBCREATE in alphabetical order, concluding with the command OMBCREATE PLSQL_TABLE_TYPE. Subsequent commands associated with OMBCREATE are contained in the next chapter.

OMBCREATE

Purpose

Create a new component.

Prerequisites

Should be in the context that stores the component type.

Syntax

```
createCommand = OMBCREATE [ TRANSIENT ] "fco_type" "fco_name" [  
    "setPropertiesClause" ] { "setReferenceClause" } { "addSCOClauses" }  
setPropertiesClause = SET PROPERTIES "(" "propertyNameList" ")" VALUES "("  
    "propertyValueList" ")"  
setReferenceClause = SET ( REF | REFERENCE ) [ "qualifier" ] "type"  
    "quotedNameList" [ { "parentSCOClauses" } OF "fco_type" "fco_name" ]  
addSCOClauses = ADD "sco_type" "sco_name" { "parentSCOClauses" } [  
    "setPropertiesClause" ] { "setReferenceClause" }  
propertyNameList = "propertyName" { "," "propertyName" }  
propertyValueList = "propertyValue" { "," "propertyValue" }  
quotedNameList = "QUOTED_STRING" | "(" "QUOTED_STRING" { "," "  
    "QUOTED_STRING" } ")"  
parentSCOClauses = OF "sco_type" "sco_name"  
propertyName = "UNQUOTED_STRING"  
propertyValue = "QUOTED_STRING" | "INTEGER_LITERAL" |  
    "FLOATING_POINT_LITERAL"
```

Keywords And Parameters

`createCommand`

Specify metadata for a new component.

`TRANSIENT`

Keyword used to specify that the object being created will not be persisted in the repository.

`fco_type`

The type of the component.

`fco_name`

The physical name of the component in single quotes.

`setPropertiesClause`

Set object properties.

`setReferenceClause`

Specify reference objects.

qualifier

Specify which reference to set, if there are more than one pointing to the same type.

addSCOClause

Add a new child object to the component.

propertyNameList

A list of property names.

propertyValueList

A list of property values.

quotedNameList

A list of single-quoted physical names.

parentSCOClause

Used to specify the path from a child object to the component

propertyName

An unquoted string representing the name of a property.

propertyValue

The value of a property.

Examples

This is an example for creating an empty table:

```
OMBCREATE TABLE 'T1' SET PROPERTIES (DESCRIPTION) VALUES ('My First  
Table')
```

The following statement creates a view with a column:

```
OMBCREATE VIEW 'V1'  
ADD COLUMN 'COL1'  
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2')
```

See Also

OMBALTER, OMBDROP

OMBCREATE ACTIVITY_TEMPLATE

Purpose

To create an activity template.

Valid Activity Template types are:

ASSIGN

EMAIL

FILE_EXISTS

FTP

MANUAL

NOTIFICATION

SET_STATUS

SQLPLUS

USER_DEFINED

WAIT

Prerequisites

Should be in the context of an Activity Template Folder.

Syntax

```
createActivityTemplate = ( OMBCREATE ACTIVITY_TEMPLATE "QUOTED_STRING" [ (
    OF ACTIVITY_TYPE "UNQUOTED_STRING" ) ] { ( SET "setPropertiesClause"
    ) | "setReferenceIconSetClause" } { "addParameterClause" } )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = SET ( REFERENCE | REF ) ICONSET
    "QUOTED_STRING"
addParameterClause = ( ADD PARAMETER "QUOTED_STRING" ) [ SET
    "setPropertiesClause" ]
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

setPropertiesClause

Basic properties for ACTIVITY_TEMPLATE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Activity Template

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Activity Template

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBCREATE ACTIVITY_TEMPLATE 'ASSIGN_TEMPLATE' OF ACTIVITY_TYPE
ASSIGN
```

```
OMBCREATE ACTIVITY_TEMPLATE 'EMAIL_TEMPLATE' OF ACTIVITY_TYPE
EMAIL
```

```
OMBCREATE ACTIVITY_TEMPLATE 'FILE_EXISTS_TEMPLATE' OF ACTIVITY_
TYPE
```

```
FILE_EXISTS
```

```
OMBCREATE ACTIVITY_TEMPLATE 'FTP_TEMPLATE' OF ACTIVITY_TYPE FTP
```

```
OMBCREATE ACTIVITY_TEMPLATE 'MANUAL_TEMPLATE' OF ACTIVITY_TYPE
MANUAL
```

```
OMBCREATE ACTIVITY_TEMPLATE 'NOTIFICATION_TEMPLATE' OF ACTIVITY_
TYPE
```

```
NOTIFICATION
```

```
OMBCREATE ACTIVITY_TEMPLATE 'SET_STATUS_TEMPLATE' OF ACTIVITY_
TYPE
```

```
SET_STATUS
```

```
OMBCREATE ACTIVITY_TEMPLATE 'SQLPLUS_TEMPLATE' OF ACTIVITY_TYPE
SQLPLUS
```

```
OMBCREATE ACTIVITY_TEMPLATE 'USER_DEFINED_TEMPLATE' OF ACTIVITY_
TYPE
```

```
USER_DEFINED
```

```
OMBCREATE ACTIVITY_TEMPLATE 'WAIT_TEMPLATE' OF ACTIVITY_TYPE
WAIT
```

See Also

OMBCREATE, OMBALTER ACTIVITY_TEMPLATE, OMBDROP ACTIVITY_TEMPLATE, OMBRETRIEVE ACTIVITY_TEMPLATE

OMBCREATE ACTIVITY_TEMPLATE_FOLDER

Purpose

To create an activity template folder.

Prerequisites

Should be in the context of a Project.

Syntax

```
createActivityTemplateFolder = OMBCREATE ACTIVITY_TEMPLATE_FOLDER
    "QUOTED_STRING" { ( SET "setPropertiesClause" ) |
    "setReferenceIconSetClause" }
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = SET ( REFERENCE | REF ) ICONSET
    "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

setPropertiesClause

Basic properties for ACTIVITY_TEMPLATE_FOLDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Activity Template Folder

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Activity Template Folder

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBCREATE ACTIVITY_TEMPLATE_FOLDER 'FOLDER1'
```

See Also

OMBCREATE, OMBALTER ACTIVITY_TEMPLATE_FOLDER, OMBDROP
ACTIVITY_TEMPLATE_FOLDER, OMBRETRIEVE ACTIVITY_TEMPLATE_FOLDER

OMBCREATE ADVANCED_QUEUE

Purpose

To create an Advanced Queue.

Prerequisites

Should be in the context of an Oracle Module. The Queue Table should exist in the same Oracle Module.

Syntax

```
createAQCommand = OMBCREATE ( ADVANCED_QUEUE "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createAQCommand

Creates an Advanced Queue with the given name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Advanced Queue.

Valid properties are as shown:

Basic properties for ADVANCED_QUEUE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Advanced Queue

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Advanced Queue

Name: QTABLE

Type: STRING(4000)

Valid Values: N/A

Default: "

Queue Table for the Advanced Queue. This has to be the name of a Queue Table(Queue_Table) existing in the same Oracle Module.

Properties for ADVANCED_QUEUE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DEQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Dequeue Enabled for AQ

Name: ENQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enqueue enabled for AQ

Name: GENERATE_ADVANCED_QUEUE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the Advanced Queue.

Name: GENERATE_TEMPORARY_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the temporary table.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MAX_RETRIES

Type: NUMBER

Valid Values: N/A

Default: 5

Max. number of Retries

Name: RETENTION_TIME

Type: NUMBER

Valid Values: N/A

Default: 0

Message retention time

Name: RETRY_DELAY

Type: NUMBER

Valid Values: N/A

Default: 0

Delay period before retry

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

setReferenceIconSetClause

Set the specified Icon Set.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE ADVANCED_QUEUE 'NEW_ADVANCED_QUEUE' SET PROPERTIES  
(DESCRIPTION,  
QTABLE) VALUES ('this is an Advanced Queue', 'SOME_OBJECT_TYPE')  
This will create an Advanced Queue named "NEW_ADVANCED_QUEUE", its  
description is "this is an Advanced Queue" and its Queue Table  
'SOME_QUEUE_TABLE'.
```

See Also

```
OMBCREATE ADVANCED_QUEUE, OMBALTER ADVANCED_QUEUE, OMBDROP  
ADVANCED_QUEUE
```

OMBCREATE ALTERNATIVE_SORT_ORDER

Purpose

Creates an Alternative Sort Order.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createAlternativeSortOrderCommand = OMBCREATE ALTERNATIVE_SORT_ORDER
  "QUOTED_STRING" [ SET "setPropertiesClauseforLOVandD2D" ] [ SET
  "setReferenceIconSetClause" ] [ "addAlternativeSortOrderDefClause" ] [
  "addAlternativeSortOrderOrdClause" ] {
  "addAlternativeSortOrderReferenceClause" }
setPropertiesClauseforLOVandD2D = PROPERTIES "("
  "propertyNameListforLOVandD2D" ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addAlternativeSortOrderDefClause = SET ( REF | REFERENCE ) DEFINING ITEM
  "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
addAlternativeSortOrderOrdClause = SET ( REF | REFERENCE ) ORDERED ITEM
  "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
addAlternativeSortOrderReferenceClause = SET ( REF | REFERENCE ) ITEM
  "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
propertyNameListforLOVandD2D = ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) {
  ", " ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) }
propertyValueList = "propertyValue" { ", " "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createAlternativeSortOrderCommand

This command creates an alternative sort order.

QUOTED_STRING

Specify the name of the alternative sort order to be created.

setPropertiesClauseforLOVandD2D

Used to set properties (core, logical, physical, user-defined) for an alternative sort order. Valid properties are as shown:

Basic properties for ALTERNATIVE_SORT_ORDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the alternative sort order

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the alternative sort order

Name: DRILL_TO_DETAIL

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the alternative sort order enables drilling between the item folders containing the items that use the alternative sort order

Name: RETRIEVE_VALUES_GROUP_SIZE

Type: Number

Valid Values: N/A

Default: '100'

The number of rows to be fetched from the database at a time

Name: CACHE_VALUES

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the list of values should be cached in memory

Name: REQUIRE_SEARCH

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether search criteria should be requested

Name: SHOW_IN_NAVIGATOR

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be shown in the item navigator

Name: SORTED_DISTINCT

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be displayed sorted with duplicates hidden

Properties for ALTERNATIVE_SORT_ORDER:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

addAlternativeSortOrderDefClause

This clause adds a defining item.

addAlternativeSortOrderOrdClause

This clause adds an ordering item.

addAlternativeSortOrderReferenceClause

This adds a reference to an item to an alternative sort order.

propertyNameListforLOVandD2D

This is the list of property names.

propertyValueList

This is the list of property values.

propertyValue

This is a property value.

Examples

```
OMBCREATE ALTERNATIVE_SORT_ORDER 'SALES_ITEM'
```

See Also

OMBALTER ALTERNATIVE_SORT_ORDER, OMBRETRIEVE ALTERNATIVE_SORT_ORDER

OMBCREATE ANALYZE_ACTION_PLAN

Purpose

Create an action plan for executing a profile.

Prerequisites

First make sure that all the sources you want to profile have been imported into the OWB repository.

Syntax

```
CreateActionPlanCommand = ( OMBCREATE TRANSIENT ( ( DEPLOYMENT_ACTION_PLAN
| ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" { "addActionClause" }
)
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
"setReferenceClause"
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
"propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
"useClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
"FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

CreateActionPlanCommand

Create a profile action plan.

QUOTED_STRING

Action plan name.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Action name.

setClause

Set the properties of an action and/or associate an object with an action.

propertiesClause

Set the properties and/or associate/disassociate an object with an action.

setReferenceClause

Associate an object with an action.

ObjType

Object type. The only valid value is DATA_PROFILE_TABLE.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Currently used only in CORRECTION_SCHEMA_ACTION_PLAN.

propertyValue

Value of a property.

Examples

```
OMBCREATE TRANSIENT ANALYZE_ACTION_PLAN 'PROFILE_PLAN'  
ADD ACTION 'ACTION1'  
SET REF PROFILE_REFERENCE 'LOC'  
ADD ACTION 'ACTION2'  
SET REF PROFILE_REFERENCE 'EMP'  
ADD ACTION 'ACTION3'  
SET REF PROFILE_REFERENCE 'DEPT'  
OMBPROFILE ANALYZE_ACTION_PLAN 'PROFILE_PLAN'
```

See Also

OMBPROFILE

OMBCREATE BUSINESS_AREA

Purpose

Creates a Business Area to group business related items.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createBusinessAreaCommand = ( OMBCREATE BUSINESS_AREA "QUOTED_STRING" [
    SET "setPropertiesClause" ] [ SET "setReferenceIconSetClause" ] [ SET
    ( REF | REFERENCE ) "(" "setfolderNameList" ")" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
setfolderNameList = "QUOTED_STRING" { ",", "QUOTED_STRING" }
propertyNameList = "UNQUOTED_STRING" { ",", "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ",", "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createBusinessAreaCommand

This command creates a business area.

QUOTED_STRING

Specify the name of the business area to be created.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for business areas. Valid properties are as shown:

Basic properties for BUSINESS_AREA:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the business area

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the business area

Properties for BUSINESS_AREA:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

setfolderNameList

Used to set item folder references for this business area.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

propertyValue

This is a property value.

Examples

OMBCREATE BUSINESS_AREA 'SALES'

See Also

OMBALTER BUSINESS_AREA, OMBRETRIEVE BUSINESS_AREA

OMBCREATE BUSINESS_DEFINITION_MODULE

Purpose

To create a business definition module.

Prerequisites

Should be in the context of project.

Syntax

```
createEULModuleCommand = OMBCREATE ( BUSINESS_DEFINITION_MODULE
    "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
        "setReferenceClauseForDataOnlyModule" ] |
        "setReferenceClauseForDataOnlyModule" ) ] [
        "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataOnlyModule = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createEULModuleCommand

This command creates a business definition module.

QUOTED_STRING

Specify the name of a business definition module to be created.

setPropertiesClause

Associate a set of properties with a business definition module.

Basic properties for BUSINESS_DEFINITION_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a business definition module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a business definition module

Properties for BUSINESS_DEFINITION_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Discoverer Location for Business Definition Module

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Name: MLS_DEPLOYMENT_LANGUAGE

Type: STRING

Valid Values: N/A

Default: MLS_BASE_LANGUAGE

MLS Language to be used for deployment

Name: OBJECT_MATCHING

Type: STRING

Valid Values: BY_IDENTIFIER, BY_NAME

Default: BY_IDENTIFIER

Whether import should match up objects by identifier or by name

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceClauseForDataOnlyModule

Set location and/or icon set for the new business definition module.

addModuleReferenceLocationClause

Add runtime locations to the new business definition module.

propertyNameList

Comma-delimited list of property names. Property names are not in quotes.

propertyValueList

Comma-delimited list of property values.

setReferenceLocationClause

Set a location for a business definition module.

setReferenceIconSetClause

Set icon set for the new business definition module.

addReferenceLocationClause

Add a runtime location to the new business definition module.

propertyValue

Value for a specified property.

Examples

```
OMBCREATE BUSINESS_DEFINITION_MODULE 'src_module' SET PROPERTIES  
(DESCRIPTION, BUSINESS_NAME) VALUES ('this is a Business Definition  
module', 'int module')
```

This will create a business definition module named "src_module", its description is "this is a business definition module", and business name is "int module".

See Also

OMBCREATE, OMBALTER BUSINESS_DEFINITION_MODULE, OMBDROP
BUSINESS_DEFINITION_MODULE

OMBCREATE BUSINESS_Presentation_MODULE

Purpose

To create a presentation module.

Prerequisites

Should be in the context of project.

Syntax

```
createReportModuleCommand = OMBCREATE ( BUSINESS_Presentation_MODULE
    "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
        "setReferenceClauseForDataOnlyModule" ] |
        "setReferenceClauseForDataOnlyModule" ) ] [
        "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataOnlyModule = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createReportModuleCommand

This command creates a presentation module.

QUOTED_STRING

Specify the name of the presentation module to be created.

setPropertiesClause

Associate a set of properties with a presentation module.

Basic properties for BUSINESS_Presentation_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a presentation module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a presentation module

Properties for BUSINESS_PRESENTATION_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

BI Beans Location for Business Presentation Module

Name: DEFAULT_CATALOG_FOLDER

Type: STRING

Valid Values: N/A

Default: "

Default Catalog Folder for deployed BI Beans presentations

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to
create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceClauseForDataOnlyModule

Set location and/or icon set for the new business presentation module.

`addModuleReferenceLocationClause`

Add runtime locations to the new business presentation module.

`propertyNameList`

Comma-delimited list of property names. Property names are not in quotes.

`propertyValueList`

Comma-delimited list of property values.

`setReferenceLocationClause`

Set a location for a presentation module.

`setReferenceIconSetClause`

Set icon set for the new business presentation module.

`addReferenceLocationClause`

Add a runtime location to the new business presentation module.

`propertyValue`

Value for a specified property.

Examples

```
OMBCREATE BUSINESS_PRESENTATION_MODULE 'salesrep_module' SET
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('this is a presentation module',
'source module')
```

This will create a presentation module named "salesrep_module", its description is "this is a presentation module", and business name is "source module".

See Also

```
OMBCREATE, OMBALTER BUSINESS_PRESENTATION_MODULE, OMBDROP
BUSINESS_PRESENTATION_MODULE
```

OMBCREATE CALENDAR

Purpose

To create a calendar.

Prerequisites

Should be in the context of a Calendar Folder.

Syntax

```
createCalendarCommand = ( OMBCREATE CALENDAR "QUOTED_STRING" [ SET ( (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] ) |
    "setReferenceIconSetClause" ) ] ( ADD SCHEDULE ) [ SET
    "setPropertiesClause" ] )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

setPropertiesClause

Basic properties for CALENDAR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Calendar

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Calendar

Each calendar contains a single schedule with the name 'LOCALWINDOW' which has the following properties which define the various aspects of the schedule

Basic properties for SCHEDULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Schedule

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Schedule

Basic properties for the owned SCHEDULE OBJECT :

Name: START_TIME

Type: STRING(4000)

Valid Values: Start time in the format specified by property DATE_FORMAT.

The value NULL can also be passed if a schedule that takes on a start time of ASAP, that is upon deployment.

Default: "

Start time for the schedule

Name: END_TIME

Type: STRING(4000)

Valid Values: End time in the format specified by property DATE_FORMAT. The value NULL can also be passed if a schedule that repeats forever is to be created..

Default: "

End time for the schedule

Name: TIMEZONE

Type: STRING(4000)

Valid Values: N/A

Default: "

Time zone which times refer to.

Name: REPEAT_EXPRESSION

Type: STRING(4000)

Valid Values: N/A

Default: "

iCal format of a repeat expression. If no REPEAT_EXPRESSION is supplied, the scheduled activity is only performed once. The REPEAT_EXPRESSION includes the following:

FREQ

This specifies the type of recurrence. It must be specified. The possible predefined frequency values are YEARLY, MONTHLY, WEEKLY, DAILY, HOURLY, MINUTELY, and SECONDLY. Alternatively, specifies an existing schedule to use as a user-defined frequency.

INTERVAL

This specifies a positive integer representing how often the recurrence repeats. The default is 1, which means every second for secondly, every day for daily, and so on. The maximum value is 999.

BYMONTH

This specifies which month or months you want the job to execute in. You can use numbers such as 1 for January and 3 for March, as well as three-letter abbreviations such as FEB for February and JUL for July.

BYWEEKNO

This specifies the week of the year as a number. byweekno is only valid for YEARLY.

BYYEARDAY

This specifies the day of the year as a number. Valid values are 1 to 366. An example is 69, which is March 10 (31 for January, 28 for February, and 10 for March). 69 evaluates to March 10 for non-leap years and March 9 in leap years. -2 will always evaluate to December 30th independent of whether it is a leap year.

BYMONTHDAY

This specifies the day of the month as a number. Valid values are 1 to 31. An example is 10, which means the 10th day of the selected month. You can use the minus sign (-) to count backward from the last day, so, for example, BYMONTHDAY=-1 means the last day of the month and BYMONTHDAY=-2

means the next to last day of the month.

BYDAY

This specifies the day of the week from Monday to Sunday in the form MON, TUE, and so on. Using numbers, you can specify the 26th Friday of the year, if using a YEARLY frequency, or the 4th THU of the month, using a MONTHLY frequency. Using the minus sign, you can say the second to last Friday of the month. For example, -1 FRI is the last Friday of the month.

BYHOUR

This specifies the hour on which the job is to run. Valid values are 0 to 23. As an example, 10 means 10 a.m.

BYMINUTE

This specifies the minute on which the job is to run. Valid values are 0 to 59. As an example, 45 means 45 minutes past the chosen hour.

BYSECOND

This specifies the second on which the job is to run. Valid values are 0 to 59. As an example, 30 means 30 seconds past the chosen minute.

BYSETPOS (10gR2 only)

This selects one or more items by position in the list of timestamps that result after the whole calendaring expression is evaluated. It is useful for requirements such as running a job on the last workday of the month. Rather than attempting to express this with the other BY clauses, you can code the calendaring expression to evaluate to a list of every workday of the month, and then add the BYSETPOS clause to select only the last item of that list.

All of the preceding properties can be used in the SET PROPERTIES clause as well.

The following properties are supported for the GET PROPERTIES clause only

Name: DATE_FORMAT

Type: STRING

Valid Values: N/A

Default: N//A

Region specific time format string, eg MMM/dd/yy:HH:mm:ss.

Name: AVAILABLE_TIMEZONES

Type: STRING ARRAY

Valid Values: N/A

Default: N//A

List of available time zone ids that can be used to set the TIMEZONE property. This property is for information purposes only.

Name: PREVIEW_DATES

Type: STRING(4000)

Valid Values: N/A

Default: "

Preview of dates that this schedule includes. This property is only valid for OMBRETRIEVE.

Properties for CALENDAR:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBCREATE CALENDAR 'CAL1' ADD SCHEDULE
OMBCREATE CALENDAR 'CAL_10G' ADD SCHEDULE SET PROPERTIES (START_
TIME,
END_TIME, REPEAT_EXPRESSION, TIME_ZONE) VALUES ('22:Feb:2001:14:20:10',
'22:Feb:2002:14:20:10', 'FREQ=MINUTELY;INTERVAL=20', 'Europe/London')
OMBCREATE CALENDAR 'CAL_10G_2' ADD SCHEDULE SET PROPERTIES
(REPEAT_EXPRESSION) VALUES
('FREQ=YEARLY;INTERVAL=1;BYMONTH=APR;BYMONTHDAY=1;BYHOUR=12;B
YMINUTE=0;BYSECOND=0')
```

```
OMBCREATE CALENDAR 'ONE_TIME' ADD SCHEDULE SET PROPERTIES  
(START_TIME,  
TIME_ZONE) VALUES ('22:Feb:2001:14:20:10', 'Europe/London')
```

 For this release the user will only be able to create one schedule under a calendar, and the default name used will be 'LOCALWINDOW', the user will have to use this name to retrieve the properties of the schedule.

See Also

OMBCREATE, OMBALTER CALENDAR, OMBDROP CALENDAR, OMBRETRIEVE CALENDAR

OMBCREATE CALENDAR_MODULE

Purpose

To create an calendar module.

Prerequisites

Should be in the context of a Calendar Folder.

Syntax

```
createModuleCommand = OMBCREATE CALENDAR_MODULE "QUOTED_STRING" [
    "alterModulePropertiesOrIconSetClause" ]
alterModulePropertiesOrIconSetClause = ( SET ( ( "setPropertiesClause" [ (
    ( SET ( "setReferenceLocationClause" [ ( SET
    "setReferenceIconSetClause" ) | ( UNSET "unsetReferenceIconSetClause"
    ) ] ) ) | ( UNSET "unsetReferenceLocationClause" [ ( SET
    "setReferenceIconSetClause" ) | ( UNSET "unsetReferenceIconSetClause"
    ) ] ) ) | ( ( SET "setReferenceIconSetClause" ) | ( UNSET
    "unsetReferenceIconSetClause" ) ) ] ) | ( "setReferenceLocationClause"
    [ ( SET "setReferenceIconSetClause" ) | ( UNSET
    "unsetReferenceIconSetClause" ) ] ) | "setReferenceIconSetClause" ) )
| ( UNSET ( "unsetReferenceLocationClause" [ ( SET
    "setReferenceIconSetClause" ) | ( UNSET "unsetReferenceIconSetClause"
    ) ] ) | "unsetReferenceIconSetClause" )
) ] ) | "unsetReferenceIconSetClause" )
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
unsetReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
unsetReferenceLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

setPropertiesClause

Basic properties for CALENDAR_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Calendar Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Calendar Module

Properties for CALENDAR_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location to which the schedule will be deployed.

Name: EVAL_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location in which the scheduled object will be evaluated.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBCREATE CALENDAR_MODULE 'CAL_MODULE1'
```

See Also

OMBCREATE, OMBALTER CALENDAR_MODULE, OMBDROP CALENDAR_MODULE, OMBRETRIEVE CALENDAR_MODULE

OMBCREATE CHANGE_DATA_CAPTURE

Purpose

This command creates a change data capture with the given name and properties

Prerequisites

This command can only be executed in the context of a module

Syntax

```
createChangeSetCommand = OMBCREATE ( CHANGE_DATA_CAPTURE "QUOTED_STRING" (
    CAPTURE CHANGES | USE CHANGES ) { ( SET "setPropertyClause" ) |
    "addChangeSetSCOClauses" } )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
addChangeSetSCOClauses = ( ( ADD CAPTURE ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING" ) { "addCaptureSCOClauses" } [
    "specialCaptureColumnsClause"+ ] )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addCaptureSCOClauses = ( "addCaptureColumnsClause" | "addChangeSpecClause"
    | "addRowIdentifierClause" | "addTxnIdentifierClause" )
specialCaptureColumnsClause = [ DONT ] CAPTURE ( OLD_VALUES | USER_NAME )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
addCaptureColumnsClause = ( "addEachCaptureColumnClause" | (
    CAPTURE_COLUMNS ( NULL | ( "(" "columnNameList" ")" ) ) ) )
addChangeSpecClause = ( IDENTIFY ( INSERT | UPDATE | DELETE ) BY
    CHANGE_COLUMN "QUOTED_STRING" [ USING CHANGE_EXPRESSION
    "QUOTED_STRING" ] )
addRowIdentifierClause = ( "addEachRowIdentifierColumnClause" | (
    ROW_IDENTIFIER_COLUMNS ( NULL | ( "(" "columnNameList" ")" ) ) ) )
addTxnIdentifierClause = ( "addEachTxnIdentifierColumnClause" | (
    TXN_IDENTIFIER_COLUMNS ( NULL | ( "(" "columnNameList" ")" ) ) ) )
addEachCaptureColumnClause = ( ADD CAPTURE_COLUMN "QUOTED_STRING" AT
    POSITION "INTEGER_LITERAL" )
columnNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
addEachRowIdentifierColumnClause = ( ( ADD ROW_IDENTIFIER_COLUMN
    "QUOTED_STRING" ) )
addEachTxnIdentifierColumnClause = ( ( ADD TXN_IDENTIFIER_COLUMN
    "QUOTED_STRING" ) )
```

Keywords And Parameters

setPropertyClause

Basic properties for CHANGE_DATA_CAPTURE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Change Data Capture

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Change Data Capture

ombalter_changeset\$alterChangeSetCommand = This clause alters a change data capture.

Properties for CHANGE_DATA_CAPTURE:

Name: CAPTUREFROM

Type: STRING

Valid Values: FROM_START_DATE, ALL_AVAILABLE

Default: ALL_AVAILABLE

This property is used to specify whether the Change Data Capture object will capture all available changes or changes that occurred after a specified date.

Name: CAPTUREFROMDATE

Type: STRING

Valid Values: N/A

Default: "

This property is used to specify the date from which changes will be captured by the Change Data Capture object.

Name: DBA_LOCATION

Type: STRING

Valid Values: N/A

Default: "

This property specifies the DBA location from which the Supplemental Log scripts will need to be deployed. This property is used if the schema in which the Change Data Capture will be deployed is not the owner of the source table and is also not a DBA.

Name: DELETE_DANGLING_REF

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the dangling references to tables need to be deleted during a reconcile operation on the Change Data Capture.

Name: DEPLOY_GET_TIME_FUNCTION

Type: BOOLEAN

Valid Values: true, false

Default: true

This property is used to indicate whether a function that returns the system time on the Source system needs to be deployed. This is used if the Change Data Capture object is used in a mapping that will be deployed on a database instance different from the instance containing the source table.

Name: DEPLOY_SOURCE_SCRIPTS

Type: BOOLEAN

Valid Values: true, false

Default: true

This property specifies whether OWB should generate and deploy supplemental log and instantiation script for the source tables.

Name: FIRSTSCN

Type: STRING

Valid Values: N/A

Default: "

This value is needed if the Change Data Capture and the source table exist on different database instances.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GET_TIME_FUNCTION

Type: STRING(28)

Valid Values: N/A

Default: OWB\$SYSDATE

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.CHANGELOGS.GETTIMEFUNC:DESCRIPTION"

Name: LONG_TRANSACTION_WAIT_TIME

Type: NUMBER

Valid Values: >= 0

Default: 0

This property specifies the number of seconds to wait for long running active transactions to complete during change extraction.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

This property is used to specify the name of tablespace where all Change Data Capture structures are to be created.

Name: TRANSACTION_WAIT_TIME

Type: NUMBER

Valid Values: >= 0

Default: 0

This property specifies the number of seconds to wait for active transactions to complete during change extraction.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBCREATE CHANGE_DATA_CAPTURE 'EMPLOYEE_CHANGES' CAPTURE
CHANGES ADD
```

```
CAPTURE TABLE 'EMPLOYEE' CAPTURE_COLUMNS ('EMPID', 'EMPNAME')
```

This creates a change data capture called EMPLOYEE_CHANGES that captures the changes to columns EMPID, EMPNAME of table EMPLOYEE

See Also

OMBCREATE CHANGE_DATA_CAPTURE, OMBALTER CHANGE_DATA_CAPTURE, OMBDROP CHANGE_DATA_CAPTURE

OMBCREATE CMI_DEFINITION

Purpose

To create an CMI definition.

Prerequisites

Should be in the root context.

Syntax

```
createMIVDefinitionCommand = OMBCREATE ( CMI_DEFINITION "QUOTED_STRING"
    USING DEFINITION_FILE "QUOTED_STRING" ) [ SET "setPropertiesClause" ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`createMIVDefinitionCommand`

This command creates an CMI definition

`QUOTED_STRING`

Name of the CMI definition to be created.

`setPropertiesClause`

Associate a set of properties with an CMI definition.

`propertyNameList`

Comma separated list of property names. Property names are unquoted.

`propertyValueList`

Comma separated list of property values.

`propertyValue`

Value of a property.

Examples

```
OMBCREATE CMI_DEFINITION 'src_definition' USING DEFINITION_FILE
'/private/user1/miv_navision.xml'
```

See Also

OMBCREATE, OMBDROP CMI_DEFINITION

OMBCREATE CMI_MODULE

Purpose

To create an CMI module.

Prerequisites

Should be in the context of project.

Syntax

```
createMIVModuleCommand = OMBCREATE ( CMI_MODULE "QUOTED_STRING" USING
    CMI_DEFINITION "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
        "setReferenceClauseForDataMetadataModule" ] |
        "setReferenceClauseForDataMetadataModule" ) ] [
        "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
```

Keywords And Parameters

createMIVModuleCommand

This command creates an CMI module

setPropertiesClause

Associate a set of properties with an CMI module.

Basic properties for CMI_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of an CMI Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of an CMI Module

Properties for CMI_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

If this is a source module, this value indicates the location from which data will be read. If this is a target warehouse module, this value indicates the location where generated code will be deployed to and/or where data will be written to.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the CMI module.

addModuleReferenceLocationClause

Add runtime locations to the CMI module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a runtime location to the CMI module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the CMI module.

addReferenceLocationClause

Add a runtime location to the CMI module.

propertyValue

Value of a property.

setReferenceMetadataLocationClause

Set metadata location for the CMI module.

setReferenceIconSetClause

Set icon set for the CMI module.

Examples

```
OMBCREATE CMI_MODULE 'src_module' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('this is an CMI module', 'source module')
```

This will create an CMI module named "src_module", its description is "this is an CMI module", and bousiness name is "source module".

See Also

OMBCREATE, OMBALTER CMI_MODULE, OMBDROP CMI_MODULE

OMBCREATE COLLECTION

Purpose

This is an arbitrary grouping mechanism in OWB. Any first class object can be added to the collection.

Prerequisites

Should be in the context of a project, before creating a collection.

Syntax

```
createCollectionCommand = OMBCREATE ( COLLECTION "QUOTED_STRING" [ SET
    "setPropertyClause" [ SET "setReferenceIconSetClause" ] | SET
    "setReferenceIconSetClause" ] ( { "addReferenceClause" } ) )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addReferenceClause = ADD REFERENCE TO "componentRefClause"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
componentRefClause = ( EXTERNAL_TABLE | TABLE | VIEW | MATERIALIZED_VIEW |
    SEQUENCE | VARYING_ARRAY | OBJECT_TYPE | NESTED_TABLE | MAPPING |
    DIMENSION | CUBE | ADVANCED_QUEUE | STREAMS_QUEUE | QUEUE_TABLE |
    ORACLE_MODULE | TRANSFORMATION_MODULE | FLAT_FILE_MODULE | FLAT_FILE |
    PROCESS_FLOW | PROCESS_FLOW_PACKAGE | PROCESS_FLOW_MODULE |
    SAP_MODULE | CMI_MODULE | COLLECTION | FUNCTION | PROCEDURE | PACKAGE
    | BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE |
    PRESENTATION_TEMPLATE | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL
    | ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
    GATEWAY_MODULE | CONFIGURATION | REGISTERED_FUNCTION |
    PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER | DATA_AUDITOR |
    TRANSPORTABLE_MODULE | EXPERT_MODULE | EXPERT | CALENDAR_MODULE |
    CALENDAR | DATA_PROFILE | DATA_RULE_MODULE | DATA_RULE )
    "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createCollectionCommand

Create a collection of objects.

setPropertyClause

Set values for a number of properties when creating the collection.

Basic properties for COLLECTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the collection

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the collection

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addReferenceClause

Add a reference to the collection.

propertyNameList

Comma separated list of property names to retrieve values. Property names are unquoted.

propertyValueList

Comma separated list of property values.

componentRefClause

Specify the type of the object to reference.

propertyValue

Value of property.

Examples

```
OMBCREATE COLLECTION 'PURCHASING_WAREHOUSE' ADD REFERENCE TO
TABLE
```

```
'PURCHASING/ORDER' ADD REFERENCE TO TABLE
'PURCHASING/CUSTOMER'
```

```
OMBCREATE COLLECTION 'MY_FOLDERS' ADD REFERENCE TO ORACLE_
MODULE
```

```
'PURCHASING' ADD REFERENCE TO ORACLE_MODULE 'HUMAN_RESOURCES'
```

See Also

OMBCREATE, OMBALTER COLLECTION, OMBDROP COLLECTION

OMBCREATE CONFIGURATION

Purpose

To create a Configuration for supporting of objects multi-configuration.

Prerequisites

Should be in the context of project.

Syntax

```
createConfigurationCommand = OMBCREATE ( CONFIGURATION "QUOTED_STRING" [
    SET ( "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createConfigurationCommand

This command creates an Configuration.

setPropertiesClause

Associate a set of properties with a Configuration.

Basic properties for CONFIGURATION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Configuration

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Configuration.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Associate an Icon Set a Configuration.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE CONFIGURATION 'QA_CONFIG' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('This is a configuration for the QA Control Center',  
'QA Configuration')
```

This will create a Configuration named "QA_CONFIG", its description is "This is a configuration for QA Control Center", and business name is "QA Configuration".

See Also

OMBCREATE, OMBALTER CONFIGURATION, OMBDROP CONFIGURATION

OMBCREATE CONNECTOR

Purpose

To create a connector.

Prerequisites

Can be in any context; the name is a name of the connector's owning location and a new connector name separated by slash.

Syntax

```
createConnectorCommand = OMBCREATE ( CONNECTOR "QUOTED_STRING" SET [
    "setPropertyClause" SET ] "setReferenceToLocationClause" [ SET
    "setReferenceIconSetClause" ] )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceToLocationClause = ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createConnectorCommand

Create a new connector.

setPropertyClause

Set specified properties of the connector.

setReferenceToLocationClause

Set the name of the location which the connector references.

setReferenceIconSetClause

Set specified Icon Set.

propertyNameList

The names of the properties whose values you want to set.

Properties for CONNECTOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the connector.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the connector.

Name: DATABASE_LINK_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Database Link name.

Properties for CONNECTOR:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList

The values for the named properties.

propertyValue

A property value.

Examples

```
OMBCREATE CONNECTOR 'LOCATION_NAME/NEW_CONNECTOR_NAME'  
SET PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME) VALUES ('this is a connector', 'connector')
```

```
SET REF LOCATION 'MY_TARGET_LOCATION'
```

This will create a connector named "NEW_CONNECTOR_NAME" for the location "LOCATION_NAME" , and will assign MY_TARGET_LOCATION as a target location for the created connector; its description is "this is a connector", and business name is "connector".

See Also

OMBCREATE, OMBALTER CONNECTOR, OMBDROP CONNECTOR

OMBCREATE CONTROL_CENTER

Purpose

To create a control center.

Prerequisites

Can be in any context.

Syntax

```
createControlCenterCommand = OMBCREATE ( CONTROL_CENTER "QUOTED_STRING" (
    "createControlCenterSetPropertiesClause" [ SET
        "setReferenceIconSetClause" ] { "addReferenceLocationClause" } ) )
createControlCenterSetPropertiesClause = SET PROPERTIES "("
    "propertyNameList" ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addReferenceLocationClause = ADD ( REF | REFERENCE ) LOCATION
    "QUOTED_STRING" [ SET "setPropertiesClause" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createControlCenterCommand

Create a new control center.

createControlCenterSetPropertiesClause

Set the specified properties of the control center.

setReferenceIconSetClause

Set the specified icon set.

addReferenceLocationClause

Add a referenced location and set the specified properties of that referenced location.

propertyNameList

The names of the properties whose values you want to set.

Properties for CONTROL_CENTER:

Name: HOST

Type: STRING

Valid Values: N/A

Default: N/A

The host machine the control center is installed on.

Name: PORT

Type: NUMBER

Valid Values: 1 - 65535

Default: 1521

The port number of the database in which the control center is installed.

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The service name of the database in which the control center is installed.

Name: USER

Synonym: USER_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The name of the database user you wish to connect to the control center as.

Name: PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: N/A

The name of the schema in which the control center is installed.

All of the preceding properties (except of PASSWORD) are mandatory for OMBCREATE CONTROL_CENTER.

Basic properties for CONTROL_CENTER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the control center.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the control center.

Properties for a referenced location of the control center:

Name: IS_SOURCE

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, then a referenced location is a source location.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, then a referenced location is a target location.

Properties for CONTROL_CENTER:

Name: HOST

Type: STRING

Valid Values: N/A

Default: "

Host of the location

Name: NET_SERVICE_NAME

Type: STRING(2000)

Valid Values: N/A

Default: "

Net Service Name of the location

Name: PASSWORD

Type: STRING(30)

Valid Values: N/A

Default: "

Password for the location

Name: PORT

Type: NUMBER

Valid Values: N/A

Default: 1521

Port of the location

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: "

Schema name for the location

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: "

Service Name of the location

Name: USER

Type: STRING

Valid Values: N/A

Default: "

User name for the location

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyValueList

The values for the named properties.

propertyValue

A property value.

Examples

```
OMBCREATE CONTROL_CENTER 'MY_CONTROL_CENTER' SET PROPERTIES  
(DESCRIPTION,  
BUSINESS_NAME, HOST, PORT, SERVICE_NAME, USER, SCHEMA) VALUES ('this  
is my  
control center', 'My Control Center', 'localhost', 1521, 'orcl9i', 'scott',  
'tgu_repos')
```

This will create a control center named "my_control_center", its description is "this is my control center", business name is "My Control Center", host is localhost, port is 1521, service name is orcl9i, user is Scott, schema is tgu_repos.

See Also

OMBCREATE, OMBALTER CONTROL_CENTER, OMBDROP CONTROL_CENTER

OMBCREATE CORRECTION_MAPS_ACTION_PLAN

Purpose

Create an action plan for creating correction maps.

Prerequisites

Done after creating a correction schema, using a CORRECTION_SCHEMA_ACTION_PLAN.

Syntax

```
CreateActionPlanCommand = ( OMBCREATE TRANSIENT ( ( DEPLOYMENT_ACTION_PLAN
| ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" { "addActionClause" }
)
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
"setReferenceClause"
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
"propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
"useClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
"FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

CreateActionPlanCommand

Create a correction schema action plan.

Name: DATA_RULE_USAGE_NAME

Type: String

Valid Values: Any data rule usage name associated with the profile table

Name: ERROR_HANDLING_STRATEGY

Type: String

One of IGNORE REPORT CLEANSE

This property determines the action to take when the rule fails. If the action is cleanse, then a CORRECTION_STRATEGY is used to determine the specific cleansing action.

Name: CORRECTION_STRATEGY

Type: String

One of DISCARD DOMAINSOUNDEX DOMAINSIMILARITY DISCARD MIN MAX
FD_MAPLET

FIX_REMOTE UK_MATCHMERGE

This property determines the correction action to take when the rule fails.

Not all correction strategies are supported with all data rule types.

Data Rule Type Correction Strategy

DOMAIN_LIST_RULE DISCARD

DOMAINSOUNDEX

DOMAINSIMILARITY

DOMAIN_RANGE_RULE DISCARD

MIN

MAX

FUNCTIONAL_DEPENDENCY_RULE DISCARD

FD_MAPLET

REFERENCE_RULE DISCARD

FIX_REMOTE

IDENTITY_RULE DISCARD

UK_MATCHMERGE

All other rule types DISCARD

Name: CORRECTION_ORDER

Type: NUMBER

Valid Values: 1-any number

Optional: This sets the relative order of the data corrections.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Action name.

setClause

Set the properties of an action and/or associate an object with an action.

propertiesClause

Set the properties and/or associate/disassociate an object with an action.

setReferenceClause

Associate an object with an action.

ObjType

Object type. The only valid value is DATA_PROFILE_TABLE.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Currently used only in CORRECTION_SCHEMA_ACTION_PLAN.

propertyValue

Value of a property.

Examples

```
OMBCREATE TRANSIENT CORRECTION_MAPS_ACTION_PLAN 'CMAPS_PLAN'
ADD ACTION 'ACTION1'
SET PROPERTIES (DATA_RULE_USAGE_NAME, ERROR_HANDLING_STRATEGY)
VALUES ('CHECK_MIN_SALARY', 'REPORT')
SET REF PROFILE_REFERENCE 'EMP'
ADD ACTION 'ACTION2'
SET PROPERTIES (DATA_RULE_USAGE_NAME, ERROR_HANDLING_STRATEGY,
CORRECTION_STRATEGY, CORRECTION_ORDER)
VALUES ('CHECK_JOB', 'CLEANSE', 'DOMAINSOUNDEX', 2)
SET REF PROFILE_REFERENCE 'EMP'
ADD ACTION 'ACTION3'
SET PROPERTIES (DATA_RULE_USAGE_NAME, ERROR_HANDLING_STRATEGY,
CORRECTION_STRATEGY, CORRECTION_ORDER)
VALUES ('CHECK_MIN_DEPTNO', 'CLEANSE', 'MIN', 1)
SET REF PROFILE_REFERENCE 'EMP'
OMBPROFILE CORRECTION_MAPS_ACTION_PLAN 'CMAPS_PLAN'
IN ORACLE_MODULE '/PROJ/CORR_MOD'
```

See Also

OMBPROFILE

OMBCREATE CORRECTION_SCHEMA_ACTION_PLAN

Purpose

Create an action plan for creating a correction schema, identifying the tables that will be mapped to that schema.

Prerequisites

First do OMBCREATE DATA_RULE, and OMBALTER TABLE 'T' ADD DATA_RULE_USAGE, on the tables that will be mapped.

Syntax

```
CreateActionPlanCommand = ( OMBCREATE TRANSIENT ( ( DEPLOYMENT_ACTION_PLAN
| ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" { "addActionClause" }
)
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
"setReferenceClause"
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
"propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
"useClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
"FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

CreateActionPlanCommand

Create a correction schema action plan.

QUOTED_STRING

Action plan name.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Action name.

setClause

Set the properties of an action and/or associate an object with an action.

propertiesClause

Set the properties and/or associate/disassociate an object with an action.

setReferenceClause

Associate an object with an action.

ObjType

Object type. The only valid value is DATA_PROFILE_TABLE.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Use data rule in construction of correction schema.

QUOTED_STRING

Name of data rule usage.

propertyValue

Value of a property.

Examples

```
OMBCREATE TRANSIENT CORRECTION_SCHEMA_ACTION_PLAN 'CSHEMA_
PLAN'
ADD ACTION 'ACTION1'
SET REF PROFILE_REFERENCE 'LOC'
ADD ACTION 'ACTION2'
SET REF PROFILE_REFERENCE 'EMP'
USE DATA_RULE_USAGE 'CHECK_JOB'
USE DATA_RULE_USAGE 'MIN_SALARY'
ADD ACTION 'ACTION3'
SET REF PROFILE_REFERENCE 'DEPT'
USE DATA_RULE_USAGE 'CHECK_MAX_DEPTNO'
OMBPROFILE CORRECTION_SCHEMA_ACTION_PLAN 'CSHEMA_PLAN'
```

IN ORACLE_MODULE '/PROJ/CORR_MOD'

See Also

OMBPROFILE

OMBCREATE CUBE

Purpose

This command creates a Cube.

Prerequisites

Should be in Oracle Module context.

Syntax

```
createCubeCommand = OMBCREATE CUBE "cubeName" [ SET ( (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] ) |
    "setReferenceIconSetClause" ) ] [ "setCubeAggFunctionClause" ] { ADD (
    "addMeasureClause" | "addDimensionUseClause" |
    "addCompositeDimensionClause" ) } [ "cubeImplementationClause" ]
cubeName = "QUOTED_STRING"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
setCubeAggFunctionClause = SET AGGREGATE_FUNCTION TO "aggFunctionName"
addMeasureClause = MEASURE "measureName" [ SET "setPropertiesClause" ] [
    "setAggFunctionClause" ] [ "setPreComputedLevelsClause" ] [
    "setCalcExprClause" ]
addDimensionUseClause = DIMENSION_USE "dimensionUseName" [ SET
    "setPropertiesClause" ] [ "setDimensionUseReferenceClause" ] [ AT
    POSITION "INTEGER_LITERAL" ]
addCompositeDimensionClause = COMPOSITE_DIMENSION "compositeDimensionName"
    [ SET "setPropertiesClause" ] [
    "setCompositeDimensionReferenceClause" ]
cubeImplementationClause = IMPLEMENTED BY ( ( "cubeBindingClause" {
    "measureBindingClause" | "dimensionUseBindingClause" } ) |
    "cubeAutoBindingClause" )
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
aggFunctionName = "QUOTED_STRING"
measureName = "QUOTED_STRING"
setAggFunctionClause = SET AGGREGATE_FUNCTIONS "aggFunctionList" FOR
    DIMENSIONS "dimensionList"
setPreComputedLevelsClause = SET PRECOMPUTE_LEVELS "preComputedLevelList"
setCalcExprClause = SET CALCULATED_EXPRESSION "(" "calcExpr" ")"
dimensionUseName = "QUOTED_STRING"
setDimensionUseReferenceClause = [ USE ROLE "roleName" ] SET ( REF |
    REFERENCE ) [ LEVEL "levelName" OF ] DIMENSION "dimensionName"
compositeDimensionName = "QUOTED_STRING"
setCompositeDimensionReferenceClause = SET ( REF | REFERENCE ) DIMENSIONS
    "(" "dimensionName" { "," "dimensionName" } ")"
cubeBindingClause = ( TABLE "tableName" | VIEW "viewName" )
measureBindingClause = "measureLocator" BOUND TO COLUMN "columnName"
dimensionUseBindingClause = "dimensionUseLocator" BOUND TO COLUMN
    "columnName"
cubeAutoBindingClause = SYSTEM
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
aggFunctionList = "(" "aggFunction" { "," "aggFunction" } ")"
dimensionList = "(" "dimension" { "," "dimension" } ")"
preComputedLevelList = "(" "preComputedLevel" { "," "preComputedLevel" } )
```

```

    ) "
    calcExpr = "QUOTED_STRING"
    roleName = "QUOTED_STRING"
    levelName = "QUOTED_STRING"
    dimensionName = "QUOTED_STRING"
    tableName = "QUOTED_STRING"
    viewName = "QUOTED_STRING"
    measureLocator = MEASURE "measureName"
    columnName = "QUOTED_STRING"
    dimensionUseLocator = DIMENSION_USE "dimensionUseName"
    aggFunction = "QUOTED_STRING"
    dimension = "QUOTED_STRING"
    preComputedLevel = "QUOTED_STRING"

```

Keywords And Parameters

setPropertiesClause

Basic properties for CUBE, MEASURE and DIMENSION_USE:

Basic properties for CUBE :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Cube

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Cube

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: 'true', 'false'

Default: 'true'

The dimension is visible to OLAP end user

Name: UNIQUE_KEY_CONSTRAINT

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

set the Unique Key constraint on the Business Key

Name: STORAGE

Type: STRING

Valid Values: 'RELATIONAL', 'MOLAP'

Default: 'RELATIONAL'

The storage of a cube can be relational or molap

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: ''

Set the analytical workspace name where the cube is implemented

Name: AW_CUBE_NAME

Type: STRING(32)

Valid Values: N/A

Default: ''

Set the Analytical Workspace cube physical object name

Name: USE_GLOBAL_INDEX

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether to generate a composite for measure partition combination

Name: BITMAP_INDEX

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether to generate a bitmap for a cube

Name: SUMMARY_REFRESH_METHOD

Type: STRING

Valid Values: 'On Demand', 'On Commit'

Default: 'On Commit'

Sets the Solve flag for Relational Cube whether to solve the cube 'On

Demand' or 'On Commit'

Name: SPARSITY_ULTRA_COMPRESS

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether its a compressed cube. Only valid for Molap cube.

Basic properties for Measure :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Cube

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Cube

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: 'true', 'false'

Default: 'true'

The dimension is visible to OLAP end user

Name: CALCULATED_MEASURE

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the measure to be aggregatable

Name: AUTO_SOLVE

Type: STRING

Valid Values: 'true', 'false'

Default: 'false'

Sets the flag to say whether to measure is Auto Solve

Name: AW_MEASURE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace measure physical object name

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH

LOCAL TIME ZONE, VARCHAR, VARCHAR2

Default: NUMBER

Sets the datatype of a Cube Measure

Name: SCALE

Type: NUMBER

Valid Values: -85 - 125

Default: 1

The scale of a number.

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 39

Default: 1

The precision of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Basic properties for Cube Dimension Use :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Cube Dimension Use

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Cube Dimension Use

Properties for CUBE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS

Type: STRING

Valid Values: DEPLOY_AGGREGATION, DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY,

DEPLOY_TO_CATALOG_ONLY

Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for cube, they are DDL Scripts for Relational Cube or Scripts for OLAP API-II or Scripts for AW.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MATERIALIZED_VIEW_INDEX_TABLESPACE

Type: STRING

Valid Values: N/A

Default: USERS

Tablespace for materialized view indexes.

Name: MATERIALIZED_VIEW_TABLESPACE

Type: STRING

Valid Values: N/A

Default: USERS

Tablespace for materialized view.

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Cube is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBCREATE CUBE 'ALL_SALES'
SET PROPERTIES (BUSINESS_NAME, DESCRIPTION)
VALUES ('Sales Cube', 'Sales Cube')
```

```
ADD MEASURE 'AMOUNT'  
SET PROPERTIES (DATATYPE,PRECISION,SCALE) VALUES ('NUMBER',10,2)  
ADD MEASURE 'QUANTITY'  
SET PROPERTIES (DATATYPE) VALUES ('NUMBER')  
ADD MEASURE 'COST'  
SET PROPERTIES (DATATYPE,PRECISION,SCALE) VALUES ('NUMBER',10,2)  
ADD DIMENSION_USE 'CHANNELS'  
SET REF LEVEL 'CHANNEL' OF DIMENSION 'CHANNELS'
```

See Also

OMBCREATE, OMBALTER CUBE, OMBRETRIEVE CUBE, OMBDROP CUBE

OMBCREATE DATA_AUDITOR

Purpose

Create a data auditor in an Oracle Module.

Prerequisites

1. The current context of scripting must be an Oracle Module.
2. No concurrent user should be locking the Oracle Module or any of its ancestors exclusively at the moment the map is being created.
3. The data auditor name must not conflict with existing data auditor names, map names and the maps names that concurrent user tries to use.

Syntax

```
createDataAuditorCommand = OMBCREATE DATA_AUDITOR "dataAuditorName" [ SET
    ( ( "setPropertiesClause" [ SET "setReferenceIconSetClause" ] ) |
      "setReferenceIconSetClause" ) ] { ADD "addOperatorClause" }
dataAuditorName = "QUOTED_STRING"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addOperatorClause = "operatorType" OPERATOR "operatorName" [ SET
    "setPropertiesClause" ] [ BOUND TO "setBindingClause" ]
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
operatorType = "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
setBindingClause = "bindableLocator"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
bindableLocator = "bindableType" "bindableName"
bindableType = "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"
```

Keywords And Parameters

`createDataAuditorCommand`

Create a data auditor in an Oracle Module.

`dataAuditorName`

Name of data auditor.

`setPropertiesClause`

Set the properties of the data auditor.

`setReferenceIconSetClause`

Set icon set for the data auditor.

addOperatorClause

Adds an operator to a data auditor.OMBCREATE DATA_AUDITOR 'MAP1'
 SET PROPERTIES (business_name, description)
 VALUES ('My map', 'Audit table foo')
 ADD TABLE OPERATOR 'CUST_SRC'
 BOUND TO TABLE '../SRC_MODULE/CUST_SRC'

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

operatorType

Type of a mapping operator. The following operator types are available:
 CUBE, DIMENSION, MATERIALIZED_VIEW, TABLE, VIEW.

operatorName

Name of an operator.

setBindingClause

Set the binding during the creation of a operator.

propertyKey

A property key for an object.

Basic properties for DATA_AUDITOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the data auditor

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the daa auditor

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

#

Properties for DATA_AUDITOR:

Name: ANALYZE_TABLE_SAMPLE_PERCENTAGE

Type: NUMBER

Valid Values: N/A

Default: 90

The default percentage of rows to be sampled when the target tables are analyzed for statistics to improve performance during insertion.

Name: ANALYZE_TABLE_STATEMENTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate statistics collection statement if this is true.

Name: ANSI_SQL_SYNTAX

Type: BOOLEAN

Valid Values: true, false

Default: true

A switch between ANSI and Oracle SQL Syntax.

Name: BULK_PROCESSING_CODE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate bulk processing code if this is true.

Name: BULK_SIZE

Type: NUMBER

Valid Values: N/A

Default: 50

The default number of rows to be fetched in batch during cursor processing.

Name: COMMIT_CONTROL

Type: STRING

Valid Values: AUTO_COMMIT, AUTO_CORR_COMMIT, MANUAL_COMMIT

Default: AUTO_COMMIT

Options for how commit is performed

Name: COMMIT_FREQUENCY

Type: NUMBER

Valid Values: N/A

Default: 1000

The default number of rows processed before a commit is issued.

Name: DEFAULT_AUDIT_LEVEL

Type: STRING

Valid Values: COMPLETE, ERROR_DETAILS, NONE, STATISTICS

Default: ERROR_DETAILS

The default audit level when the step is executed.

Name: DEFAULT_OPERATING_MODE

Type: STRING

Valid Values: ROW_BASED, ROW_BASED_TARGET_ONLY, SET_BASED,
SET_BASED_FAIL_OVER_TO_ROW_BASED,
SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY
Default: SET_BASED_FAIL_OVER_TO_ROW_BASED
The default operating mode.

Name: DEFAULT_PURGE_GROUP
Type: STRING
Valid Values: N/A
Default: WB
The default purge group to be used when the step is executed.

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true
True if the map is deployable to a physical implementation

Name: ENABLE_PARALLEL_DML
Type: BOOLEAN
Valid Values: true, false
Default: true
Determine if PDML is enabled at runtime.

Name: ERROR_TRIGGER
Type: STRING
Valid Values: N/A
Default: "
Error trigger procedure name.

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Name: GENERATION_MODE

Type: STRING

Valid Values: ALL_MODES, ROW_BASED, ROW_BASED_TARGET_ONLY, SET_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY

Default: ALL_MODES

The operating modes for which code should be generated.

Name: MAXIMUM_NUMBER_OF_ERRORS

Type: NUMBER

Valid Values: N/A

Default: 50

The default maximum number of errors encountered before terminating the step execution.

Name: OPTIMIZED_CODE

Type: BOOLEAN

Valid Values: true, false

Default: true

Attempt to generate optimized code if this is true.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TARGET_LOAD_ORDERING

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate target load ordering code.

Name: THRESHOLD_MODE

Type: STRING

Valid Values: PERCENTAGE, SIX_SIGMA

Default: PERCENTAGE

Use six sigma or percentage for failure thresholds.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

bindableLocator

Location of the object to be bound to an operator.

bindableType

Type of object bound to an operator.

bindableName

Name of the object bound to operator.

Examples

```
OMBCREATE DATA_AUDITOR 'MAP1'
```

```
OMBCREATE DATA_AUDITOR 'MAP1'
```

```
SET PROPERTIES (business_name, description)
```

```
VALUES ('My map', 'Audit table foo')
```

```
ADD TABLE OPERATOR 'CUST_SRC'
```

```
BOUND TO TABLE '../SRC_MODULE/CUST_SRC'
```

See Also

OMBCREATE DATA_AUDITOR, OMBALTER DATA_AUDITOR, OMBRETRIEVE DATA_AUDITOR, OMBDROP DATA_AUDITOR

OMBCREATE DATA_PROFILE

Purpose

createDataProfileCommandPurposeTag??

Prerequisites

Should be in the context of project.

Syntax

```
createDataProfileCommand = OMBCREATE ( DATA_PROFILE "QUOTED_STRING" [ SET
  ( "setPropertiesClause" [ SET "setReferenceClause" ] |
    "setReferenceClause" ) ] { ADD ( TABLE | VIEW | EXTERNAL_TABLE |
    MATERIALIZED_VIEW | DIMENSION | CUBE | FLAT_FILE ) "QUOTED_STRING" } )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
  "propertyValueList" ")"
setReferenceClause = ( "setReferenceLocationClause" [ SET
  "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createDataProfileCommand

This command creates a Data Profile

QUOTED_STRING

Name of the Data Profile to be created.

setPropertiesClause

Associate a set of properties with a Data Profile.

Configuration properties for DATA_PROFILE that affect loading:

Name: COPY_DATA

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable copying of data from source to profile workspace.

Name: CALCULATE_DATATYPES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data type discovery for the selected table.

Name: CALCULATE_COMMON_FORMATS

Type: BOOLEAN

Valid Values: true | false

Default: false

This tells the profiler if common formats are to be discovered for all sources in this profile.

Name: NULL_VALUE

Type: STRING

Valid Values: any string value

Default: null

This value will be considered as the null value when profiling. Please enclose the value in single quotes. An unquoted null (the current default value) will be considered a database null.

Name: SAMPLE_RATE

Type: NUMBER

Valid Values: 1-100

Default: 100

This value will be the percent of total rows that will be randomly selected during loading.

Configuration properties for DATA_PROFILE that affect profiling:

Name: CALCULATE_DOMAINS

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable domain discovery.

Name: DOMAIN_MAX_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

The maximum number of distinct values in a column in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MAX_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

The maximum number of distinct values in a column, expressed as a percentage of the total number of rows in the table, in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MIN_COUNT

Type: NUMBER

Valid Values: 1-any number

Default: true

The minimum number of rows for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: DOMAIN_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: true

The minimum number of rows, expressed as a percentage of the total number of rows, for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: CALCULATE_UK

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable unique key discovery.

Name: UK_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a unique key relationship.

Name: CALCULATE_FD

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable functional dependency discovery.

Name: FD_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a functional dependency relationship.

Name: FD_UK_LHS_COUNT

Type: NUMBER

Valid Values: 1-number of attributes of source less 1

Default: 1

This is the maximum number of attributes for unique key and functional dependency profiling.

Name: CALCULATE_FK

Type: BOOLEAN

Valid Values: true | false

Default: true

Setting this to true will enable foreign key discovery.

Name: FK_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that need to satisfy a foreign key relationship.

Name: CALCULATE_REDUNDANT_COLUMNS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable redundant column discovery with respect to a foreign key-unique key pair.

Name: REDUNDANT_MIN_PERCENT

Type: NUMBER

Valid Values: 1-100

Default: 75

This is the minimum percentage of rows that are redundant.

Name: CALCULATE_DATA_RULES

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable data rule profiling for the selected table.

Name: CALCULATE_PATTERNS

Type: BOOLEAN

Valid Values: true | false

Default: false

Setting this to true will enable pattern discovery.

Name: MAX_NUM_PATTERNS

Type: NUMBER

Valid Values: any number less than the number of rows of the source

Default: 10

This tells the profiler to get the top-N patterns for the attribute.

setReferenceClause

The location where the profile will be deployed.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a location to a Data Profile.

setReferenceIconSetClause

Set icon set for the profile.

propertyValue

Value of a property.

Examples

```
OMBCREATE DATA_PROFILE 'HR_DP' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME)
```

```
VALUES ('This is a Data Profile', 'data profile') \  
ADD TABLE 'HR/EMPLOYEES' \  
ADD TABLE 'HR/DEPARTMENTS' \  
ADD TABLE 'HR/REGIONS' \  
ADD TABLE 'HR/LOCATIONS' \  
ADD TABLE 'HR/JOBS' \  
ADD TABLE 'HR/JOB_HISTORY' \  
ADD TABLE 'HR/COUNTRIES'
```

This will create a Data Profile named "data_profile", with a description of "this is a Data Profile", business name of "data_profile" and all the tables in the Oracle supplied schema HR.

See Also

OMBCREATE, OMBALTER DATA_PROFILE, OMBDROP DATA_PROFILE

OMBCREATE DATA_RULE

Purpose

Create a DataRule rule in an dataRule rule Module.

Prerequisites

1. The current context of scripting must be an DataRule Rule Module.
2. No concurrent user should be locking the DataRule Rule Module or any of its ancestors exclusively at the moment the rule is being created.
3. The rule name must not conflict with existing rule names.

Syntax

```
createDataRuleCommand = OMBCREATE ( DATA_RULE "QUOTED_STRING" [ SET (
    "setInitialPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] [ ADD "addDomainValueClause" ] )
setInitialPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES
    "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addDomainValueClause = ( DOMAIN_VALUE "QUOTED_STRING" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createDataRuleCommand

Create a data rule.

QUOTED_STRING

Data rule name.

setInitialPropertiesClause

Set properties associated with the created data rule. The property **RULE_TYPE** must be set.

setReferenceIconSetClause

Set icon set for the data rule.

addDomainValueClause

Add a domain value.

QUOTED_STRING

Domain value.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE DATA_RULE 'NOT_NULL'
SET PROPERTIES(RULE_TYPE)
VALUES('ATTR_VALUE_RULE')
```

The data rule will be created with the appropriate groups and attributes for the specific rule type.

Rule types Group Name Attribute Name

ATTR_VALUE_RULE THIS VALUE

DOMAIN_NO_NULL_RULE THIS NO_NULL_VALUE

DOMAIN_LIST_RULE THIS VALUE

DOMAIN_PATTERN_LIST_RULE THIS VALUE

DOMAIN_RANGE_RULE THIS VALUE

DOMAIN_FORMAT_TELEPHONE_RULE THIS PHONE_NUMBER

DOMAIN_FORMAT_IP_RULE THIS IP_ADDRESS

DOMAIN_FORMAT_SSN_RULE THIS SOCIAL_SECURITY_NUMBER

DOMAIN_FORMAT_DATE_RULE THIS DATE_VALUE

DOMAIN_FORMAT_NUMBER_RULE THIS NUMBER_VALUE

DOMAIN_FORMAT_URL_RULE THIS URL

DOMAIN_FORMAT_EMAIL_RULE THIS EMAIL

FUNCTIONAL_DEP_RULE DEPENDENCY DEPENDENCY

: DETERMINANTS DETERMINANT_1

IDENTITY_RULE THIS KEY_ATTRIBUTE_1

REFERENCE_RULE LOCAL LOCAL_KEY_ATTRIBUTE_1

: REMOTE REMOTE_KEY_ATTRIBUTE_1

NAMEADDRESS_RULE THIS NA_LINE1
: NA_LINE2
: NA_LINE3
: NA_LINE4
: NA_LINE5
: NA_LINE6
: NA_LINE7
: NA_LINE8
: NA_LINE9
: NA_LINE10
: NA_FIRSTNAME
: NA_MIDDLENAME
: NA_MIDDLENAME2
: NA_MIDDLENAME3
: NA_LASTNAME
: NA_FIRSTPARTNAME
: NA_LASTPARTNAME
: NA_PRENAME
: NA_POSTNAME
: NA_PERSON
: NA_PERSON2
: NA_PERSON3
: NA_FIRMNAME
: NA_PRIMARYADDRESS
: NA_SECONDARYADDRESS
: NA_ADDRESS
: NA_ADDRESS2
: NA_NEIGHBORHOOD
: NA_LASTLINE
: NA_LASTLINE_2
: NA_CITY
: NA_LOCALITYNAME
: NA_LOCALITY_2
: NA_LOCALITY_3
: NA_LOCALITY_4
: NA_STATE
: NA_POSTALCODE

: NA_COUNTRYNAME

: NA_COUNTRYCODE

/

See Also

OMBCREATE, OMBALTER DATA_RULE, OMBDROP DATA_RULE

OMBCREATE DATA_RULE_MODULE

Purpose

To create a data rule module.

Prerequisites

Should be in the context of project.

Syntax

```
createDataRuleModuleCommand = OMBCREATE ( DATA_RULE_MODULE "QUOTED_STRING"
    ( [ SET ( "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
      "setReferenceIconSetClause" ) ] ) )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createDataRuleModuleCommand

Create data rule module.

QUOTED_STRING

Data rule module name.

setPropertiesClause

Set the properties of the data rule module.

Basic properties for DATA_RULE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Data Rule Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Data Rule Module

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Set icon set for the data rule module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE DATA_RULE_MODULE 'br_mod' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('this is a data rule module', 'br module')
```

This will create a data rule module named "br_module", its description is "this is a data rule module", and data name is "br module".

See Also

OMBCREATE, OMBALTER DATA_RULE_MODULE, OMBDROP DATA_RULE_MODULE

OMBCREATE DEPLOYMENT

Purpose

To create a Deployment.

Prerequisites

Should be in the context of CONFIGURATION.

Syntax

```
createDeploymentCommand = OMBCREATE ( DEPLOYMENT "QUOTED_STRING" [ SET
    "setPropertiesClause" ] [ "setReferenceControlCenterClause" ] [
    "setAsActiveClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceControlCenterClause = SET ( REFERENCE | REF ) CONTROL_CENTER
    "QUOTED_STRING"
setAsActiveClause = SET AS ACTIVE
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createDeploymentCommand

This command creates a Deployment.

setPropertiesClause

Associate a set of properties with a Deployment.

Basic properties for DEPLOYMENT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Deployment

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Deployment.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceControlCenterClause

Sets the reference of a Control Center to the created Deployment.

setAsActiveClause

Sets this Deployment as Active.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE DEPLOYMENT 'QA_DEPLOYMENT' SET REF CONTROL_CENTER  
'QA_CC' SET  
PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('This is a Deployment of  
the QA_CONFIG Configuration of QA_CC Control Center', 'QA Deployment') SET  
AS ACTIVE
```

This will create a Deployment named "QA_DEPLOYMENT", set the reference of a control center "QA_CC", set this Deployment as active; the Deployment description is "This is a Deployment of the QA_CONFIG Configuration of QA_CC Control Center", and business name is "QA Deployment".

See Also

OMBCREATE, OMBALTER DEPLOYMENT, OMBDROP DEPLOYMENT

OMBCREATE DEPLOYMENT_ACTION_PLAN

Purpose

Create a deployment action plan.

Prerequisites

There must be a current working project.

Syntax

```

CreateActionPlanCommand = ( OMBCREATE TRANSIENT ( ( DEPLOYMENT_ACTION_PLAN
| ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" { "addActionClause" }
)
addActionClause = ADD ACTION "QUOTED_STRING" [ SET "setClause" ]
setClause = ( "propertiesClause" [ SET "setReferenceClause" ] ) |
"setReferenceClause"
propertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
"propertyValueList" ")"
setReferenceClause = ( REF | REFERENCE ) "ObjType" "QUOTED_STRING" {
"useClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
useClause = USE DATA_RULE_USAGE "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
"FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

CreateActionPlanCommand

Create a deployment action plan.

QUOTED_STRING

Name of the action plan in a single-quoted string. It is case-insensitive.

Must be unique within a project.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Name of the action in a single-quoted string. It is case-insensitive.

Must be unique within an action plan.

setClause

Set the properties of an action and/or associate an object with an action.

propertiesClause

Associate a set of properties with an action.

PROPERTIES

The only valid property is OPERATION, which specifies the type of action to be taken.

setReferenceClause

Associate an object with an action.

ObjType

Object type. Valid values are ADVANCED_QUEUE, CUBE, DIMENSION, EXTERNAL_TABLE, CONNECTOR, FUNCTION, MAPPING, MATERIALIZED_VIEW, PROCEDURE, PROCESS_FLOW_PACKAGE, SCHEDULABLE, SEQUENCE, TABLE, and VIEW.

QUOTED_STRING

Absolute or relative path name of an object (for example '/MY_PROJECT/MODULE_X/TABLE_Y').

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

useClause

Currently used only in CORRECTION_SCHEMA_ACTION_PLAN.

propertyValue

Value of a property. Valid values for OPERATION are DROP, CREATE, REPLACE and UPGRADE.

Examples

```
OMBCREATE TRANSIENT DEPLOYMENT_ACTION_PLAN 'MY_PLAN'
```

```
OMBCREATE TRANSIENT DEPLOYMENT_ACTION_PLAN 'MY_PLAN' ADD  
ACTION 'DUMMY'
```

```
OMBCREATE TRANSIENT DEPLOYMENT_ACTION_PLAN 'MY_PLAN'  
ADD ACTION 'MY_TABLE_DEPLOY' SET PROPERTIES (OPERATION) VALUES  
( 'DROP' )  
SET REFERENCE TABLE ' /MY_PROJECT/MY_MODULE/MY_TABLE'  
ADD ACTION 'MY_VIEW_CREATE' SET PROPERTIES (OPERATION) VALUES  
( 'CREATE' )  
SET REFERENCE VIEW 'MY_MODULE/MY_VIEW'
```

```
OMBCREATE TRANSIENT DEPLOYMENT_ACTION_PLAN 'MY_PLAN'  
ADD ACTION 'MY_SCHEDULE_MAP_DEPLOY' SET PROPERTIES (OPERATION)  
VALUES  
( 'CREATE' )  
SET REFERENCE SCHEDULABLE 'MY_MODULE/MY_MAP'
```

OMBCREATE DIMENSION

Purpose

This command creates a dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
createDimensionCommand = OMBCREATE DIMENSION "dimensionName" [
    "setPropertiesClause" ] [ "setReferenceIconSetClause" ] [
    "setDimensionKeySequenceClause" ] [ ADD
    "addSurrogateKeyDimensionAttributeClause" ] [ ADD
    "addParentKeyDimensionAttributeClause" ] { ADD (
    "addDimensionAttributeClause" | "addDimensionRoleClause" |
    "addLevelClause" | "addLevelAttributeClause" | "addHierarchyClause" |
    "addSkipLevelsClause" ) } [ "createDimensionBindingClause" ]
dimensionName = "QUOTED_STRING"
setPropertiesClause = SET PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = SET ( REFERENCE | REF ) ICONSET
    "QUOTED_STRING"
setDimensionKeySequenceClause = SET ( REF | REFERENCE ) SEQUENCE
    "sequenceName"
addSurrogateKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS SURROGATE_KEY [ "setPropertiesClause"
    ]
addParentKeyDimensionAttributeClause = DIMENSION_ATTRIBUTE
    "dimensionAttributeName" SET AS PARENT_KEY [ "setPropertiesClause" ]
addDimensionAttributeClause = DIMENSION_ATTRIBUTE "dimensionAttributeName"
    [ SET AS BUSINESS_KEY ] [ "setPropertiesClause" ]
addDimensionRoleClause = DIMENSION_ROLE "roleName" [ "setPropertiesClause"
    ]
addLevelClause = LEVEL "levelName" [ "setPropertiesClause" ]
addLevelAttributeClause = LEVEL_ATTRIBUTE "levelAttributeName" OF
    "levelLocator" [ "setPropertiesClause" ]
    "setLevelAttributeReferenceClause"
addHierarchyClause = HIERARCHY "hierarchyName" [ "setPropertiesClause" ] [
    "hierarchyLevelReferenceClause" ]
addSkipLevelsClause = SKIP_LEVELS FROM "levelLocator" TO "levelLocator" IN
    "hierarchyLocator"
createDimensionBindingClause = IMPLEMENTED BY ( SYSTEM ( STAR | SNOWFLAKE
    ) | STAR ( ( DIMENSION_KEY BOUND TO COLUMN "columnName" (
    "levelBindingClause" { "levelAttributeBindingClause" } )+ {
    "setBindRelationshipClause" } [ "setSkipBindRelationshipClause" ] ) )
    | SNOWFLAKE ( ( ( "levelBindingClause" { "levelAttributeBindingClause"
    } )+ { "setBindRelationshipClause" } [
    "setSkipBindRelationshipClause" ] ) ) )
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
sequenceName = "QUOTED_STRING"
dimensionAttributeName = "QUOTED_STRING"
roleName = "QUOTED_STRING"
levelName = "QUOTED_STRING"
levelAttributeName = "QUOTED_STRING"
levelLocator = LEVEL "levelName"
```

```

setLevelAttributeReferenceClause = SET ( REF | REFERENCE ) (
    "dimensionAttributeLocator" [ TYPE_THREE_SCD_PREVIOUS
    "levelAttributeLocator" ] )
hierarchyName = "QUOTED_STRING"
hierarchyLevelReferenceClause = SET ( REF | REFERENCE ) LEVELS "(
    "levelName" { ", " "levelName" } )"
hierarchyLocator = HIERARCHY "hierarchyName"
columnName = "QUOTED_STRING"
levelBindingClause = "levelLocator" BOUND TO ( TABLE "tableName" | VIEW
    "viewName" )
levelAttributeBindingClause = LEVEL_ATTRIBUTE "levelAttributeName" OF
    LEVEL "levelName" BOUND TO COLUMN "columnName"
setBindRelationshipClause = LEVEL_RELATIONSHIP OF "levelLocator" IN
    "hierarchyLocator" BOUND TO COLUMN "columnName"
setSkipBindRelationshipClause = SKIP_LEVEL_RELATIONSHIP OF "levelLocator"
    IN "hierarchyLocator" BOUND TO COLUMN "columnName"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
dimensionAttributeLocator = DIMENSION_ATTRIBUTE "dimensionAttributeName"
levelAttributeLocator = LEVEL_ATTRIBUTE "levelAttributeName"
tableName = "QUOTED_STRING"
viewName = "QUOTED_STRING"

```

Keywords And Parameters

setPropertiesClause

Basic properties for DIMENSION, DIMENSION_ATTRIBUTE, LEVEL, LEVEL_ATTRIBUTE

and HIERARCHY:

Basic properties for DIMENSION :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Dimension

Name: SCD_TYPE

Type: INTEGER

Valid Values: 1, 2, 3

Default: 1

Slowly changing policy to be applied on the dimension. Give Integer values
1, 2, 3
for Slowly changing type one, two and three

Name: TYPE

Type: STRING

Valid Values: 'NONE', 'TIME'

Default: 'NONE'

'NONE' it does not recognize it as any specific type of dimension.

'TIME' dimension is a time dimension

Name: OLAP_TYPE

Type: STRING

Valid Values: NONE, TIME

Default: NONE

Dimension type for OLAP, get regular dimension 'NONE' and for OLAP Time

Dimension 'TIME'

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The dimension is visible to OLAP end user

Name: UNIQUE_KEY_CONSTRAINT

Type: STRING

Valid Values: true, false

Default: false

set the Unique Key constraint on the Business Key

Name: STORAGE

Type: STRING

Valid Values: RELATIONAL, AW

Default: 'RELATIONAL'

The storage of a dimension can be AW or relational

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the analytical workspace name where the dimension is implemented

Name: AW_DIMENSION_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace dimension physical object name

Name: USE_BUSINESS_KEYS

Type: STRING

Valid Values: true, false

Default: false

Set the flag for Analytical Workspace dimension to use Business Keys as data source

Basic properties for DIMENSION_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension_Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Dimension_Attribute

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,
INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,
NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH
LOCAL TIME ZONE, VARCHAR, VARCHAR2
Default: NUMBER
Sets the datatype of a Dimension_Attribute

Name: SCALE
Type: NUMBER
Valid Values: -85 - 125
Default: 1
The scale of a number.

Name: LENGTH
Type: NUMBER
Valid Values:
Default: 1
The length of a number

Name: PRECISION
Type: NUMBER
Valid Values: 0 - 39
Default: 1
The precision of a number.

Name: DESCRIPTOR
Type: STRING
Valid Values: NONE, SHORT_DESCRIPTION, LONG_DESCRIPTION, END_DATE, TIME_SPAN, PRIOR_PERIOD, YEAR_AGO_PERIOD
Default: NONE
The following properties are set on dimension attribute so that it is recognized by OLAP service
'NONE' it is not specially recognized type by OLAP service
'SHORT_DESCRIPTION' sets as a short description

'LONG_DESCRIPTION' sets as a long description

'END_DATE' sets as a last date of a period.

'TIME_SPAN' sets as a number of days in a period.

'PRIOR_PERIOD' sets as the prior period number.

'YEAR_AGO_PERIOD' sets as the time period a year before this period

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The dimension attribute is visible to OLAP end user

Name: TYPE

Type: STRING

Valid Values: NONE, START_DATE, END_DATE, TIME_SPAN

Default: NONE

'NONE' dimension attribute so OWB does not recognize it as any specific type.

'START_DATE' dimension attribute of time dimension as the start date of a period

'END_DATE' dimension attribute of time dimension as the end date of a period

'TIME_SPAN' dimension attribute of time dimension as the time span

Name: AW_ATTRIBUTE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

set the AW object name implementing the dimension attribute

Basic properties for Level :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Level

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Level

Name: TYPE

Type: STRING

Valid Values: NONE, DAY, FISCAL_WEEK, FISCAL_MONTH, FISCAL_QUARTER, FISCAL_YEAR, CALENDAR_WEEK, CALENDAR_MONTH, CALENDAR_QUARTER, CALENDAR_YEAR

Default: NONE

For regular relational dimension level (non-time dimension level) use 'NONE'. For relational time dimension use other values.

Name: OLAP_TYPE

Type: STRING

Valid Values: NONE, DAY, MONTH, QUARTER, YEAR, TOTAL

Default: NONE

Level has an olap-type for OLAP-based levels; use 'NONE' for regular levels, and other values for OLAP Time Dimension

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The level is visible to OLAP end user

Basic properties for Level_Attribute :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Level_Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Level_Attribute

Name: DEFAULT_VALUE

Type: STRING(200)

Valid Values: N/A

Default: "

This is used to construct the default parent policy for loading data into dimension.

Name: OLAP_TYPE

Type: STRING

Valid Values: NONE, SHORT_DESCRIPTION, END_DATE, TIME_SPAN, PRIOR_PERIOD,

YEAR_AGO_PERIOD

Default: NONE

The following properties are set on level attribute so that it is recognized by OLAP service

'NONE' it is not specially recognized type by OLAP service

'SHORT_DESCRIPTION' sets as a short description

'END_DATE' sets as a last date of a period.

'TIME_SPAN' sets as a number of days in a period.

'PRIOR_PERIOD' sets as the prior period number.

'YEAR_AGO_PERIOD' sets as the time period a year before this period

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The level attribute is visible to OLAP end user

Name: TYPE

Type: STRING

Valid Values: NONE, START_DATE, END_DATE, TIME_SPAN

Default: 'NONE'

'NONE' level attribute so OWB does not recognize it as any specific type.

'START_DATE' level attribute of time dimension as the start date of a period

'END_DATE' level attribute of time dimension as the end date of a period

'TIME_SPAN' level attribute of time dimension as the time span

Name: TYPE_TWO_SCD_EFFECTIVE_DATE

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Effective Date for Slowly changing type 2

Name: TYPE_TWO_SCD_EXPIRATION_DATE

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Expiration Date for Slowly changing type 2

Name: TYPE_TWO_SCD_TRIGGER

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Trigger for saving history for Slowly changing type 2

Name: TYPE_THREE_SCD_EFFECTIVE_DATE

Type: STRING

Valid Values: true, false

Default: false

The level attribute is defined as Effective Date for Slowly changing type 3

Basic properties for Hierarchy :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Hierarchy

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Hierarchy

Name: OLAP_USER_VISIBLE

Type: STRING

Valid Values: true, false

Default: true

The hierarchy is visible to OLAP end user

Name: TYPE

Type: STRING

Valid Values: NONE, FISCAL, CALENDAR_YEAR, CALENDAR_WEEK

Default: NONE

'NONE' hierarchy so OWB does not recognize it as any specific type.

'FISCAL' fiscal hierarchy for time dimension

'CALENDAR_YEAR' calendar year hierarchy time dimension

'CALENDAR_WEEK' calendar week hierarchy time dimension

Name: DEFAULT_DISPLAY

Type: STRING

Valid Values: true, false

Default: false

The hierarchy is set as Default display hierarchy

Name: VALUE_BASED

Type: STRING

Valid Values: true, false

Default: false

Sets the flag to define a Value Based Hierarchy for AW only

Properties for DIMENSION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS

Type: STRING

Valid Values: DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY, DEPLOY_TO_CATALOG_ONLY

Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for Dimension, they are DDL Scripts for Relational Dimensional or Scripts for ROLAP or or Scripts for AW.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: VIEW_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Name of the view that is generated to hide the control rows on the dimension implementation table of a star schema. If this field is left blank, the view name will default to '<Name of Dimension>_v'

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Dimension is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```

OMBCREATE SEQUENCE 'PRODUCT_STAR_SEQ'
OMBCREATE DIMENSION 'PRODUCTS'
SET PROPERTIES (BUSINESS_NAME, DESCRIPTION)
VALUES ('products','this is a products dimension')
SET REF SEQUENCE 'PRODUCT_STAR_SEQ'
ADD DIMENSION_ATTRIBUTE 'ID' SET AS SURROGATE_KEY
ADD DIMENSION_ATTRIBUTE 'NAME'
SET PROPERTIES (DATATYPE,LENGTH,DESCRIPTOR)
VALUES ('VARCHAR2',60,'SHORT_DESCRIPTION')
ADD DIMENSION_ATTRIBUTE 'DESCRIPTION'
SET PROPERTIES (DATATYPE,LENGTH,DESCRIPTOR)
VALUES ('VARCHAR2',100,'LONG_DESCRIPTION')
ADD DIMENSION_ATTRIBUTE 'SOURCE_ID'
SET PROPERTIES (DATATYPE) VALUES ('NUMBER')
ADD DIMENSION_ATTRIBUTE 'PACK_SIZE'
SET PROPERTIES (DATATYPE,LENGTH) VALUES ('VARCHAR2',30)
ADD DIMENSION_ATTRIBUTE 'LIST_PRICE'
SET PROPERTIES (DATATYPE,LENGTH) VALUES ('VARCHAR2',10)
ADD DIMENSION_ATTRIBUTE 'DATE_UPDATED'

```

```

SET PROPERTIES (DATATYPE,LENGTH) VALUES ('VARCHAR2',10)
ADD DIMENSION_ATTRIBUTE 'SUPPLIER_ID'
SET PROPERTIES (DATATYPE) VALUES ('NUMBER')
ADD LEVEL 'PRODUCT'
ADD LEVEL 'SUBCATEGORY'
ADD LEVEL 'CATEGORY'
ADD LEVEL 'TOTAL'
ADD LEVEL_ATTRIBUTE 'ID' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'ID'
ADD LEVEL_ATTRIBUTE 'NAME' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'NAME'
ADD LEVEL_ATTRIBUTE 'DESCRIPTION' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'DESCRIPTION'
ADD LEVEL_ATTRIBUTE 'SOURCE_ID' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'SOURCE_ID'
ADD LEVEL_ATTRIBUTE 'PACK_SIZE' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'PACK_SIZE'
ADD LEVEL_ATTRIBUTE 'LIST_PRICE' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'LIST_PRICE'
ADD LEVEL_ATTRIBUTE 'SUPPLIER_ID' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'SUPPLIER_ID'
ADD LEVEL_ATTRIBUTE 'DATE_UPDATED' OF LEVEL 'PRODUCT'
SET REF DIMENSION_ATTRIBUTE 'DATE_UPDATED'
ADD LEVEL_ATTRIBUTE 'ID' OF LEVEL 'SUBCATEGORY'
SET REF DIMENSION_ATTRIBUTE 'ID'
ADD LEVEL_ATTRIBUTE 'NAME' OF LEVEL 'SUBCATEGORY'
SET REF DIMENSION_ATTRIBUTE 'NAME'
ADD LEVEL_ATTRIBUTE 'DESCRIPTION' OF LEVEL 'SUBCATEGORY'
SET REF DIMENSION_ATTRIBUTE 'DESCRIPTION'
ADD LEVEL_ATTRIBUTE 'SOURCE_ID' OF LEVEL 'SUBCATEGORY'
SET REF DIMENSION_ATTRIBUTE 'SOURCE_ID'
ADD LEVEL_ATTRIBUTE 'ID' OF LEVEL 'CATEGORY'
SET REF DIMENSION_ATTRIBUTE 'ID'
ADD LEVEL_ATTRIBUTE 'NAME' OF LEVEL 'CATEGORY'
SET REF DIMENSION_ATTRIBUTE 'NAME'
ADD LEVEL_ATTRIBUTE 'DESCRIPTION' OF LEVEL 'CATEGORY'
SET REF DIMENSION_ATTRIBUTE 'DESCRIPTION'

```

```
ADD LEVEL_ATTRIBUTE 'SOURCE_ID' OF LEVEL 'CATEGORY'  
SET REF DIMENSION_ATTRIBUTE 'SOURCE_ID'  
ADD LEVEL_ATTRIBUTE 'ID' OF LEVEL 'TOTAL'  
SET REF DIMENSION_ATTRIBUTE 'ID'  
ADD LEVEL_ATTRIBUTE 'NAME' OF LEVEL 'TOTAL'  
SET REF DIMENSION_ATTRIBUTE 'NAME'  
ADD LEVEL_ATTRIBUTE 'DESCRIPTION' OF LEVEL 'TOTAL'  
SET REF DIMENSION_ATTRIBUTE 'DESCRIPTION'  
ADD HIERARCHY 'PROD_STD'  
SET REF LEVELS ('TOTAL','CATEGORY','SUBCATEGORY','PRODUCT')  
IMPLEMENTED BY SYSTEM STAR
```

See Also

OMBCREATE, OMBALTER DIMENSION, OMBRETRIEVE DIMENSION, OMBDROP DIMENSION

OMBCREATE DRILL_PATH

Purpose

Creates a Drill path to enable drilling to detail.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createDrillPathCommand = OMBCREATE DRILL_PATH "QUOTED_STRING" [ SET
    "setPropertiesClause" ] [ SET "setReferenceIconSetClause" ] {
    "DrillLevelClause" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
DrillLevelClause = ADD DRILL_LEVEL "QUOTED_STRING" [ SET
    "setPropertiesClause" ] { "setDrillItemClause" }
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
setDrillItemClause = SET ( REF | REFERENCE ) ITEM "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING" "itemJoinUsages"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
itemJoinUsages = { SET ( REF | REFERENCE ) USING JOIN "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING" }
```

Keywords And Parameters

createDrillPathCommand

This command creates a drill path.

QUOTED_STRING

Specify the name of the drill path to be created.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for drill paths. Valid properties are as shown:

Basic properties for DRILL_PATH:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill path

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the drill path

Basic properties for DRILL_LEVEL:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the drill level

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the drill level

Properties for DRILL_PATH:

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true
Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

DrillLevelClause

This clause adds a drill level to a drill path.

DRILL_LEVEL

A level in a drill path.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

setDrillItemClause

This clause adds a reference to an item for the level.

propertyValue

This is a property value.

itemJoinUsages

The specific joins to be used.

Examples

```
OMBCREATE DRILL_PATH 'SALES'
```

See Also

```
OMBALTER DRILL_PATH, OMBRETRIEVE DRILL_PATH
```

OMBCREATE DRILL_TO_DETAIL

Purpose

Creates a Drill to detail to allow drilling between items.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createDrillToDetailCommand = OMBCREATE DRILL_TO_DETAIL "QUOTED_STRING" [
    SET "setPropertiesClause" ] [ SET "setReferenceIconSetClause" ] {
    "addDrillToDetailReferenceClause" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addDrillToDetailReferenceClause = SET ( REF | REFERENCE ) ITEM
    "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createDrillToDetailCommand

This command creates a drill to detail.

QUOTED_STRING

Specify the name of the drill to detail to be created.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for a drill to detail. Valid properties are as shown:

Basic properties for DRILL_TO_DETAIL:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill to detail

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill to detail

Properties for DRILL_TO_DETAIL:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

addDrillToDetailReferenceClause

This adds a reference to an item to a drill to detail.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

propertyValue

This is a property value.

Examples

```
OMBCREATE DRILL_TO_DETAIL 'SALES_ITEM'
```


See Also

OMBALTER DRILL_TO_DETAIL, OMBRETRIEVE DRILL_TO_DETAIL

OMBCREATE EXPERT

Purpose

To create an expert.

Prerequisites

In the context of an expert module.

Syntax

```
createExpertCommand = OMBCREATE EXPERT "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] { "addExpertDetailClauses" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addExpertDetailClauses = ADD ( "addParameterClause" | "addVariableClause"
    | "addNestedExpertClause" | "addTaskClause" | "addTransitionClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addParameterClause = PARAMETER "QUOTED_STRING" [ OF TASK "QUOTED_STRING" ]
    [ SET "setPropertiesClause" ] [ "parameterBindingClause" ]
addVariableClause = VARIABLE "QUOTED_STRING" [ SET "setPropertiesClause" ]
addNestedExpertClause = NESTED_EXPERT TASK "QUOTED_STRING" SET [
    "collectPropertiesClause" SET ] ( REF | REFERENCE ) EXPERT
    "QUOTED_STRING" [ SET "setReferenceIconSetClause" ]
addTaskClause = "TASK_TYPE" TASK "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ]
addTransitionClause = TRANSITION "QUOTED_STRING" FROM TASK "QUOTED_STRING"
    TO "QUOTED_STRING" [ SET "setPropertiesClause" ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
parameterBindingClause = UNBIND | BIND TO ( "parameterLocator" |
    "variableLocator" )
collectPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
parameterLocator = PARAMETER "QUOTED_STRING" [ OF TASK "QUOTED_STRING" ]
variableLocator = VARIABLE "QUOTED_STRING"
```

Keywords And Parameters

createExpertCommand

Creates an expert. The following expert parameters will be created by default:

1) EXP_LAUNCH_CONTEXT, type: STRING, direction: IN

The console context in which this expert is being launched.

2) EXP_LAUNCH_CONTEXT_TYPE, type: STRING, direction: IN

The type of the console context in which this expert is being launched.

3) EXP_OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) EXP_TYPE_TO_CREATE, type: STRING, direction: IN

The object type selected to be created when this expert is launched.

setPropertyClause

This clause sets properties for the corresponding object.

Basic properties for EXPERT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the expert

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the expert

Basic properties for TASK:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the task

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the task. This is equivalent to the Goal of task in the expert editor.

Name: INSTRUCTION

Type: STRING(4000)

Valid Values: N/A

Default: "

The instruction for the task

Name: PREPROCESSING

Type: STRING

Valid Values: N/A

Default: N/A

The pre-processing script for the task

Name: MAIN

Type: STRING

Valid Values: N/A

Default: N/A

The main script for the task

Name: POSTPROCESSING

Type: STRING

Valid Values: N/A

Default: N/A

The post-processing script for the task

Basic properties for START TASK:

Name: PROC_DECL

Type: STRING(4000)

Valid Values: N/A

Default: "

The procedure declaration for the expert.

Basic properties for TRANSITION:

Name: BUSINESS_NAME

Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the transition

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the transition

Name: TRANSITION_CONDITION
Type: STRING(4000)
Valid Values: N/A
Default: "
Condition of the transition

Name: TRANSITION_ORDER
Type: NUMBER
Valid Values: N/A
Default: N/A
Order of the transition

Basic properties for PARAMETER:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the parameter

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the parameter

Name: DATATYPE

Type: STRING

Valid Values: STRING, NUMBER, BOOLEAN, ARRAY

Default: STRING

Datatype of the parameter

Name: DIRECTION

Type: STRING

Valid Values: IN, OUT, INOUT

Default: IN

Direction of the parameter

Name: VALUE

Type: Same as datatype of the parameter

Valid Values: N/A

Default: N/A

The static value of the parameter

Basic properties for VARIABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the variable

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the variable

Name: DATATYPE

Type: STRING

Valid Values: STRING, NUMBER, BOOLEAN, ARRAY

Default: STRING

Datatype of the variable

Name: VALUE

Type: Same as datatype of the variable

Valid Values: N/A

Default: N/A

The static value of the variable

Properties for EXPERT:

Name: CLOSE_ASSISTANT_ON_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the task assistant should be automatically closed after the expert has been run.

Name: CLOSE_WINDOWS_ON_EXECUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Close all open windows when this expert is run.

Name: FINISH_DIALOG_ON_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: true

Shows the finish dialog upon completion of expert.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LOGGING

Type: BOOLEAN

Valid Values: true, false

Default: false

Log output to file when this expert is being run. A log file will be created in <shiphome>/owb/log directory whenever this expert is run.

Name: MENU_ITEM_DISPLAY_STRING

Type: STRING

Valid Values: N/A

Default: "

The display string when this expert is added as a menu item.

Name: ONLY_RUN_FROM_MENU

Type: BOOLEAN

Valid Values: true, false

Default: false

Only allow this expert to be run when it is attached to a menu item.

Name: REVERT_TO_SAVED_ON_ERROR

Type: BOOLEAN

Valid Values: true, false

Default: false

Revert to saved metadata if error occurs when the expert is run.

Name: RUN_STANDALONE

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the expert should be run as a standalone in expert assistant mode or not.

Name: SAVE_ALL_BEFORE_START

Type: BOOLEAN

Valid Values: true, false

Default: false

Save all metadata before running the expert.

Name: SHOW_BUSY_DIALOG

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether busy dialog should be shown when OMB or Java tasks are executed in non-standalone mode.

Name: SHOW_LOG_WINDOW

Type: BOOLEAN

Valid Values: true, false

Default: false

Sets whether the log window should be shown when running the expert.

Name: SHOW_PROGRESS_GRAPH

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the progress graph dialog should be shown when running the expert.

Name: SHOW_TASK_ASSISTANT

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the task assistant should be shown when running the expert.

Note:

1. N/A means any valid character in supported character set.

2. " " represents an empty string

setReferenceIconSetClause

Set icon set for the expert module.

addExpertDetailClauses

Add child objects for the expert.

propertyNameList

The list of property names.

propertyValueList

The list of property values being set.

addParameterClause

Add a parameter to the expert or one of it's tasks.

addVariableClause

Add a variable to the expert.

addNestedExpertClause

Add a nested expert to the expert. The nested expert is added by reference.

addTaskClause

Add a task to the expert. Valid task types include: ADVANCED_QUEUE, ALTERNATIVE_SORT_ORDER, ANALYZE_IMPACT, ANALYZE_LINEAGE, BUSINESS_AREA, CHANGE_MANAGER, COMMIT, CONTROLCENTERJOBS, CUBE, CUSTOM_DIALOG, DATA_AUDITOR, DATA_PROFILE, DATA_RULE, DATA_VIEWER, DEPLOY, DERIVATION, DIMENSION, DRILL_PATH, DRILL_TO_DETAIL, END, EXTERNAL_TABLE, FLAT_FILE, FUNCTION, GENERATION, ITEM_FOLDER, JAVA, LIST_OF_VALUES, MAPPING, MATERIALIZED_VIEW, NESTED_EXPERT, OBJECT_SELECTOR, OMB, PLUGGABLE_MAPPING, PRESENTATION_TEMPLATE, PROCEDURE, PROCESS_FLOW, REGISTERED_FUNCTION, SELECT_SOURCE, SELECT_TARGET, SEQUENCE, SOURCE_IMPORT, START, STARTJOB, TABLE, VALIDATION, VIEW.

Definition of each task is as follow:

Task type : ADVANCED_QUEUE

Group : MML

Description: A task to create or alter an advanced queue.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : ALTERNATIVE_SORT_ORDER

Group : MML

Description: A task to create or alter an alternative sort order.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : ANALYZE_IMPACT

Group : Service

Description: A task to analyze impact of an object.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object for analyzing impact.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object whose impact will be analyzed.

Task type : ANALYZE_LINEAGE

Group : Service

Description: A task to analyze lineage of an object.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object for analyzing lineage.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object whose lineage will be analyzed.

Task type : BUSINESS_AREA

Group : MML

Description: A task to create or alter a business area.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : CHANGE_MANAGER

Group : Service

Description: A task to invoke the change manager.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : COMMIT

Group : Service

Description: A task to perform commit in the design repository.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : CONTROLCENTERJOBS

Group : Service

Description: A task to launch the Control Center Job Monitor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) CONTROL_CENTER_NAME, type: STRING, direction: IN

The Control Center to use.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : CUBE

Group : MML

Description: A task to create or alter a cube.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : CUSTOM_DIALOG

Group : UI

Description: A task to show a custom dialog for user interaction.

Built-in parameters:

1) GUI_RETURN_VALUE, type: ARRAY, direction: OUT

The return value of type ARRAY that stores the name and return value pair for each UI component.

2) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

3) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : DATA_AUDITOR

Group : MML

Description: A task to create or alter a data auditor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DATA_PROFILE

Group : MML

Description: A task to create or alter a data profile.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1

for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DATA_RULE

Group : MML

Description: A task to create or alter a data rule.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DATA_VIEWER

Group : Service

Description: A task to launch the data viewer.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of the OWB FCO, such as TABLE.

4) OBJECT_NAME, type: STRING, direction: IN

The name of the OWB FCO.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : DEPLOY

Group : Service

Description: A task to launch the Control Center deployment monitor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) ACTION_PLAN_NAME, type: STRING, direction: IN

The deployment action plan to use.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : DERIVATION

Group : Service

Description: A task to run derivation service.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to derive.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object to be derived.

Task type : DIMENSION

Group : MML

Description: A task to create or alter a dimension.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DRILL_PATH

Group : MML

Description: A task to create or alter a drill path.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : DRILL_TO_DETAIL

Group : MML

Description: A task to create or alter a drill to detail.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : END

Group : FlowControl

Description: A task that marks the end of the flow.

Built-in parameters:

Task type : EXTERNAL_TABLE

Group : MML

Description: A task to create or alter an external table.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : FLAT_FILE

Group : MML

Description: A task to create or alter a file.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : FUNCTION

Group : MML

Description: A task to create or alter a function.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : GENERATION

Group : Service

Description: A task to invoke the generation dialog.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to generate.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object to be generated.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : ITEM_FOLDER

Group : MML

Description: A task to create or alter an item folder.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is

specified as EDIT.

Task type : JAVA

Group : Integration

Description: A task to execute a Java program.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) CLASS_URL, type: STRING, direction: IN

The URL specification of the Jar file. An example for a local jar file xyz.jar can be specified as file:/xyz.jar

4) CLASS_NAME, type: STRING, direction: IN

The class to load. This includes the package as well in dotted notation. For example, oracle.owb.Test

5) METHOD_NAME, type: STRING, direction: IN

The static method to execute.

6) ARGUMENT_LIST, type: STRING, direction: IN

The argument list to be passed into the static method.

Task type : LIST_OF_VALUES

Group : MML

Description: A task to create or alter a list of values.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : MAPPING

Group : MML

Description: A task to create or alter a mapping.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : MATERIALIZED_VIEW

Group : MML

Description: A task to create or alter a materialized view.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : NESTED_EXPERT

Group : FlowControl

Description: A task that references another expert.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : OBJECT_SELECTOR

Group : UI

Description: A task that shows a dialog for user to select an object.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) OBJECT_TYPE, type: STRING, direction: IN

The type of object for selection.

Task type : OMB

Group : Integration

Description: A task to launch a generic OMB script.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

Task type : PLUGGABLE_MAPPING

Group : MML

Description: A task to create or alter a pluggable mapping.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : PRESENTATION_TEMPLATE

Group : MML

Description: A task to create or alter a presentation template.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is

specified as EDIT.

Task type : PROCEDURE

Group : MML

Description: A task to create or alter a procedure.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : PROCESS_FLOW

Group : MML

Description: A task to create or alter a process flow.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : REGISTERED_FUNCTION

Group : MML

Description: A task to create or alter a registered function.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : SELECT_SOURCE

Group : UI

Description: A task that allows users to pick a metadata source.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) MODULE_NAME, type: STRING, direction: OUT

The name of the module.

5) LOCATION_NAME, type: STRING, direction: OUT

The name of the location.

6) OBJECT_TYPE, type: STRING, direction: OUT

The type of object selected as the source.

7) SOURCE_TYPE, type: STRING, direction: INOUT

The type of the source, for example file or database.

Task type : SELECT_TARGET

Group : UI

Description: A task that allows users to pick a target.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) MODULE_NAME, type: STRING, direction: OUT

The name of the module.

5) LOCATION_NAME, type: STRING, direction: OUT

The name of the location.

6) OBJECT_TYPE, type: STRING, direction: OUT

The type of object selected as the target.

7) SOURCE_TYPE, type: STRING, direction: INOUT

The type of the source, for example file or database.

Task type : SEQUENCE

Group : MML

Description: A task to create or alter a sequence.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : SOURCE_IMPORT

Group : Service

Description: A task to invoke the import wizard.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

4) OBJECT_TYPE, type: STRING, direction: IN

The type of object to import.

5) IMPORT_MODE, type: STRING, direction: IN

The mode for import. Select FULL_MODE for importing multiple objects, or MINIMAL_MODE for single import.

Task type : START

Group : FlowControl

Description: A task that marks the beginning of the flow.

Built-in parameters:

Task type : STARTJOB

Group : Service

Description: A task to launch the start job monitor.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to be started.

4) OBJECT_PATH, type: STRING, direction: IN

The context path of the object to be started.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : TABLE

Group : MML

Description: A task to create or alter a table.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

Task type : VALIDATION

Group : Service

Description: A task to invoke the validation dialog.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OBJECT_TYPE, type: STRING, direction: IN

The type of object to validate.

4) OBJECT_PATH, type: STRING, direction: IN

The path to the object to be validated.

5) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

Task type : VIEW

Group : MML

Description: A task to create or alter a view.

Built-in parameters:

1) EXIT_CODE, type: NUMBER, direction: OUT

The exit code represents the exit status on the execution of a task. An exit code of 0 means the task executed successfully, 1 for error, and -1 for cancel.

2) RETURN_VALUE, type: STRING, direction: OUT

The return value from the execution of a task.

3) OPERATION, type: STRING, direction: IN

The operation to be performed. It is either CREATE or EDIT.

4) PARENT_CONTEXT, type: STRING, direction: IN

The parent context for executing this task.

5) OBJECT_TO_EDIT, type: STRING, direction: IN

The OWB object to be edited for this OWB object task, when the OPERATION is specified as EDIT.

addTransitionClause

Add a transition to the expert.

propertyValue

Value of a property.

parameterBindingClause

Bind or unbind two parameters. Note that this is setting the binding attribute of the parameter object and is not necessarily the same as the direction of data flow. The parameter to be set should be the one whose

binding attribute is modified by the Object Inspector in the expert editor.

collectPropertiesClause

This clause sets properties for the corresponding object.

parameterLocator

Specify a parameter, either one of the expert or of a task.

variableLocator

Specify a variable in the expert.

Examples

This command will create a simple expert with just a table task between the START and END task:

```
OMBCREATE EXPERT 'EXP1' \  
SET PROPERTIES (DESCRIPTION) VALUES ('My first expert') \  
ADD TABLE TASK 'MY_TABLE_TASK' \  
ADD TRANSITION 'X1' FROM TASK 'START_TASK' TO 'MY_TABLE_TASK' \  
ADD TRANSITION 'X2' FROM TASK 'MY_TABLE_TASK' TO 'END_TASK'
```

See Also

OMBCREATE, OMBALTER EXPERT, OMBRETRIEVE EXPERT, OMBDROP EXPERT

OMBCREATE EXPERT_MODULE

Purpose

To create an expert module.

Prerequisites

In the context of a project.

Syntax

```
createExpertModuleCommand = OMBCREATE EXPERT_MODULE "QUOTED_STRING" [ SET
    ( "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
      "setReferenceIconSetClause" ) ]
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createExpertModuleCommand

Create an expert module.

setPropertiesClause

Set properties for the expert module.

Basic properties for EXPERT_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the expert module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the expert module

setReferenceIconSetClause

Set icon set for the expert module.

propertyNameList

The list of properties to set.

propertyValueList

The list of property values to set.

propertyValue

The value of the property.

Examples

This command will create an expert module with name 'EM1':

```
OMBCREATE EXPERT_MODULE 'EM1'
```

See Also

OMBCREATE, OMBALTER EXPERT_MODULE, OMBDROP EXPERT_MODULE

OMBCREATE EXTERNAL_TABLE

Purpose

To create an external table.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
createExternalTableCommand = OMBCREATE ( EXTERNAL_TABLE "QUOTED_STRING" [
    SET "setPropertiesAndReferencesClauses" ] [
    "addExternalTableSCOClauses" ] )
setPropertiesAndReferencesClauses = "setPropertiesClause" [ SET ( REF |
    REFERENCE ) "setReferencesToRecordAndLocationClauses" ] | ( REF |
    REFERENCE ) "setReferencesToRecordAndLocationClauses"
addExternalTableSCOClauses = ADD ( "addExternalTableColumnClause" [
    "addExternalTableSCOClauses" ] | "addExternalTableDatafileClause" {
    ADD "addExternalTableDatafileClause" } )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferencesToRecordAndLocationClauses =
    "setReferencesToRecordFileModuleClause" [
    "setReferencesToLocationClause" ] [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | "setReferencesToFileAndModuleClause" [
    "setReferencesToLocationClause" ] [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | "setReferencesToLocationClause" [ SET
    ( REF | REFERENCE ) "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause"
addExternalTableColumnClause = COLUMN "QUOTED_STRING" [ SET
    "setPropertiesAndReferencesToFieldClauses" ]
addExternalTableDatafileClause = DATA_FILE "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferencesToRecordFileModuleClause = RECORD "QUOTED_STRING" OF
    FLAT_FILE "QUOTED_STRING"
setReferencesToLocationClause = DEFAULT_LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
setReferencesToFileAndModuleClause = FLAT_FILE "QUOTED_STRING"
setPropertiesAndReferencesToFieldClauses = "setPropertiesClause" [ SET (
    REF | REFERENCE ) "setReferencesToFieldClause" ] | ( REF | REFERENCE )
    "setReferencesToFieldClause"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferencesToFieldClause = FIELD "QUOTED_STRING"
```

Keywords And Parameters

createExternalTableCommand

Create a new external table.

QUOTED_STRING

The name of the new external table.

setPropertyAndReferencesClauses

Set the properties and/or flat file reference of the external table.

addExternalTableSCOClauses

Add columns and/or data files to the external table.

setPropertyClause

Set specified properties of the external table.

setReferencesToRecordAndLocationClauses

Set the referenced record and/or default location.

addExternalTableColumnClause

Add an external table column.

addExternalTableDatafileClause

Add a data file to the external table.

propertyNameList

The names of the properties whose values you want to set.

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for EXTERNAL_TABLE:

Name: BAD_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the bad file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: BAD_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the bad file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

When deployable is set to true, a script to create an External Table is generated.

Name: DISCARD_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the discard file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: DISCARD_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the discard file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: ENDIAN

Type: STRING

Valid Values: BIG, LITTLE, PLATFORM

Default: PLATFORM

Data endian should be platform default, little or big. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LOAD_NULLS_WHEN_MISSING_VALUES

Type: BOOLEAN

Valid Values: true, false

Default: false

If TRUE, then NULLs are loaded for any missing values in the record. If FALSE, then records with missing values are rejected. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: LOG_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the log file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: LOG_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the log file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: NLS_CHARACTERSET

Type: STRING

Valid Values: N/A

Default: "

NLS Characterset of the file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: NUMBER_OF_REJECTS_ALLOWED

Type: NUMBER

Valid Values: 0 - 2147483647

Default: 0

The number of rejects allowed before processing is terminated.

Name: PARALLEL_ACCESS_DRIVERS

Type: NUMBER

Valid Values: 1 - 63999

Default: 1

The number of parallel access drivers to enable.

Name: PARALLEL_ACCESS_MODE

Type: BOOLEAN

Valid Values: true, false

Default: false

Enable or disable parallel processing.

Name: REJECTS_ARE_UNLIMITED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable or disable limiting the number of rejected records.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: STRING_SIZES_IN

Type: STRING

Valid Values: BYTES, CHARACTERS

Default: BYTES

String sizes are in bytes or characters. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: TRIM

Type: STRING

Valid Values: BOTH, LEFT, NONE, RIGHT, SQL*LOADER

Default: NONE

Specification from trim option on input fields. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Properties for DATA_FILE:

Name: DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

The location of this data file for the external table.

Name: DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of this data file.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList

The values for the named properties.

setReferencesToRecordFileModuleClause

Specify the record and full path to the flat file for the external table to reference.

setReferencesToLocationClause

The name of the external table's default location.

setReferencesToFileAndModuleClause

Specify the full path to the flat file for the external table to reference.

setPropertyAndReferencesToFieldClauses

Set the properties and/or field reference of the external table column.

propertyValue

A property value.

setReferencesToFieldClause

Set the name of the field which the external table column references.

Examples

```
OMBCREATE EXTERNAL_TABLE 'SRC_TABLE' SET REFERENCE RECORD 'REC_1'
OF
```

FLAT_FILE '../SRC_FILES/FILE_1'

This will create an external table named "SRC_TABLE" based upon the record "REC_1" of flat file "FILE_1" of flat file module "SRC_FILES".

ombcreate_external_table\$createExternalTableCommand =

See Also

OMBCREATE, OMBALTER EXTERNAL_TABLE, OMBDROP EXTERNAL_TABLE, OMBRETRIEVE EXTERNAL_TABLE

OMBCREATE FLAT_FILE

Purpose

To create a flat file.

Prerequisites

Should be in the context of a flat file module.

Syntax

```
createFlatFileCommand = OMBCREATE ( FLAT_FILE "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] { "addRecordClause" } )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addRecordClause = ADD ( RECORD "QUOTED_STRING" [ SET "setPropertiesClause"
    ] { "addFieldClause" } )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addFieldClause = ADD FIELD "QUOTED_STRING" [ SET "setPropertiesClause" ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`createFlatFileCommand`

Create a new flat file object.

`QUOTED_STRING`

The name of the new flat file.

`setPropertiesClause`

Set the properties of the flat file, record, or field.

`addRecordClause`

Add a record to the flat file.

`QUOTED_STRING`

The name of the new record.

`propertyNameList`

The names of the properties whose values you want to set.

Properties for FLAT_FILE:

Name: DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default "

The name of the "sampled" file. Also the default data file value used in SQL*Loader maps and External Tables.

Name: IS_DELIMITED

Type: BOOLEAN

Valid Values: true, false, 1, 0

Default: true

True indicates that this flat file is delimited. False indicates that its fields are defined by fixed lengths

Name: CHARACTERSET

Type: STRING

Valid Values:

AL24UTFFSS,AR8ARABICMAC,AR8ARABICMACS,AR8ISO8859P6,AR8MSAWIN,AR8MSWIN1256,BLT8CP921,BLT8EBCDIC1112,BLT8MSWIN1257,BLT8PC775,CDN8PC863,CL8EBCDIC1025,CL8EBCDIC1025X,CL8ISO8859P5,CL8KOI8R,CL8MACCYRILLIC,CL8MACCYRILLICS,CL8MSWIN1251,D8EBCDIC273,DK8EBCDIC277,EE8EBCDIC870,EE8ISO8859P2,EE8MACCE,EE8MACCES,EE8MACCROATIAN,EE8MACCROATIANS,EE8MSWIN1250,EE8PC852,EL8EBCDIC875,EL8ISO8859P7,EL8MACGREEK,EL8MACGREEKS,EL8MSWIN1253,EL8PC437S,EL8PC737,EL8PC869,F8EBCDIC297,I8EBCDIC280,IS8MACICELANDIC,IS8MACICELANDICS,IS8PC861,IW8EBCDIC424,IW8ISO8859P8,IW8MACHEBREW,IW8MACHEBREWS,IW8MSWIN1255,JA16EBCDIC930,JA16EUC,JA16EUCYEN,JA16MACSJIS,JA16SJIS,JA16SJISYEN,JA16VMS,KO16KSC5601,LT8MSWIN921,N8PC865,NEE8ISO8859P4,RU8PC855,RU8PC866,S8EBCDIC278,SE8ISO8859P3,TH8MACTHAI,TH8MACTHAIS,TH8TISASCII,TR8EBCDIC1026,TR8MAC TURKISH,TR8MACTURKISHS,TR8MSWIN1254,TR8PC857,US7ASCII,US8PC437,UTF8,WE8EBCDIC284,WE8EBCDIC285,WE8EBCDIC37,WE8EBCDIC37C,WE8EBCDIC500,WE8EBCDIC500C,WE8EBCDIC871,WE8ISO8859P1,WE8ISO8859P9,WE8MACROMAN8,WE8MACROMAN8S,WE8MSWIN1252,WE8PC850,WE8PC860,ZHS16CGB231280,ZHS16GBK,ZHS16MACCGB231280,ZHT16BIG5,ZHT16MSWIN950,ZHT32EUC

Default: WE8MSWIN1252

The character set of the data file.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

The character(s) which denote the end of a physical record in a data file.

A hex value may be entered by entering embedded single quotes twice as:

'x'0f'" (all are single quotes). The outside single quote indicates a quoted string and the inside single quotes single-quote x single-quote single-quote 0F single-quote single-quote single-quote. (Please note that this is not the FIELD_DELIMITER.

Name: RECORD_LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0 (Records are delimited by default)

The length (in characters) of the records in the data file.

Name: RECORD_TYPE_COLUMN_NUMBER

Type: NUMBER

Valid Values: 0+

Default: 0

The column which contains the record type values for a delimited, multi-record type file.

Name: RECORD_TYPE_START_POSITION

Type: NUMBER

Valid Values: 0+

Default: 0

The starting position of the field (relative to 0) which contains the record type values for a fixed-length, multi-record type file.

Name: RECORD_TYPE_END_POSITION

Type: NUMBER

Valid Values: 0+

Default: 0

The ending position of the field (relative to 0) which contains the record type values for a fixed-length, multi-record type file.

Name: NUMBER_OF_RECORDS_TO_SKIP

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The default number of records to skip when loading this file.

Name: FIELD_DELIMITER

Type: STRING

Valid Values: Any single character

Default: ',' (Comma)

The character to divide the fields in a delimited file.

Name: FIELD_LEFT_ENCLOSURE

Type: STRING

Valid Values: Any single character

Default: None

A character to enclose fields which may contain the field delimiter.

Name: FIELD_RIGHT_ENCLOSURE

Type: STRING

Valid Values: Any single character

Default: None

A character to enclose fields which may contain the field delimiter.

Name: NUMBER_OF_PHYSICAL_RECORDS_PER_LOGICAL

Type: Number

Valid Values: 0+

Default: 0

Set this value if you wish to concatenate a fixed number of physical records to form a single logical record.

Name: CONTINUE_IF_ENDS_WITH

Type: STRING

Valid Values: Any single character

Default: None

Set this value if you wish to concatenate a variable number of physical records to form a single logical record, determined by records ending with this character.

Name: CONTINUE_IF_STARTS_WITH

Type: STRING

Valid Values: N/A

Default: None

Set this value if you wish to concatenate a variable number of physical records to form a single logical record, determined by records beginning with this character.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the record

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the record

Properties for RECORD:

Name: RECORD_TYPE_VALUE

Type: STRING

Valid Values: N/A

Default: None

This is a mandatory property for each record of a multi-record type file.

It is the string which will identify this record type in the data file.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the record

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the record

Properties for FIELD:

Name: DATATYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARRAW,

VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

This is the SQL*Loader data type for the field.

Name: MAXIMUM_LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0

This is the maximum length of the field.

Name: LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0

Deprecated. This is the length of the field in a fixed length file. This is the max length of the field in a delimited file.

Name: PRECISION

Type: NUMBER

Valid Values: Depends on data type

Default: 0

Precision of the field.

Name SCALE

Type: NUMBER

Valid Values: Depends on data type

Default: 0

Scale of the field

Name: START_POSITION

Type: NUMBER

Valid Values: 1+

Default: 1

The starting position of a field for a fixed length file.

Name: END_POSITION

Type: NUMBER

Valid Values: 1+

Default: 1

The ending position of a field for a fixed length file.

Name: SQL_DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: DEFAULT. This will derive the SQL_DATATYPE from the value of DATATYPE.

The data type which the field will be treated as in mapping and for External Tables.

Name: SQL_LENGTH

Type: NUMBER

Valid Values: 1 - 4000

Default: 0

Name: SQL_PRECISION

Type: NUMBER

Valid Values: 1 - 38

Default: 1

Name: SQL_SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 0

Name: MASK

Type: STRING

Valid Values: N/A

Default: None

This is the mask used to define the format of DATE fields in the data file.

Name: NULL_IF

Type: STRING

Valid Values: Either = or != followed by either 'BLANKS', a single quoted string, or a hexadecimal string

Default: None

If this condition is true for a field, the value loaded will be NULL.

Name: DEFAULT_IF

Type: STRING

Valid Values: Either = or != followed by either 'BLANKS', a single quoted string, or a hexadecimal string

Default: None

If this condition is true for a field, the value loaded will be either NULL or 0, dependent on data type.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the field

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the field

propertyValueList

The values for the named properties.

addFieldClause

Add a field to the record.

QUOTED_STRING

The name of the new field.

propertyValue

A property value.

Examples

```

OMBCREATE FLAT_FILE 'SRC_FILE' \
SET PROPERTIES (DATA_FILE_NAME, FIELD_DELIMITER, FIELD_LEFT_
ENCLOSURE,
FIELD_RIGHT_ENCLOSURE, RECORD_TYPE_COLUMN_NUMBER) \
VALUES ('src_data.dat', '|', '{', '}', 1) \
ADD RECORD 'REC1' \
SET PROPERTIES (RECORD_TYPE_VALUE) VALUES ('E') \
ADD FIELD 'F1' \
SET PROPERTIES (DATATYPE, MASK) VALUES ('DATE', 'dd-mon-yyyy') \
ADD FIELD 'F2' \
ADD RECORD 'REC2' \
SET PROPERTIES (RECORD_TYPE_VALUE) VALUES ('P') \
ADD FIELD 'F1' \
ADD FIELD 'F2' \
ADD FIELD 'F3'

```

This will create a flat file named "SRC_FILE" with multiple record types.

See Also

OMBCREATE, OMBALTER FLAT_FILE, OMBDROP FLAT_FILE

OMBCREATE FLAT_FILE_MODULE

Purpose

To create a flat file module.

Prerequisites

Should be in the context of project.

Syntax

```
createFlatFileModuleCommand = OMBCREATE ( FLAT_FILE_MODULE "QUOTED_STRING"
    [ SET ( "setPropertiesClause" [ SET
        "setReferenceClauseForDataMetadataModule" ] |
        "setReferenceClauseForDataMetadataModule" ) ] [
        "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
```

Keywords And Parameters

createFlatFileModuleCommand

Create a new flat file module.

QUOTED_STRING

The name for the new flat file module.

setPropertiesClause

Set specified properties of the new flat file module.

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the new flat file module.

addModuleReferenceLocationClause

Add runtime locations to the new flat file module.

propertyNameList

The names of the properties whose values you want to set.

Basic properties for FLAT_FILE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of the flat file module.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the flat file module.

propertyValueList

The values for the named properties.

setReferenceLocationClause

Set a runtime location to the new flat file module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the new flat file module.

addReferenceLocationClause

Add a runtime location to the new flat file module.

propertyValue

A property value.

setReferenceMetadataLocationClause

Set metadata location for the new flat file module.

setReferenceIconSetClause

Set icon set for the new flat file module.

Examples

```
OMBCREATE FLAT_FILE_MODULE 'src_module' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('this is a flat file module', 'source module')
```

This will create a flat file module named "src_module", its description is "this is a flat file module", and business name is "source module".

See Also

OMBCREATE, OMBALTER FLAT_FILE_MODULE, OMBDROP FLAT_FILE_MODULE

OMBCREATE FUNCTION

Purpose

To create a Function.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
createFunctionCommand = OMBCREATE ( FUNCTION "QUOTED_STRING" ( [ SET (
    "setPropertyClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] ) { ADD ( "addFuncProcParameterClause"
    | "addRelationalDependentClause" ) } )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addFuncProcParameterClause = PARAMETER "QUOTED_STRING" [ SET
    "setPropertyClause" ]
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createFunctionCommand

This command creates a Function

QUOTED_STRING

Name of the Function to be created.

setPropertyClause

Used to set properties (core, user-defined) for function. Valid properties are as shown:

Basic properties for FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Function

Name: RETURN_TYPE

Type: STRING

Valid Values: PLS_INTEGER, BINARY_INTEGER, BOOLEAN, NUMBER, FLOAT, CHAR,

VARCHAR, VARCHAR2, DATE

Default: NUMBER

Set the Return Type for Function

Name: IMPLEMENTATION

Type: STRING

Valid Values: N/A

Default: "

Set the code for Function which is included global variable declaration and code between BEGIN and END.

Name: IS_DETERMINISTIC

Type: BOOLEAN

Valid Values: true, false

Default: false

This setting helps the optimizer avoid redundant function calls.

Name: IS_PARALLEL_ENABLE

Type: BOOLEAN

Valid Values: true, false

Default: false

This option sets flag to a stored function can be used safely in the slave

sessions of parallel DML evaluations.

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB,
BOOLEAN,

CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL
YEAR TO

MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW,
TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,
VARHCHAR,

VARCHAR2, XMLTYPE

Default: NUMBER

Set the data type for Parameter

Name: IN_OUT

Type: STRING

Valid Values: IN, OUT, INOUT

Default: 'IN'

Set the parameter mode for Parameter

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Set the default value for Parameter

Properties for FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addFuncProcParameterClause

Adds one or more Parameters to this Function.

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE FUNCTION 'func' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME,
```

```
RETURN_TYPE, IMPLEMENTATION, IS_DETERMINISTIC, IS_PARALLEL_
ENABLE) VALUES
```

```
('this is a Function', 'function', 'NUMBER', 'BEGIN RETURN 1 END func \;',
'true', 'true')
```

```
ADD PARAMETER 'PARAM_1'
```

```
SET PROPERTIES (DESCRIPTION, BUSINESS_NAME, IN_OUT, DATATYPE,
DEFAULT_VALUE) VALUES ('param_1', 'this is a param_1','IN', 'VARCHAR2',
'this is a Varchar2')
```

```
ADD PARAMETER 'PARAM_2'
```

```
SET PROPERTIES (DESCRIPTION, BUSINESS_NAME, IN_OUT, DATATYPE,
DEFAULT_VALUE) VALUES ('param_2', 'this is a param_2','INOUT', 'DATE',
'this is a Date')
```

This will create a Function named "func", its description is "this is a Function", and business name is "function", return datatype NUMBER, and body of function as 'BEGIN RETURN 1 END func;'. It creates two parameters 'PARAM_1' and 'PARAM_2'

See Also

OMBCREATE, OMBALTER FUNCTION, OMBDROP FUNCTION

OMBCREATE GATEWAY_MODULE

Purpose

To create a Gateway module.

Prerequisites

Should be in the context of project.

Syntax

```
createGatewayModuleCommand = OMBCREATE ( GATEWAY_MODULE "QUOTED_STRING" (
    SET "setPropertiesClause" ) [ SET
        "setReferenceClauseForDataMetadataModule" ] [
        "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
```

Keywords And Parameters

createGatewayModuleCommand

This command creates a Gateway module

QUOTED_STRING

Name of the Gateway module to be created.

setPropertiesClause

Associate a set of properties with the Gateway module.

Basic properties for GATEWAY_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of an Oracle Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of an Oracle Module

Name: GATEWAY_TYPE

Type: STRING

Valid Values: N/A

Default: N/A

Type of gateway module. Supported values are: 'DB2 Gateway Module', 'Sybase Gateway Module', 'Sql*server Gateway Module', 'Informix Gateway Module', 'ODBC Gateway Module', 'DRDA Gateway Module', 'RDB Gateway Module', 'Teradata Gateway Module', 'Other Gateway Module'.

Properties for GATEWAY_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for referenced objects

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the gateway module.

addModuleReferenceLocationClause

Add runtime locations to the gateway module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a runtime location to the gateway module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the gateway module.

addReferenceLocationClause

Add a runtime location to the gateway module.

propertyValue

Value of a property.

setReferenceMetadataLocationClause

Set metadata location for the gateway module.

setReferenceIconSetClause

Set icon set for the gateway module.

Examples

```
OMBCREATE GATEWAY_MODULE 'db2_module' SET PROPERTIES (GATEWAY_
TYPE) VALUES
```

```
('DB2 Gateway Module')
```

This will create a gateway module named "db2_module", and its type is DB2.

See Also

OMBCREATE, OMBALTER GATEWAY_MODULE, OMBDROP GATEWAY_MODULE

OMBCREATE ICONSET

Purpose

To create an icon set.

Prerequisites

In any context.

Syntax

```
createIconSetCommand = OMBCREATE ( ICONSET "QUOTED_STRING" [ SET
    "setPropertyClause" ] )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createIconSetCommand

This command creates an iconset.

QUOTED_STRING

Specify the name of the iconset to be created.

setPropertyClause

This clause sets a list properties to the specified values.

Basic properties for ICONSET:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the iconset

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the iconset

Name: BELONGS_TO_GROUP

Type: STRING

Valid Values: N/A

Default: "

Name of the Group to which the iconset belongs

Name: CANVAS_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the canvas icon (36x36)

Name: PALETTE_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the palette icon (18x18)

Name: TREE_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the tree icon (16x16)

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

A property value.

Examples

```
OMBCREATE ICONSET 'NEW_ICONSET' SET PROPERTIES (DESCRIPTION,
BELONGS_TO_GROUP, CANVAS_ICON, PALETTE_ICON, TREE_ICON) VALUES
('this is a
```

```
new iconset', 'Tasks', 'canvas_icon.gif', 'palette_icon.gif',  
'tree_icon.gif')
```

See Also

OMBCREATE ICONSET, OMBALTER ICONSET, OMBDROP ICONSET

OMBCREATE IMPORT_ACTION_PLAN

Purpose

To create a transient import action plan.

Prerequisites

In the context of a project.

Syntax

```
createImportActionPlanCommand = ( OMBCREATE TRANSIENT IMPORT_ACTION_PLAN
  "QUOTED_STRING" { "addActionClause" } )
addActionClause = ADD ACTION "QUOTED_STRING" ( "setPropertyClause" [
  "setRefSourceAndTargetClause" ] | "setRefSourceAndTargetClause" )
setPropertyClause = SET PROPERTIES "(" "propertyNameList" ")" VALUES "("
  "propertyValueList" ")"
setRefSourceAndTargetClause = SET ( REF | REFERENCE ) "sourcesClause" SET
  ( REF | REFERENCE ) "targetClause"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
sourcesClause = SOURCE "ObjType" "QUOTED_STRING" [ SET ( REF | REFERENCE )
  "sourcesClause" ]
targetClause = TARGET "ObjType" "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
  "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`createImportActionPlanCommand`

This command is for creating a transient import action plan, which is composed of a set of import actions. Each import action includes specification of source objects from which metadata will be extracted and one target object, into which the metadata will be added.

`QUOTED_STRING`

The name of the import action plan to be created.

`addActionClause`

For adding an action to the import action plan.

`QUOTED_STRING`

The name of the action to be added.

`setPropertyClause`

For setting any properties for the import action. For the current release, there are no predefined property for import actions.

setRefSourceAndTargetClause

For specifying source and target objects for the import action. The source objects are to be imported into target.

sourcesClause

For specifying source objects in an import action. Valid object types are as follows.

TABLE

VIEW

SEQUENCE

MATERIALIZED_VIEW

EXTERNAL_TABLE

ADVANCED_QUEUE

QUEUE_TABLE

OBJECT_TYPE

VARRAY

NESTED_TABLE

DIMENSION

CUBE

FUNCTION

PROCEDURE

PACKAGE

QUOTED_STRING

The name of the source object. Note that the name of the source object must be qualified with schema name, such as 'SCOTT.EMP'.

For importing into a transportable module, the object specification requires an optional tablespace name qualifier, such as 'USERS.SCOTT.EMP' or 'DEFAULT.SCOTT.EMP'. The tablespace name is required because schemas are listed under tablespaces in the transportable module tree in OWB designer console. If 'DEFAULT' is specified as tablespace name, then the default tablespace name of the schema is used. If the tablespace name is omitted, then the following rule is used to deduce the appropriate tablespace name. First, if the object is a database segment (that is, table and materialized view), then use the tablespace name of the tablespace where the segment resides.

Second if the object is not a database segment, then the default tablespace name for the schema is used, that is, it is equivalent to specifying 'DEFAULT' as tablespace name.

targetClause

For specifying the target object in an import action. There can only be one target object in an import action.

QUOTED_STRING

The name of the target object.

Examples

```
OMBCREATE TRANSIENT IMPORT_ACTION_PLAN 'PLAN1'  
ADD ACTION 'A1'  
SET REF SOURCE TABLE 'SCOTT.T1'  
SET REF SOURCE VIEW 'SCOTT2.V1'  
SET REF TARGET TRANSPORTABLE_MODULE 'TM101'
```

This command will create an import plan named PLAN1, which contains one import action named A1. The import action is for importing the metadata definitions of table T1 in schema SCOTT and view V1 in schema SCOTT2 into transportable module TM101.

See Also

OMBIMPORT

OMBCREATE ITEM_FOLDER

Purpose

Creates an item folder that can be used in a business view.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createItemFolderCommand = ( OMBCREATE ITEM_FOLDER "QUOTED_STRING" [ SET
    "setPropertyClauseDelayed" ] [ SET "setReferenceIconSetClause" ] {
    "addItemFolderSCOClauses" } )
setPropertyClauseDelayed = PROPERTIES "(" "propertyNameListVector" ")"
    VALUES "(" "propertyValueListVector" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addItemFolderSCOClauses = ADD ( "addItemClause" | "addConditionClause" |
    "addJoinClause" )
propertyNameListVector = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueListVector = "propertyValue" { ", " "propertyValue" }
addItemClause = ITEM "QUOTED_STRING" [ SET "setPropertyClause" ] [ SET (
    REF | REFERENCE ) ( "ItemItemReferencesClause" |
    "ItemListOfValuesReferencesClause" |
    "ItemDrillToDetailReferencesClause" |
    "ItemAlternativeSortOrderReferencesClause" |
    "ItemColumnReferencesClause" ) ]
addConditionClause = CONDITION "QUOTED_STRING" [ SET "setPropertyClause"
    ]
addJoinClause = JOIN "QUOTED_STRING" [ SET "setPropertyClause" ] [ SET (
    REF | REFERENCE ) "JoinForeignKeyReferencesClause" ] {
    "joinComponentClause" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
ItemItemReferencesClause = ITEM "QUOTED_STRING" OF ITEM_FOLDER
    "QUOTED_STRING" "itemJoinUsages"
ItemListOfValuesReferencesClause = LIST_OF_VALUES "QUOTED_STRING"
ItemDrillToDetailReferencesClause = DRILL_TO_DETAIL "QUOTED_STRING"
ItemAlternativeSortOrderReferencesClause = ALTERNATIVE_SORT_ORDER
    "QUOTED_STRING"
ItemColumnReferencesClause = COLUMN "QUOTED_STRING" OF ( TABLE | (
    EXTERNAL_TABLE | VIEW ) ) "QUOTED_STRING"
JoinForeignKeyReferencesClause = FOREIGN_KEY "QUOTED_STRING" OF ( TABLE |
    VIEW ) "QUOTED_STRING"
joinComponentClause = ADD JOIN_COMPONENT "QUOTED_STRING" [ SET
    "setPropertyClause" ] { SET ( REF | REFERENCE )
    "setJoinComponentClauseDetails" }
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
itemJoinUsages = { SET ( REF | REFERENCE ) USING JOIN "QUOTED_STRING" OF
    ITEM_FOLDER "QUOTED_STRING" }
setJoinComponentClauseDetails = LOCAL ITEM "QUOTED_STRING" | REMOTE ITEM
    "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
```

Keywords And Parameters

createItemFolderCommand

This command creates an item folder.

QUOTED_STRING

Specify the name of the item folder to be created.

setPropertyClauseDelayed

This clause sets the properties.

setReferenceIconSetClause

Set specified Icon Set.

addItemFolderSCOClauses

This clause adds items to an item folder.

propertyNameListVector

This clause holds the names of the properties.

propertyValueListVector

This clause holds the values of the properties.

addItemClause

This clause adds an item to an item folder.

ITEM

A field within the item folder.

QUOTED_STRING

The name of the item to be added.

addConditionClause

This clause adds a condition to an item folder.

CONDITION

A field within the item folder.

QUOTED_STRING

The name of the condition to be added.

addJoinClause

This clause adds a foreign key relationship to another item folder.

JOIN

A foreign key relationship with another item folder.

QUOTED_STRING

The name of the join to be added.

propertyValue

This is a property value.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for item folders. Valid properties are as shown:

Basic properties for ITEM_FOLDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the item folder

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the item folder

Name: EXTERNAL_TABLE_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The physical name for the corresponding table or view. This is automatically set if the Folder is associated with a Table

Name: VISIBLE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the item folder should be visible to the user

Name: FOLDER_TYPE

Type: STRING(40)

Valid Values: SIMPLE, COMPLEX

Default: ''

The type of item folder

Basic properties for ITEM:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: ''

Business name of the item

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: ''

Description of the item

Name: ALIGNMENT

Type: STRING(40)

Valid Values: GENERAL, LEFT, CENTER, RIGHT

Default: 'GENERAL'

The default alignment for displaying the item

Name: DISPLAY_CASE

Type: STRING(40)

Valid Values: GENERAL, LOWER, UPPER, INITCAPPED

Default: 'GENERAL'

How alphabetic characters should be displayed

Name: CASE_STORAGE

Type: STRING(40)

Valid Values: GENERAL, LOWER, UPPER, MIXED

Default: 'GENERAL'

How alphabetic characters are stored

Name: CONTENT_TYPE

Type: STRING(40)

Valid Values: No Value or FILE. For datatypes such as BLOB, it may contain a file extension such as DOC, AVI, WAV, JPG

Default: "

Details on whether the Item contains a file name or should be processed by an external application

Name: DEFAULT_AGGREGATE

Type: STRING(255)

Valid Values: Detail, AVG, COUNT, MAX, MIN, SUM

Default: 'SUM' when the datatype is Numeric, 'Detail' otherwise

Name of the default rollup function for the item

Name: DEFAULT_POSITION

Type: STRING(40)

Valid Values: MEASURE, TOP OR SIDE, TOP, SIDE, PAGE

Default: 'MEASURE' when the datatype is NUMBER or FLOAT, 'TOP OR SIDE' otherwise

Default position for the item

Name: REPLACE_NULL_WITH

Type: STRING(255)

Valid Values: N/A

Default: "

The value to be displayed for null values

Name: FORMULA

Type: STRING

Valid Values: N/A

Default: "

The text of the derivation expression for a derived item

Name: EXTERNAL_COLUMN_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The external name of the corresponding column. This is automatically set if the Item is associated with a Column

Name: FORMAT_MASK

Type: STRING(255)

Valid Values: N/A

Default: "

The display format mask for the item

Name: HEADING

Type: STRING(255)

Valid Values: N/A

Default: "

The displayed heading text for the item

Name: DATATYPE

Type: STRING(40)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.ROW_LCR, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE

TIMESTAMP WITH TIME ZONE, UNSPECIFIED, VARCHAR, VARCHAR2, XMLTYPE,

SYS.XMLFORMAT, BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS, SYS.XMLSEQUENCETYPE,

SYS_REFCURSOR, BLAST_MATCH_PLSQLRECORDTYPE

Default: 'VARCHAR2'

The datatype for the item

Name: VISIBLE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the item should be visible to the user

Name: MAX_CHAR_FETCHED

Type: Number

Valid Values: N/A

Default: "

The maximum number of characters fetched for an item

Name: DEFAULT_WIDTH

Type: Number

Valid Values: N/A

Default: "

The default number of characters to display

Name: WORD_WRAP

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether wordwrap is allowed in the display

Basic properties for JOIN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the join

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the join

Name: OUTER_JOIN_ON_MASTER

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether detail rows with no related master row should be included in the join

Name: OUTER_JOIN_ON_DETAIL

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether master rows with no related detail rows should be included in the join

Name: EXTERNAL_KEY_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The external name of the corresponding foreign key. This is automatically set if the Join is associated with a Foreign Key

Name: DETAIL_ALWAYS_HAS_MASTER

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether every detail row must reference a unique master row

Name: ONE_TO_ONE

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether a master row only ever has a single detail row

Basic properties for JOIN_COMPONENT:

Name: JOIN_OPERATOR

Type: STRING(200)

Valid Values: =, <>, <, <=, > or >=

Default: "

Business name of the join

Basic properties for CONDITION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the condition

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the condition

Name: MATCH_CASE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the case of alphabetic characters must match exactly

Name: FORMULA

Type: STRING

Valid Values: N/A

Default: "

The expression for the condition

Name: MANDATORY

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether the Condition is optional or mandatory

Properties for ITEM_FOLDER:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for the referenced database object

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Name: OPTIMIZER_HINT

Type: STRING

Valid Values: N/A

Default: "

Optimizer Hint to be added when this Item Folder is used in a query

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

ItemItemReferencesClause

This clause is used to reference another item.

QUOTED_STRING

name of the referenced item.

ItemListOfValuesReferencesClause

This clause is used to reference a list of values.

QUOTED_STRING

name of the list of values.

ItemDrillToDetailReferencesClause

This clause is used to reference a drill to detail.

QUOTED_STRING

name of the drill to detail.

ItemAlternativeSortOrderReferencesClause

This clause is used to reference an alternative sort order.

QUOTED_STRING

name of the alternative sort order.

ItemColumnReferencesClause

This clause is used to reference a column.

QUOTED_STRING

name of the referenced column.

JoinForeignKeyReferencesClause

The foreign key reference.

joinComponentClause

The join components.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

itemJoinUsages

The specific joins to be used.

setJoinComponentClauseDetails

The structure of the join component.

Examples

```
OMBCREATE ITEM_FOLDER 'SALES'
```

See Also

OMBALTER ITEM_FOLDER, OMBRETRIEVE ITEM_FOLDER

OMBCREATE LIST_OF_VALUES

Purpose

Creates a List of Values.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createListOfValuesCommand = OMBCREATE LIST_OF_VALUES "QUOTED_STRING" [ SET
    "setPropertiesClauseforLOVandD2D" ] [ SET "setReferenceIconSetClause"
    ] [ "addListOfValuesClause" ] { "addListOfValuesReferenceClause" }
setPropertiesClauseforLOVandD2D = PROPERTIES "("
    "propertyNameListforLOVandD2D" ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addListOfValuesClause = SET ( REF | REFERENCE ) DEFINING ITEM
    "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
addListOfValuesReferenceClause = SET ( REF | REFERENCE ) ITEM
    "QUOTED_STRING" OF ITEM_FOLDER "QUOTED_STRING"
propertyNameListforLOVandD2D = ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) {
    ", " ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) }
propertyValueList = "propertyValue" { ", " "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createListOfValuesCommand

This command creates a list of values.

QUOTED_STRING

Specify the name of the list of values to be created.

setPropertiesClauseforLOVandD2D

Used to set properties (core, logical, physical, user-defined) for a list of values. Valid properties are as shown:

Basic properties for LIST_OF_VALUES:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the list of values

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the list of values

Name: DRILL_TO_DETAIL

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the list of values enables drilling between the item folders containing the items that use the list of values

Name: RETRIEVE_VALUES_GROUP_SIZE

Type: Number

Valid Values: N/A

Default: '100'

The number of rows to be fetched from the database at a time

Name: CACHE_VALUES

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the list of values should be cached in memory

Name: REQUIRE_SEARCH

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether search criteria should be requested

Name: SHOW_IN_NAVIGATOR

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be shown in the item navigator

Name: SORTED_DISTINCT

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be displayed sorted with duplicates hidden

Properties for LIST_OF_VALUES:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceIconSetClause

Set specified Icon Set.

addListOfValuesClause

This clause modifies a list of values.

addListOfValuesReferenceClause

This adds a reference to an item to a list of values.

propertyNameListforLOVandD2D

This is the list of property names.

propertyValueList

This is the list of property values.

propertyValue

This is a property value.

Examples

OMBCREATE LIST_OF_VALUES 'SALES_ITEM'

See Also

OMBALTER LIST_OF_VALUES, OMBRETRIEVE LIST_OF_VALUES

OMBCREATE LOCATION

Purpose

To create a location.

Prerequisites

Can be in any context.

Syntax

```
createLocationCommand = OMBCREATE ( LOCATION "QUOTED_STRING" (
    "createLocationSetPropertiesClause" [ SET "setReferenceIconSetClause"
    ] ) )
createLocationSetPropertiesClause = SET PROPERTIES "(" "propertyNameList"
    ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createLocationCommand

Create a new location.

createLocationSetPropertiesClause

Set specified properties of the new location.

setReferenceIconSetClause

Set the Icon Set for the new location.

propertyNameList

The names of the properties whose values you want to set.

Properties for LOCATION:

Basic properties:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the location.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the location.

Name: TYPE

Type: STRING

Valid Values:

'CONCURRENT_MANAGER'

'AUTOSYS_AGENT'

'AUTOSYS_INSTANCE'

'BIBEANS'

'DISCOVERER'

'FILE_SYSTEM'

'OEM_AGENT'

'ORACLE_DATABASE'

'ORACLE_GATEWAY'

'ORACLE_WORKFLOW'

'SAP'

'TRANSPORTABLE_MODULE_SOURCE'

'TRANSPORTABLE_MODULE_TARGET'

Default: N/A

The type of system the location represents.

Name: VERSION

Type: STRING

Valid Values:

for 'CONCURRENT_MANAGER' : '11i'

for 'AUTOSYS_AGENT' : '0'

for 'AUTOSYS_INSTANCE' : '0'

for 'BIBEANS' : '10.1'

for 'DISCOVERER' : '10.1'

for 'FILE_SYSTEM' : do not set version

for 'OEM_AGENT' : '9.0','9.2'

for 'ORACLE_DATABASE' : '8.1','9.0','9.2','10.1','10.2'

for 'ORACLE_GATEWAY' : do not set version

for 'ORACLE_WORKFLOW' : '2.6.2','2.6.3','2.6.4','11i'

for 'SAP' : '4.x','3.x'

for 'TRANSPORTABLE_MODULE_SOURCE' : '8.1','9.0','9.2','10.1','10.2'

for 'TRANSPORTABLE_MODULE_TARGET' : '8.1','9.0','9.2','10.1','10.2'

Default: N/A

The version of the system(s) the location represents.

Lists of available properties for different types of LOCATION:

for 'CONCURRENT_MANAGER' :

TYPE,PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA,VERSION,APPLICATION,APPLICATION_USER,RESPONSIBILITY

for 'AUTOSYS_AGENT':

TYPE,VERSION,PASSWORD,HOST

for 'AUTOSYS_INSTANCE':

TYPE,VERSION,USER (or USER_NAME),PASSWORD,INSTANCE

for 'BIBEANS':

TYPE,VERSION,USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME

for 'DISCOVERER':

TYPE,VERSION,USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME

for 'FILE_SYSTEM':

TYPE,USER (or USER_NAME),PASSWORD,HOST,ROOTPATH

for 'OEM_AGENT':

TYPE,USER (or USER_NAME),PASSWORD,VERSION,DOMAIN,AGENT

for 'ORACLE_DATABASE':

TYPE,VERSION,CONNECT_AS_USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,DATABASE_NAME,SCHEMA

for 'ORACLE_GATEWAY':

TYPE,CONNECT_AS_USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA

for 'ORACLE_WORKFLOW':

TYPE,VERSION,PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA

for 'SAP':

TYPE, VERSION, USER (or USER_NAME), PASSWORD, APPLICATION_SERVER, SYSTEM_NUMBER, CLIENT, LANGUAGE, HOST_LOGIN_USER, HOST_LOGIN_PASSWORD,

FTP_DIRECTORY, EXECUTION_FM

for 'TRANSPORTABLE_MODULE_SOURCE':

TYPE, VERSION,CONNECT_AS_USER (or USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,FTP_USER,FTP_PASSWORD

for 'TRANSPORTABLE_MODULE_TARGET':

TYPE, VERSION,CONNECT_AS_USER (or USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME

Some other properties for LOCATIONS:

Name: CONNECTION_TYPE

Type: STRING

Valid Values: 'HOST_PORT_SERVICE', 'SQL_NET_CONNECTION', 'DATABASE_LINK'

Default: 'HOST_PORT_SERVICE'

The location connection details format.

Name: HOST

Type: STRING

Valid Values: N/A

Default: N/A

The machine name.

Name: PORT

Type: NUMBER

Valid Values: 1 - 65535

Default: 1521

The port number of a database listener.

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The database service name.

Name: NET_SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The database netservice name.

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: N/A

The database schema name.

Name: PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: CONNECT_AS_USER

Synonym: USER_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: DOMAIN

Type: STRING

Valid Values: N/A

Default: N/A

The address of a machine running the Oracle Management Service.

Name: AGENT

Type: STRING

Valid Values: N/A

Default: N/A

The name of an Oracle Enterprise Manager (OEM) node running an OEM Agent.

This name must be entered exactly as shown under the nodes in the Oracle Management Service.

Name: ROOTPATH

Type: STRING

Valid Values: N/A

Default: N/A

The file system directory.

Name: APPLICATION

Type: STRING

Valid Values: N/A

Default: N/A

The Application name.

Name: APPLICATION_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: DATABASE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The Data Base name.

Name: RESPONSIBILITY

Type: STRING

Valid Values: N/A

Default: N/A

The responsibility role.

Name: APPLICATION_SERVER

Type: STRING

Valid Values: N/A

Default: N/A

The application server.

Name: SYSTEM_NUMBER

Type: STRING

Valid Values: N/A

Default: N/A

The number of SAP system.

Name: CLIENT

Type: STRING

Valid Values: N/A

Default: N/A

The client.

Name: LANGUAGE

Type: STRING

Valid Values: N/A

Default: N/A

The language of SAP.

Name: HOST_LOGIN_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user.

Name: HOST_LOGIN_PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: EXECUTION_FM

Type: STRING

Valid Values: N/A

Default: N/A

RFC Function Module for remote ABAP report execution

Name: FTP_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name used for creating ftp connection.

Name: FTP_PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The ftp password.

Name: FTP_DIRECTORY

Type: STRING

Valid Values: N/A

Default: N/A

The directory used in a ftp session

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyValueList

The values for the named properties.

propertyValue

A property value.

Examples

```
OMBCREATE LOCATION 'MY_LOCATION' SET PROPERTIES (TYPE, VERSION,
DESCRIPTION, BUSINESS_NAME) VALUES ('ORACLE_DATABASE', '9.2','this is a
location', 'location')
```

This will create a location named "MY_LOCATION". Its type is "Oracle Database", version is "9.2", description is "this is a location", and business name is "location".

By default, the CONNECTION_TYPE for the newly created location is HOST:PORT:SERVICE. Other available values are: SQL*NET Connection, Database Link.

To create a dblink type location, it is user responsibility to issue a separate OMBCREATE CONNECTOR command. For example, to create a dblink type location "DBLINK_LOC", whose from location is "SCOTT_LOC", and database link name is "P1.US.ORACLE.COM", try following:

```
OMBCREATE LOCATION 'DBLINK_LOC' SET PROPERTIES
(TYPE,VERSION,CONNECTION_TYPE) VALUES('Oracle Database', '10.1', 'Database
Link')
```

```
OMBCREATE CONNECTOR 'SCOTT_LOCSOTT_LOC_TO_DBLINK_LOC' SET
PROPERTIES
(DATABASE_LINK_NAME) VALUES ('P1.US.ORACLE.COM') SET REF LOCATION
'SCOTT_LOC'.
```

See Also

OMBCREATE, OMBALTER LOCATION, OMBDROP LOCATION

OMBCREATE MAPPING

Purpose

Create a mapping in an Oracle Module.

Prerequisites

1. The current context of scripting must be an Oracle Module.
2. No concurrent user should be locking the Oracle Module or any of its ancestors exclusively at the moment the map is being created.
3. The map name must not conflict with existing map names and the maps names that concurrent user tries to use.

Syntax

```
createMappingCommand = OMBCREATE MAPPING "mappingName" ( [ AS (
    TRICKLE_FEED_MAPPING | BATCH_MAPPING ) ] )
    "createOperatorOwnerDetailClause"
mappingName = "QUOTED_STRING"
createOperatorOwnerDetailClause = [ SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] ) | "setReferenceIconSetClause" ) ]
    "createOperatorOwnerDescendantsClause"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
createOperatorOwnerDescendantsClause = { ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) }
propertyKeyList = "( " "propertyKey" { ", " "propertyKey" } )"
propertyValueList = "( " "propertyValue" { ", " "propertyValue" } )"
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
```



```

operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
attributeName = "QUOTED_STRING"
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { ", " IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
pluggableMapName = "QUOTED_STRING"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
attributeNameList = "(" "attributeName" { "," "attributeName" } ")"
bindableType = PLUGGABLE_MAPPING | OBJECT_TYPE | "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

createMappingCommand

Create a mapping in an Oracle Module.

mappingName

Name of the mapping.

createOperatorOwnerDetailClause

Create the desired detail of a pluggable mapping.

setPropertyClause

Describe the keys of properties for the map or objects in the map.

createOperatorOwnerDescendantsClause

Create the desired child objects of a mapping or a pluggable mapping.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

addChildClause

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

The following is an example for creating a child object under a mapping

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

Note: Key word "OPERATOR" "GROUP" "ATTRIBUTE" are important for their respective ADD clauses. Without the key words, OMBPlus will interpret the ADD clause as an addChildClause. Here is an example:

```
OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'
```

```
OMB02932: Error getting child objects of type TABLE in M1
```

OMBPLUS interprets the ADD clause as one for creating a non-operator child object under the mapping object. Therefore, it tries to find type definition for non-operator child object "TABLE" and cannot find it.

addConnectionClause

Add connections between mapping groups or mapping attributes.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0
Length of the attribute.

Name: PRECISION
Type: NUMBER
Valid Values: N/A
Default: 0
Precision of the attribute.

Name: SCALE
Type: NUMBER
Valid Values: N/A
Default: 0
Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION
Type: NUMBER
Valid Values: 0 - 9
Default: 0
The precision of a timestamp or interval.

Properties for MAPPING:

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true
True if the map is deployable to a physical implementation

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Name: GENERATION_LANGUAGE

Type: STRING

Valid Values: ABAP, PLSQL, SQLLOADER, UNDEFINED

Default: UNDEFINED

The language used when generating code for the mapping.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A
 Default: "
 The data type of the attribute

Name: DEFAULT_VALUE
 Type: STRING
 Valid Values: N/A
 Default: "
 The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME
 Type: STRING
 Valid Values: N/A
 Default: "
 The dimension attribute referenced to by this level attribute.

Name: EXPRESSION
 Type: STRING
 Valid Values: N/A
 Default: "
 The output expression for the attribute

Name: FIELD_DATA_TYPE
 Type: STRING
 Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC, VARRAW, VARRAWC, ZONED, ZONED EXTERNAL
 Default: CHAR
 SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE,

NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,

NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_
 NEIGHBORHOOD, NA_NONE,
 NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE,
 NA_POSTNAME,
 NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE
 Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_
 FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH
Type: NUMBER
Valid Values: N/A
Default: 0
The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME
Type: STRING
Valid Values: N/A
Default: "
Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME
Type: STRING
Valid Values: N/A
Default: "
Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME
Type: STRING
Valid Values: N/A
Default: "
The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME
Type: STRING
Valid Values: N/A
Default: "
The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW
Type: STRING(3)
Valid Values: NO, YES
Default: YES
A boolean value to indicate whether this attribute will participate in the

insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15, NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2, NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,
 NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,
 NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
 NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
 NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,
 NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA, NA_NAMEDSIGNATOR,
 NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE, NA_STREETCOMPCORRECTED,
 NA_STREETCOMPMATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR, NA_UNITNUMBER,
 NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4, NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup

Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE of SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | | =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before

loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should

be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name

as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT
 Type: STRING
 Valid Values: DELIMITED, FIXED
 Default: DELIMITED
 File Format (Fixed or Delimited).

Name: LOADING_TYPE
 Type: STRING(16)
 Valid Values: INSERT, NONE, UPDATE
 Default: INSERT
 The loading operation to be performed

Name: OUTPUT_AS_XML
 Type: BOOLEAN
 Valid Values: true, false
 Default: false
 Output data to file in XML format.

Name: RECORD_DELIMITER
 Type: STRING
 Valid Values: N/A
 Default: "
 Character that indicates the end of the record.

Name: RECORD_SIZE
 Type: NUMBER
 Valid Values: N/A
 Default: 0
 Size of a fixed length record.

Name: RECORD_TYPE_LENGTH
 Type: NUMBER
 Valid Values: N/A
 Default: 0
 If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type

Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"MATCHMERGE.GENERAL.MATCH_NEW_RECORD_
CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is

common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS,
NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will
be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some
transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false
 Default: false
 Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME
 Type: STRING
 Valid Values: N/A
 Default: "
 Exceptions Table Name

Name: EXTRACTION_HINT
 Type: STRING
 Valid Values: N/A
 Default: "
 Hint used when extracting from this table using SQL

Name: JOINRANK
 Type: FLOAT
 Valid Values: N/A
 Default: 0
 Join Rank

Name: LOADING_HINT
 Type: STRING
 Valid Values: N/A
 Default: "
 Hint used when loading into this table using SQL

Name: LOADING_TYPE
 Type: STRING(16)
 Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,
 NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT
 Default: INSERT
 The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A
 Default: "
 Sorted Indexes Clause

Name: SUBPARTITION_NAME
 Type: STRING
 Valid Values: N/A
 Default: "
 Subpartition Name

Name: TARGET_FILTER_FOR_DELETE
 Type: STRING
 Valid Values: N/A
 Default: "
 A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE
 Type: STRING
 Valid Values: N/A
 Default: "
 A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER
 Type: STRING(65535)
 Valid Values: N/A
 Default: "
 The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST
 Type: STRING
 Valid Values: N/A
 Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCAST, LCRSPLITTER,
MATCHMERGE,
MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER,
OUTPUT_SIGNATURE,
PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_
PROCESS,
SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION,
TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

attributeName

Name of a mapping attribute.

groupBottomUpLocator

Location of a mapping group.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributeBottomUpLocator

Location of a mapping attribute.

attributesBottomUpLocator

Location of a list of mapping attributes.

pluggableMapName

Name of the pluggable map.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBCREATE MAPPING 'MAP1'
```

```
OMBCREATE MAPPING 'MAP1'
```

```
SET PROPERTIES (business_name, description)
```

```
VALUES ('My map', 'Map to load customer look up table')
```

```
ADD VARIABLE 'LAST_CUST' SET PROPERTIES (SCALE, PRECISION) VALUES  
(10,20)
```

```
ADD TABLE OPERATOR 'CUST_SRC'
```

```
BOUND TO TABLE '../SRC_MODULE/CUST_SRC'
```

```
ADD TABLE OPERATOR 'CUST_LOOK_UP'
```

```
ADD CONNECTION FROM GROUP 'INOUTGRP1' OF OPERATOR 'CUST_SRC'  
TO GROUP 'INOUTGRP1' OF OPERATOR 'CUST_LOOK_UP'
```

See Also

OMBCREATE, OMBALTER MAPPING, OMBRETRIEVE MAPPING, OMBDROP MAPPING

OMBCREATE MATERIALIZED_VIEW

Purpose

To create a materialized view.

Prerequisites

In the context of an Oracle Module.

Syntax

```
createMaterializedViewCommand = OMBCREATE ( MATERIALIZED_VIEW
    "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET ( REF | REFERENCE
    ) "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] [
    "addMaterializedViewSCOandDependentClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
addMaterializedViewSCOandDependentClause = ADD ( "addColumnClause" |
    "addViewConstraintClause" | "addSCOClauses" |
    "addRelationalDependentClause" ) [
    "addMaterializedViewSCOandDependentClause" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addColumnClause = COLUMN "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ]
    [ SET "setPropertiesClause" ]
addViewConstraintClause = "addUkPkClause" | "addFkClause"
addSCOClauses = "addIndexClause" | "addIndexPartitionClause" |
    "addIndexPartitionKeyClause" | "addPartitionClause" |
    "addPartitionKeyClause" | "addSubpartitionClause" |
    "addaddMaterializedViewSCOandDependentClauseClause" |
    "addSubPartitionKeyClause" | "addIndexColumnClause"
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
addUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" [ SET
    "setUkPkPropertiesAndReferencesColumnsClauses" ]
addFkClause = FOREIGN_KEY "QUOTED_STRING" [ SET "setFkSubClauses" ]
addIndexClause = INDEX "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addIndexPartitionClause = INDEX_PARTITION "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] (
    "renameSCOConfigurationClause" [ SET
    "setSCOConfigurationPropertiesClauses" ] | [ SET
    "setSCOConfigurationPropertiesClauses" ] )
addIndexPartitionKeyClause = INDEX_PARTITION_KEY "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionClause = PARTITION "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionKeyClause = PARTITION_KEY "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addSubpartitionClause = SUBPARTITION "QUOTED_STRING" OF PARTITION
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
    "setSCOConfigurationPropertiesClauses" ]
addaddMaterializedViewSCOandDependentClauseClause = TEMPLATE_SUBPARTITION
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
```

```

    "setSCOConfigurationPropertiesClauses" ]
addSubPartitionKeyClause = SUBPARTITION_KEY "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addIndexColumnClause = INDEX_COLUMN "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
setUkPkPropertiesAndReferencesColumnsClauses = "setPropertiesClause" [ SET
    ( REF | REFERENCE ) "constraintColumnReferencesClause" ] | ( REF |
    REFERENCE ) "constraintColumnReferencesClause"
setFkSubClauses = "setPropertiesClause" [ SET ( REF | REFERENCE )
    "setFkReferencesClauses" ] | ( REF | REFERENCE )
    "setFkReferencesClauses"
setSCOConfigurationPropertiesClauses = PROPERTIES "(" "propertyNameList"
    ")" VALUES "(" "propertyValueList" ")"
renameSCOConfigurationClause = RENAME TO "QUOTED_STRING"
constraintColumnReferencesClause = COLUMNS "(" "quotedNameList" ")"
setFkReferencesClauses = "constraintColumnReferencesClause" [ SET ( REF |
    REFERENCE ) "constraintUkReferencesClause" ] |
    "constraintUkReferencesClause" [ SET ( REF | REFERENCE )
    "constraintColumnReferencesClause" ]
quotedNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
constraintUkReferencesClause = ( UNIQUE_KEY | PRIMARY_KEY )
    "QUOTED_STRING" [ OF ( TABLE | VIEW ) "QUOTED_STRING" ]

```

Keywords And Parameters

createMaterializedViewCommand

This command creates a materialized view.

QUOTED_STRING

Specify the name of the materialized view to be created.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for materialized views (including partitions and subpartitions) and their columns, indexes (including index partitions), unique keys, foreign keys, and primary keys.

Note:

Constraints can be specified but will not be generated for either View or Materialized View in this release.

Basic properties for VIEW, MATERIALIZED_VIEW:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the View, MaterializedView

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the View, MaterializedView

Name: VIEW_QUERY

Type: STRING(4000)

Valid Values: N/A

Default: "

Sets the query definition in View and MaterializedView.

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD,
SYS.XMLFORMAT,
TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME
ZONE,

VARHCAR, VARCHAR2, XMLTYPE

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Default value of the column

Name: NOT_NULL

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify "true" to enforce Not Null restriction on a column.

Basic properties for PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the table rows according to a Hash Algorithm, lists of values, or specified ranges.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH partitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of partitions that is a power of 2. If you have multiple Partition Keys, you only have to specify once.

Basic properties for PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Commas can be escaped using "" (for example '1,2,3'). Always specify DEFAULT as the value of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH customized subpartitions the database should create for a particular main RANGE partition (RANGE-HASH BY QUANTITY partitioning). For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once. Set it to 0 to reverse to the use of generic template HASH_QUANTITY specified in SUBPARTITION_KEY.

Basic properties for SUBPARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

For partition-level partitioning according to a Hash Algorithm or lists of values. Each partition is further sorted into subpartitions.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH subpartitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once.

Basic properties for SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST subpartition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any SUBPARTITION. Each LIST subpartition must have at least one value. No value, including NULL, can appear in more than one subpartition.

Basic properties for TEMPLATE_SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

In composite partitioning, template subpartitions are automatically applied to those partitions without their subpartitions specified. Here for LIST subpartitions only, specify a comma-delimited, ordered list of literal values corresponding to the LIST subpartitioning column. Always specify DEFAULT as the value of the last template LIST subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any TEMPLATE_SUBPARTITION. Each LIST template subpartition must have at least one value. No value, including NULL, can appear in more than one template subpartition.

Basic properties for INDEX:

Name: INDEX_TYPE

Type: STRING

Valid Values: UNIQUE, NON-UNIQUE, BITMAP, FUNCTION-BASED

Default: (No default, must be one of the preceding choices)

Specify the type of an index. NORMAL can be used in place of NON-UNIQUE.

Name: LOCAL_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify if an index is Global or Local. The default is Global.

Specify Local so that the index is partitioned on the same columns, with the same number of partitions and the same partition bounds as table.

Oracle Database automatically maintains local index partitioning as the underlying table is repartitioned.

Name: COLUMN_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

Specify an expression built from columns of table, constants, SQL functions, and user-defined functions to create a FUNCTION-BASED index.

Basic properties for INDEX_PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the index rows according to a Hash Algorithm, lists of values, or specified ranges. Hash index partitioning is supported starting with Oracle Database 10g version.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH index partitions the database should create based on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of index partitions that is a power of 2. If you have multiple index Partition Keys, you only have to specify once.

Basic properties for INDEX_PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition. No need to specify VALUES_LESS_THAN for Local index.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last partition. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition. No need to specify VALUES_EQUAL_TO for Local index.

Basic properties for CHECK_CONSTRAINTS:

Name: CHECK_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Specify a condition that each row in the table must satisfy.

Properties for MATERIALIZED_VIEW:

Name: BASE_TABLES

Type: STRING

Valid Values: N/A

Default: "

Specify a comma separated list of base tables for generating materialized view log.

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for the database object. The default is DEFAULT.

Name: BUILD

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE, PREBUILT

Default: "

Specify IMMEDIATE to populate the view when it is created. Specify DEFERRED to delays population until the next refresh operation. IMMEDIATE is the default.

Name: CONSTRAINTS

Type: STRING

Valid Values: , ENFORCED, TRUSTED

Default: "

Specify TRUSTED to let Oracle Database use dimension and constraint information that has been declared trustworthy by the database

administrator but that has not been validated by the database. If the dimension and constraint information is valid, then performance may improve. However, if this information is invalid, then the refresh procedure may corrupt the materialized view even though it returns a success status. ENFORCED is the default.

Name: DEFAULTINDEXBUFFERPOOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for the database object. The default is DEFAULT.

Name: DEFAULTINDEXFREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: DEFAULTINDEXFREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the database object. The default is 1.

Name: DEFAULTINDEXINITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: DEFAULTINDEXINITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2.

Name: DEFAULTINDEXMAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the database object.

Name: DEFAULTINDEXMAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: DEFAULTINDEXMINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: DEFAULTINDEXNEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: DEFAULTINDEXPCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the

preceding extent. The default is 50.

Name: DEFAULT_INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Specify tablespace for default index storage.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FOR_UPDATE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES (FOR UPDATE) to allow a subquery, primary key, object, or rowid materialized view to be updated. When used in conjunction with Advanced Replication, these updates will be propagated to the master. The default is NO.

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the database object. The default is 1.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: HASH_PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

A comma separated list of tablespaces to use for [sub]partition storage.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the database object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: NEXTDATE

Type: STRING

Valid Values: N/A

Default: "

Specify a datetime expression for calculating the interval between automatic refreshes.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Specify NOPARALLEL for serial execution. This is the default. Specify PARALLEL if you want Oracle to select a degree of parallelism equal to the number of CPUs available on all participating instances times the value of the PARALLEL_THREADS_PER_CPU initialization parameter.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Specify the number of parallel threads used in the parallel operation. Normally Oracle calculates the optimum degree of parallelism, so it is not necessary for you to specify it.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: QUERY_REWRITE

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE to mark the materialized view eligible for query rewrite or DISABLE to mark the materialized view ineligible for query rewrite. DISABLE is the default.

Name: REFRESH

Type: STRING

Valid Values: , COMPLETE, FAST, FORCE, NEVER

Default: "

Specify FAST to indicate the incremental refresh method. Specify COMPLETE to indicate the complete refresh method, which is implemented by executing the defining query of the materialized view. Specify FORCE to indicate that when a refresh occurs, Oracle Database will perform a fast refresh if one is possible or a complete refresh otherwise. FORCE is the default. Specify NEVER to prevent the materialized view from being refreshed with any Oracle Database refresh mechanism or packaged procedure.

Name: REFRESH_ON

Type: STRING

Valid Values: , COMMIT, DEMAND

Default: "

Specify COMMIT to indicate that a fast refresh is to occur whenever the database commits a transaction that operates on a master table of the materialized view. Specify DEMAND to indicate that the materialized view will be refreshed on demand by calling one of the three DBMS_MVIEW refresh procedures. DEMAND is the default.

Name: ROLLBACK

Type: STRING

Valid Values: , DEFAULT, DEFAULT LOCAL, DEFAULT MASTER, NONE

Default: DEFAULT LOCAL

Specify DEFAULT for Oracle Database to choose automatically which rollback segment to use. Specify DEFAULT MASTER for the remote rollback segment to be used at the remote master site for the individual materialized view.

Specify DEFAULT LOCAL for the remote rollback segment to be used for the local refresh group that contains the materialized view. DEFAULT LOCAL is the default. Specify NONE to name both master and local rollback segments.

Name: ROLLBACKSEGMENTLOCAL

Type: STRING

Valid Values: N/A

Default: "

Specify a named remote rollback segment to be used for the local refresh group that contains the materialized view. Default is null. Ignore if DEFAULT or DEFAULT LOCAL is specified for default rollback segment.

Name: ROLLBACKSEGMENTMASTER

Type: STRING

Valid Values: N/A

Default: "

Specify a named remote rollback segment to be used at the remote master site for the individual materialized view. Default is null. Ignore if DEFAULT or DEFAULT MASTER is specified for default rollback segment.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: STARTWITH

Type: STRING

Valid Values: N/A

Default: "

Specify a datetime expression for the first automatic refresh time.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: USING_INDEX_MODE

Type: STRING

Valid Values: , USING_INDEX, USING_NO_INDEX

Default: "

Specify USING_NO_INDEX to suppress the creation of the default index for Materialized View. You can create an alternative index for a Materialized View explicitly. The default is USING_INDEX.

Name: WITH

Type: STRING

Valid Values: , PRIMARY_KEY, ROWID

Default: "

Specify PRIMARY KEY to create a primary key materialized view. Specify ROWID to create a rowid materialized view. Rowid materialized views are useful if the materialized view does not include all primary key columns of the master tables. Rowid materialized views must be based on a single table and meet other restrictions. PRIMARY KEY is the default.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The

default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the

constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is

ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for CHECK_CONSTRAINT:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for INDEX:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: COMPUTESTATISTICS

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to collect statistics at relatively little cost during the creation of an index.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: INDEXORDER

Type: STRING

Valid Values: , ASC, DESC

Default: "

Use ASC or DESC to indicate whether the index should be created in ascending or descending order. The Oracle default is ASC.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2 for Index.

Name: KEYCOMPRESS

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Specify COMPRESS to enable key compression.

Name: KEYCOMPRESSPREFIXLENGTH

Type: STRING

Valid Values: N/A

Default: "

Specify the prefix length (number of prefix columns to compress). For unique indexes, the valid range of prefix length values is from 1 to the number of key columns minus 1. The default prefix length is the number of key columns minus 1. For nonunique indexes, the valid range of prefix length values is from 1 to the number of key columns. The default prefix length is the number of key columns. Oracle compresses only nonpartitioned indexes that are nonunique or unique indexes of at least two columns. You cannot specify COMPRESS for a bitmap index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (2-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: ONLINE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to indicate that DML operations on the table will be allowed during creation of the index.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of Index tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow Index tablespaces. The number of Index tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of Index tablespaces, then Oracle cycles through the names of the Index tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for a locally partitioned index. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. If specified, then individual local Hash index partitions specified will be ignored for Local Hash or Range-Hash Index partitioning.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: SORT

Type: STRING

Valid Values: , NOSORT, REVERSE, SORT

Default: "

Specify NOSORT to indicate to Oracle that the rows are already stored in the database in ascending order. Specify REVERSE to store the bytes of the index block in reverse order, excluding the rowid.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PARTITION:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1 for Table and 2 for Index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

addColumnClause

This clause adds a column.

When you create a table or alter a table to add a set of columns, the position that you specify for a column must be either less than or equal to the number of columns that you have listed up to that point in the command.

For example, the following OMBCREATE command does not add the specified columns to the table:

```
OMBCREATE TABLE 'MY_TABLE' \  
SET PROPERTIES (DESCRIPTION) VALUES ('TRIAL TABLE') \  
ADD COLUMN 'C1' \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \  
ADD COLUMN 'C2' AT POSITION 3 \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \  
ADD COLUMN 'C3' AT POSITION 2 \  
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2')
```

This is because at the point when you specify the position of the column C2 as 3, you have added just two columns to the table. But the following OMBALTER command adds the specified columns to the table. This is because at the point when you specify the position of the column C2 as 2, you are adding the second column to the table.

```
OMBCREATE TABLE 'MY_TABLE' \  
SET PROPERTIES (DESCRIPTION) VALUES ('TRIAL TABLE') \  
ADD COLUMN 'C1' \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \  
ADD COLUMN 'C2' AT POSITION 2 \  
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2') \  
ADD COLUMN 'C3' AT POSITION 1 \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10)
```

In the preceding example, the order in which the columns are added are as follows:

C1

C1, C2

C3, C1, C2

addViewConstraintClause

This clause adds the view's configuration clause.

addSCOClause

This clause will add SCOs.

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

propertyValue

This clause adds the property values.

addUkPkClause

This clause adds the adds unique key and primary keys.

QUOTED_STRING

name of the unique key or primary key.

addFkClause

This clause adds foreign key.

QUOTED_STRING

Name of the foreign key.

addIndexClause

This clause adds an index.

QUOTED_STRING

Name of the index.

addPartitionClause

This clause adds a partition.

QUOTED_STRING

Name of the partition.

addPartitionKeyClause

This clause adds a partition key.

QUOTED_STRING

Name of the partition key. This should be a column identifier.

addIndexColumnClause

This clause will add index column to a specified index.

QUOTED_STRING

This should be a column identifier of owning object (such as a table) of the index.

setUkPkPropertiesAndReferencesColumnsClauses

This clause adds properties and references to columns.

setFkSubClauses

This clause set references to a foreign key.

setSCOConfigurationPropertiesClauses

Set the configuration properties for the following:

- Partition, Subpartition, and Template Subpartition: All refer to configuration properties of Partition.
- Index, and Index Partition: For Index Partition, refer to configuration properties of Partition.

renameSCOConfigurationClause

This clause renames configuration objects.

constraintColumnReferencesClause

This clause provides names of all columns.

setFkReferencesClauses

This clause sets foreign key references.

quotedNameList

This clause gives column names.

constraintUkReferencesClause

The first QUOTED_STRING denotes the UniqueKey or Primary key name, and the latter denotes the table's or view's name.

Examples

```
OMBCREATE MATERIALIZED_VIEW 'NEW_MATERIALIZED_VIEW' SET
PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('this is a new materialized view',
'New MaterializedView')
```

This will create a materialized view named "NEW_VIEW", its description is "this is a new materialized view", and business name is "New MaterializedView".

See Also

OMBCREATE, OMBALTER MATERIALIZED_VIEW, OMBDROP MATERIALIZED_VIEW, OMBRETRIEVE MATERIALIZED_VIEW

OMBCREATE MDL_ACTION_PLAN

Purpose

Create a metadata loader action plan.

Prerequisites

Connection must be established to the repository.

Syntax

```
createMDLActionPlanCommand = ( OMBCREATE TRANSIENT MDL_ACTION_PLAN
    "QUOTED_STRING" { "addActionClause" } )
addActionClause = ADD ACTION "QUOTED_STRING" "setReferenceClause"
setReferenceClause = SET ( REF | REFERENCE ) ( "referenceValueClause" )
referenceValueClause = "objectTypeValue" "QUOTED_STRING" [ SET ( REF |
    REFERENCE ) "referenceValueClause" ]
objectTypeValue = ( PROJECT | ORACLE_MODULE | TABLE | VIEW | SEQUENCE |
    MATERIALIZED_VIEW | FUNCTION | PROCEDURE | PACKAGE | DIMENSION | CUBE
    | ADVANCED_QUEUE | STREAMS_QUEUE | MAPPING | REAL_TIME_MAPPING |
    PROCESS_FLOW_MODULE | PROCESS_FLOW_PACKAGE | PROCESS_FLOW | SAP_MODULE
    | CMI_MODULE | GATEWAY_MODULE | EXTERNAL_TABLE | FLAT_FILE_MODULE |
    FLAT_FILE | BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE
    | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL |
    ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
    PRESENTATION_TEMPLATE | LOCATION | CONNECTOR | CONTROL_CENTER |
    CONFIGURATION | COLLECTION | SNAPSHOT | ROLE | USER | ICONSET |
    TRANSFORMATION_MODULE | CALENDAR_MODULE | CALENDAR_FOLDER | CALENDAR |
    EXPERT_MODULE | EXPERT | DATA_RULE_MODULE | DATA_RULE | DATA_AUDITOR
    | STREAMS_CAPTURE_PROCESS | QUEUE_TABLE | QUEUE_PROPAGATION |
    OBJECT_TYPE | NESTED_TABLE | VARYING_ARRAY | DEPLOYMENT | DATA_PROFILE
    | PROFILE_REFERENCE | PLSQL_TABLE_TYPE | PLSQL_RECORD_TYPE |
    PLSQL_REF_CURSOR_TYPE | PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER |
    CMI_DEFINITION | ACTIVITY_TEMPLATE | ACTIVITY_TEMPLATE_FOLDER |
    TRANSPORTABLE_MODULE )
```

Keywords And Parameters

createMDLActionPlanCommand

Create a metadata loader action plan.

QUOTED_STRING

Name of the action plan in a single-quoted string. It is case-insensitive.

addActionClause

Add an action to an action plan.

QUOTED_STRING

Name of the action in a single-quoted string. It is case-insensitive.

Must be unique within an action plan.

setReferenceClause

Specify the object type and the absolute path name of an object.

referenceValueClause

Specify a first-class object type and the absolute path name of an object.

QUOTED_STRING

Absolute path name of an object (for example '/MY_PROJECT/MODULE_X/TABLE_Y').

objectTypeValue

The first-class object type that is allowed to be specified in the referenceValueClause.

Examples

```
OMBCREATE TRANSIENT MDL_ACTION_PLAN 'MY_ACTION_PLAN'
```

```
OMBCREATE TRANSIENT MDL_ACTION_PLAN 'MY_PROJECT_ACTION_PLAN'
ADD ACTION 'MY_PROJECT'
SET REFERENCE PROJECT '/MY_PROJECT'
```

```
OMBCREATE TRANSIENT MDL_ACTION_PLAN 'GRANULAR_OBJECTS_
ACTION_PLAN'
ADD ACTION 'GRANULAR_OBJJS'
SET REFERENCE TABLE '/MY_PROJECT/DW/TABLE_1'
SET REFERENCE VIEW '/MY_PROJECT/DW/*.*'
SET REFERENCE DIMENSION '/MY_PROJECT/DW/D.*'
SET REFERENCE MAPPING '/MY_PROJECT/DW/[M,P,S].*'
```

```
OMBCREATE TRANSIENT MDL_ACTION_PLAN 'MULTI_PROJECT_ACTION_
PLAN'
ADD ACTION 'MULTI_PROJ'
SET REFERENCE PROJECT '/PUBLIC_PROJECT'
SET REFERENCE PROJECT '/MY_PROJECT'
SET REFERENCE PROJECT '/ABC_PROJECT'
```

```
OMBCREATE TRANSIENT MDL_ACTION_PLAN 'ADMIN_ACTION_PLAN'
ADD ACTION 'ADMIN'
```

```
SET REFERENCE USER '.*'  
SET REFERENCE ROLE 'S.*'  
SET REFERENCE ICONSET 'ICONSET_1'  
SET REFERENCE CMI_DEFINITION 'CMI_DEF1'  
  
OMBCREATE TRANSIENT MDL_ACTION_PLAN 'SNAPSHOT_ACTION_PLAN'  
ADD ACTION 'SNAPSHOTS'  
SET REFERENCE SNAPSHOT 'SNAPSHOT_MY_PROJECT'
```

See Also

OMBALTER MDL_ACTION_PLAN, OMBDROP MDL_ACTION_PLAN,
OMBRETRIEVE MDL_ACTION_PLAN, OMUEXPORT MDL_FILE

OMBCREATE MINING_MODEL

Purpose

Create a data mining model in an Oracle Module

Prerequisites

Should be in Oracle Module context

Syntax

```
createMiningModelCommand = OMBCREATE MINING_MODEL "miningModelName" [
    "miningFunctionAndAlgorithmClause" ]
    "createMiningMapOperatorOwnerDetailClause"
miningModelName = "QUOTED_STRING"
miningFunctionAndAlgorithmClause = FOR MINING_FUNCTION "QUOTED_STRING" [
    USING MINING_ALGORITHM "QUOTED_STRING" ]
createMiningMapOperatorOwnerDetailClause = [
    "setMiningModelAndMapPropertiesClause" ]
    "createOperatorOwnerDescendantsClause"
setMiningModelAndMapPropertiesClause = ( SET "setPropertiesClause" |
    "setMiningMapPropertiesClause" )
createOperatorOwnerDescendantsClause = { ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) }
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setMiningMapPropertiesClause = SET MINING_BUILD_MAP PROPERTIES
    "propertyKeyList" VALUES "propertyValueList"
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyValueList = "(" "propertyValue" { "," "propertyValue" } ")"
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
attributeName = "QUOTED_STRING"
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
childType = "UNQUOTED_STRING"
```

```

childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { ", " IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
pluggableMapName = "QUOTED_STRING"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
attributeNameList = "( "attributeName" { ", "attributeName" } )"
bindableType = PLUGGABLE_MAPPING | MINING_MODEL | OBJECT_TYPE |
    "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

createMiningModelCommand

Create a data mining model in an Oracle Module, and specify the mapping that builds the mining model

createOperatorOwnerDescendantsClause

Create the desired child objects of a mapping or a pluggable mapping.

setPropertyClause

Describe the keys of properties for the map or objects in the map.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating a child object under a mapping (which is not an operator)

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

In the second example, when user forgets to type "OPERATOR" "GROUP" "ATTRIBUTE" key word, instead of complaining the keywords are missing, OMBPlus will complain about error getting child objects. Here is an example:

```
OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'
```

```
OMB02932: Error getting child objects of type TABLE in M1
```

TO A USER: it looks like OMBPlus should complain they forget to type a keyword.

TO OMBPLUS: the syntax is actually for creating a non-operator child object under the mapping. Therefore, it goes and tries to find type definition for non-operator child object "TABLE" and cannot find it. Therefore the exception is thrown.

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

addChildClause

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

addConnectionClause

Add connections between mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CUBE, DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPRESSION, EXTERNAL_PROCESS, EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, JOINER, KEY_LOOKUP, LCRCast, LCRSPplitter, MATCHMERGE, MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, PIVOT, POSTMAPPING_PROCESS, PREMAPPING_PROCESS, SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

attributeName

Name of a mapping attribute.

groupBottomUpLocator

Location of a mapping group.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributeBottomUpLocator

Location of a mapping attribute.

attributesBottomUpLocator

Location of a list of mapping attributes.

propertyKey

A property key for an object.

Basic properties for Mining Model:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mining model

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mining model

Name: ALGO_NAME

Type: STRING(200)

Valid Values: NAIVE_BAYES, ADAPTIVE_BAYES_NETWORK, SUPPORT_VECTOR_MACHINES,

KMEANS, PREDICTOR_VARIANCE, NONNEGATIVE_MATRIX_FACTORIZATION

Default: 'Each mining function has a corresponding default'

Description: Name of the algorithm for the mining function

Properties for CLASSIFICATION mining function

Name: PRIORS_TABLE_NAME

Type: String

The name of the prior probabilities table

Settings for Naive Bayes algorithm:

Name: SINGLETON_THRESHOLD

Type: float

Valid Values: 0 .. 1

Default: '0.01'

Singleton Threshold for Naive Bayes Alogrithm

Name: PAIRWISE_THRESHOLD

Type: float

Valid Values: 0 .. 1

Default: '0.01'

Pairwise Threshold for Naive Bayes Alogrithm

Settings for the Adaptive Naye Bayes algorithm

Name: MODEL_TYPE

Type: String

Default: 'MULTI_FEATURE'

The model type for the Adaptive Bayes Network Algorithm

Name: MAX_NB_PREDICTORS

Type: Integer

Valid Values: 1 ..

Default: '10'

Maximum Naive Bayes Predictors

Name: MAX_PREDICTORS

Type: Integer

Valid Values: 1 ..

Default: '25'

Maximum Predictors

Name: MAX_BUILD_MINUTES

Type: Integer

Valid Values: 0 ..

Default: '0'

Maximum time (in minutes) allowed to build this model

Settings for the Support Vector Machines Algorithm for Adaptive Naive Bayes

Name: KERNEL_FUNCTION

Type: String

Valid Values: LINEAR,GAUSSIAN

Default: 'LINEAR'

Kernel function

Name: KERNEL_CACHE

Type: INTEGER

Valid Values: 1 ..

Default: '500000'

Value of the kernel cache for the SVM algorithm

Name: EPSILON

Type: float

Default: '0.1'

Value of the kernel cache for the SVM algorithm

Name: CONVTOLERANCE

Type: float

Valid Values: 0 ..

Default: '0.001'

Value of the convergence tolerance

Name: STDDEV

Type: float

Valid Values: 0 ..

Standard deviation for the SVM algorithm

Name: CFACTOR

Type: float

Complexity factor for the SVM algorithm

Name: TARGETTYPE

Type: String

Valid Values: SVMS_SINGLE_TARGET,SVMS_MULTI_TARGET

Default: 'SVMS_SINGLE_TARGET'

Target Type for the SVM algorithm

Properties for CLUSTERING function

Name: NUM_CLUSTERS

Type: Integer

Valid Values: 1 ..

Default: '10'

Properties for MINING_MODEL:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: SETTINGS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of table which stores the settings for model build.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

pluggableMapName

Name of the pluggable map.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBCREATE MINING_MODEL 'MODEL1'
```

```
OMBCREATE MINING_MODEL 'MODEL1'  
FOR MINING_FUNCTION 'CLASSIFICATION'  
USING MINING_ALGORITHM 'algo_adaptive_bayes_nextwork'
```

See Also

OMBCREATE, OMBALTER MINING_MODEL, OMBRETRIEVE MINING_MODEL, OMBDROP MINING_MODEL

OMBCREATE NESTED_TABLE

Purpose

To create an Nested Table (or NestedTable)

Prerequisites

Should be in the context of an Oracle Module

Syntax

```
createNestedTableCommand = OMBCREATE ( NESTED_TABLE "QUOTED_STRING" [ SET
    ( "setPropertiesClause" [ SET ( REF | REFERENCE )
      "setReferenceIconSetClause" ] | ( REF | REFERENCE )
      "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createNestedTableCommand

Creates a Nested Table with the given name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Nested Table .

Valid properties are as shown:

Basic properties for NESTED_TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Nested Table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Nested Table

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Base Element of the Nested Table

Properties for NESTED_TABLE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE NESTED_TABLE 'NEW_NESTED_TABLE' SET PROPERTIES  
(DATATYPE) VALUES
```

```
('NUMBER')
```

This will create a NestedTable named 'NEW_NESTED_TABLE' with its base element type as 'NUMBER' .

See Also

OMBCREATE, OMBALTER NESTED_TABLE, OMBDROP NESTED_TABLE

OMBCREATE OBJECT_TYPE

Purpose

To create an Object Type.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
createObjectTypeCommand = OMBCREATE ( OBJECT_TYPE "QUOTED_STRING" [ SET (
    "setPropertyClause" [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] [ "addObjectAttributesClause" ] )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
addObjectAttributesClause = "addAttributeClause"+
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addAttributeClause = ADD OBJECT_TYPE_ATTRIBUTE "QUOTED_STRING" [ AT
    POSITION "INTEGER_LITERAL" ] [ SET "setPropertyClause" ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createObjectTypeCommand

Creates an Object Type with the given name.

setPropertyClause

Sets properties (core, logical, physical, user-defined) for Object Type or its Attributes. Valid properties are as shown:

Basic properties for OBJECT_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Object Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Object Type

Basic properties for OBJECT_TYPE_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Properties for OBJECT_TYPE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for

those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

`addObjectAttributesClause`

Adds one or more Attributes in this Object Type.

`propertyNameList`

The list of properties.

`propertyValueList`

The list of property values.

`addAttributeClause`

Adds an Attribute with the given name and properties.

`propertyValue`

This clause adds the property values.

Examples

```
OMBCREATE OBJECT_TYPE 'NEW_OBJECT_TYPE' SET PROPERTIES
(DESCRIPTION) VALUES
```

```
('this is an object type') ADD OBJECT_TYPE_ATTRIBUTE 'ATTR' SET PROPERTIES
(DATATYPE) VALUES ('VARCHAR2')
```

This will create an Object Type named "NEW_OBJECT_TYPE", its description is "this is an object type" and an Attribute 'ATTR' of Varchar2 type.

See Also

OMBCREATE, OMBALTER OBJECT_TYPE, OMBDROP OBJECT_TYPE

OMBCREATE ORACLE_MODULE

Purpose

To create an Oracle module.

Prerequisites

Should be in the context of project.

Syntax

```
createOracleModuleCommand = OMBCREATE ( ORACLE_MODULE "QUOTED_STRING" [
    SET ( "setPropertiesClause" [ SET
        "setReferenceClauseForDataMetadataModule" ] |
        "setReferenceClauseForDataMetadataModule" ) ] [
        "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
```

Keywords And Parameters

createOracleModuleCommand

This command creates an Oracle module

QUOTED_STRING

Name of the Oracle module to be created.

setPropertiesClause

Associate a set of properties with an Oracle module.

Basic properties for ORACLE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)
Valid Values: N/A
Default: NAME
Business name of an Oracle Module

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of an Oracle Module

Name: MODULE_TYPE
Type: STRING
Valid Values: N/A
Default: N/A
Type of oracle module. Supported values are: 'WAREHOUSE_TARGET',
'DATA_SOURCE'.By default, it is 'WAREHOUSE_TARGET'.

Properties for ORACLE_MODULE:

Name: ABAP_DIRECTORY
Type: STRING
Valid Values: N/A
Default: abap\
Location where SAP data is dumped as flat files

Name: ABAP_EXTENSION
Type: STRING
Valid Values: N/A
Default: .abap
File name extension for ABAP scripts

Name: ABAP_RUN_PARAMETER_FILE
Type: STRING
Valid Values: N/A
Default: _run.ini
Run Parameter File Suffix for the parameter script in a ABAP job.

Name: ABAP_SPOOL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: abap\log\

Location where ABAP scripts are buffered during script generation processing.

Name: APPLICATION_SHORT_NAME

Type: STRING

Valid Values: N/A

Default: WB

Application Short Name

Name: ARCHIVE_DIRECTORY

Type: STRING

Valid Values: N/A

Default: archive\

Archive Directory

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

If this is a source module, this value indicates the location from which data will be read. If this is a target warehouse module, this value indicates the location where generated code will be deployed to and/or where data will be written to.

Name: DDL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ddl\

Location where scripts for database objects for the target schema are stored.

Name: DDL_EXTENSION

Type: STRING
Valid Values: N/A
Default: .ddl
File name extension for DDL scripts.

Name: DDL_SPOOL_DIRECTORY
Type: STRING
Valid Values: N/A
Default: ddl\log\
Location where DDL scripts are buffered during script generation processing.

Name: DEFAULT_INDEX_TABLESPACE
Type: STRING(30)
Valid Values: N/A
Default: "
Default name of tablespace to install indexes into.

Name: DEFAULT_OBJECT_TABLESPACE
Type: STRING(30)
Valid Values: N/A
Default: "
Default name of tablespace to install objects into.

Name: DEPLOYABLE
Type: BOOLEAN
Valid Values: true, false
Default: true
Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: END_OF_LINE
Type: STRING
Valid Values: N/A
Default: \r\n
End of Line

Name: INPUT_DIRECTORY

Type: STRING

Valid Values: N/A

Default: input\

Input Directory

Name: INVALID_DIRECTORY

Type: STRING

Valid Values: N/A

Default: invalid\

Directory for SQL*Loader errors and rejected records

Name: LIB_DIRECTORY

Type: STRING

Valid Values: N/A

Default: lib\

LIB Directory

Name: LIB_EXTENSION

Type: STRING

Valid Values: N/A

Default: .lib

LIB Extension

Name: LIB_SPOOL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: lib\log\

LIB Spool Directory

Name: LOADER_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ctl\

Location where control files are stored.

Name: LOADER_EXTENSION

Type: STRING
Valid Values: N/A
Default: .ctl
Suffix for the loader scripts

Name: LOADER_RUN_PARAMETER_FILE
Type: STRING
Valid Values: N/A
Default: _run.ini
Suffix for the parameter initialization file.

Name: LOG_DIRECTORY
Type: STRING
Valid Values: N/A
Default: log\
Log Directory for the SQL*Loader

Name: MAIN_APPLICATION_SHORT_NAME
Type: STRING
Valid Values: N/A
Default: ora
Main Application Short Name

Name: PLSQL_DIRECTORY
Type: STRING
Valid Values: N/A
Default: pls\
Location where PL/SQL scripts are stored.

Name: PLSQL_EXTENSION
Type: STRING
Valid Values: N/A
Default: .pls
File name extension for PL/SQL scripts.

Name: PLSQL_GENERATION_MODE
Type: STRING

Valid Values: Default, Oracle Database 10g, Oracle Database 10gR2, Oracle Database 8i, Oracle Database 9i

Default: Default

Generation mode controls validation and generation for version specific features.

Name: PLSQL_RUN_PARAMETER_FILE

Type: STRING

Valid Values: N/A

Default: _run.ini

Suffix for the parameter script in a PL/SQL job.

Name: PLSQL_SPOOL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: pls\log\

Location where PL/SQL scripts are buffered during script generation processing.

Name: RECEIVE_DIRECTORY

Type: STRING

Valid Values: N/A

Default: receive\

Receive Directory

Name: SORT_DIRECTORY

Type: STRING

Valid Values: N/A

Default: sort\

Sort Directory

Name: STREAMS_ADMINISTRATOR_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location corresponding to the Streams Administrator

Name: TCL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: tcl\

Location for TCL scripts that are generated after registration with Oracle Enterprise Manager

Name: TOP_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ..\..\codegen\

Top Directory where generated code will get stored

Name: WORK_DIRECTORY

Type: STRING

Valid Values: N/A

Default: work\

Work Directory

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the Oracle module.

addModuleReferenceLocationClause

Add runtime locations to the Oracle module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a runtime location to the Oracle module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the Oracle module.

addReferenceLocationClause

Add a runtime location to the Oracle module.

propertyValue

Value of a property.

setReferenceMetadataLocationClause

Set metadata location for the Oracle module.

setReferenceIconSetClause

Set icon set for the Oracle module.

Examples

```
OMBCREATE ORACLE_MODULE 'src_module' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('this is an Oracle module', 'source module')
```

This will create an Oracle module named "src_module", its description is "this is an Oracle module", and business name is "source module".

See Also

OMBCREATE, OMBALTER ORACLE_MODULE, OMBDROP ORACLE_MODULE

OMBCREATE PACKAGE

Purpose

To create a Package.

Prerequisites

Should be in the context of a Oracle Module or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
createPackageCommand = OMBCREATE ( PACKAGE "QUOTED_STRING" ( [ SET (
    "setPropertyClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] ) { ADD "addRelationalDependentClause"
    } )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createPackageCommand

This command creates a Package

QUOTED_STRING

Name of the Package to be created.

setPropertyClause

Used to set properties (core, user-defined) for packages. Valid properties are as shown:

Basic properties for PACKAGE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Package

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Package

Properties for PACKAGE:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the package with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addRelationalDependentClause

This clause adds referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE PACKAGE 'pkg' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME) VALUES
```

```
('this is a Package', 'package')
```

This will create a Package named "pkg", its description is "this is a Package", and business name is "package".

See Also

OMBCREATE, OMBALTER PACKAGE, OMBDROP PACKAGE

OMBCREATE PLSQL_RECORD_TYPE

Purpose

To create a PLSQL Record Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
createPlSqlRecordTypeCommand = OMBCREATE ( PLSQL_RECORD_TYPE
    "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET ( REF | REFERENCE
    ) "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] [ "addPlSqlRecordAttributesClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
addPlSqlRecordAttributesClause = "addRecordTypeAttributeClause"+
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addRecordTypeAttributeClause = ADD ATTRIBUTE "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createPlSqlRecordTypeCommand

Creates an PLSQL Record Type with the given name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for PLSQL Record Type or its Attributes. Valid properties are as shown:

Basic properties for PLSQL_RECORD_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the PLSQL Record Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the PLSQL Record Type

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Properties for PLSQL_RECORD_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE,

NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME,
NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,
 NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,
 NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,
 NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,
 NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE
 Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)
 Valid Values: NO, YES
 Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)
 Valid Values: NO, YES
 Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING
 Valid Values: N/A
 Default: "
 Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING
 Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE, NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART, NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV, NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE, NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE, NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10, NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15, NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,

NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,
 NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,
 NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,
 NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,
 NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
 NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
 NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,
 NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA, NA_NAMEDESIGNATOR,
 NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE, NA_STREETCOMPCORRECTED,
 NA_STREETCOMPMATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR, NA_UNITNUMBER,
 NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4, NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, || =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addPISqlRecordAttributesClause

Adds one or more Attributes in this PLSQL Record Type.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

addRecordTypeAttributeClause

Adds an attribute to this PLSQL Record Type

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE PLSQL_RECORD_TYPE 'NEW_PLSQL_RECORD_TYPE' SET
PROPERTIES
(DESCRIPTION) VALUES ('this is a plsql record type') ADD ATTRIBUTE 'ATTR'
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2')
```

This will create a PLSQL Record Type named "NEW_PLSQL_RECORD_TYPE", its description is "this is a plsql record type" and an Attribute 'ATTR' of Varchar2 type.

See Also

```
OMBCREATE, OMBALTER PLSQL_RECORD_TYPE, OMBDROP PLSQL_RECORD_
TYPE
```

OMBCREATE PLSQL_REF_CURSOR_TYPE

Purpose

To create a PLSQL Ref-Cursor Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
createPLSqlRefCursorTypeCommand = OMBCREATE ( PLSQL_REF_CURSOR_TYPE
    "QUOTED_STRING" SET "setPropertiesClause" )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createPLSqlRefCursorTypeCommand

Creates a PL/SQL Ref-Cursor Type with the given name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Ref-Cursor Type

Valid properties are as shown:

Basic properties for PLSQL_REF_CURSOR_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Ref-Cursor Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Ref-Cursor Type

Name: RETURN_TYPE

Type: STRING(20)

Valid Values: N/A

Default: "

Return type of the Ref-Cursor Type. This should be a PLSQL Record Type.

Properties for PLSQL_REF_CURSOR_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for PLSQL_REF_CURSOR_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE PLSQL_REF_CURSOR_TYPE 'NEW_REF_CURSOR_TYPE' SET  
PROPERTIES  
(DESCRIPTION) VALUES ('this is a ref-cursor type')
```

See Also

```
OMBCREATE, OMBALTER PLSQL_REF_CURSOR_TYPE, OMBDROP PLSQL_REF_  
CURSOR_TYPE
```

OMBCREATE PLSQL_TABLE_TYPE

Purpose

To create a PLSQL Table Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
createPLSqlTableTypeCommand = OMBCREATE ( PLSQL_TABLE_TYPE "QUOTED_STRING"  
    SET "setPropertiesClause" )  
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("  
    "propertyValueList" ")"  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }  
propertyValueList = "propertyValue" { "," "propertyValue" }  
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |  
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createPLSqlTableTypeCommand

Creates a PL/SQL Table Type with the given name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Table Type

Valid properties are as shown:

Basic properties for PLSQL_TABLE_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Table Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Table Type

Name: RETURN_TYPE

Type: STRING(20)

Valid Values: NUMBER, VARCHAR2, VARCHAR, DATE, FLOAT

Default: "

Return type of the Table Type. This can be a scalar type or a PLSQL Record Type.

Properties for PLSQL_TABLE_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for PLSQL_TABLE_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE PLSQL_TABLE_TYPE 'NEW_TABLE_TYPE' SET PROPERTIES  
(DESCRIPTION)  
VALUES ('this is a table type')
```

See Also

OMBCREATE, OMBALTER PLSQL_TABLE_TYPE, OMBDROP PLSQL_TABLE_TYPE

10

OMBCREATE PLUGGABLE_MAPPING to OMBCREATE VIEW

This chapter lists commands associated with OMBCREATE in alphabetical order starting with OMBCREATE PLUGGABLE_MAPPING.

OMBCREATE PLUGGABLE_MAPPING

Purpose

Create a pluggable mapping in a project or a pluggable map folder.

Prerequisites

1. The current context of scripting must be a project or pluggable map folder.
2. No concurrent user should be locking the project or pluggable map folder or any of its ancestors exclusively at the moment the map is being created.
3. The pluggable mapping name must not conflict with existing pluggable mapping names and the pluggable mappings names that concurrent user tries to use.

Syntax

```
createPluggableMappingCommand = OMBCREATE PLUGGABLE_MAPPING
    "pluggableMapName" "createOperatorOwnerDetailClause"
pluggableMapName = "QUOTED_STRING"
createOperatorOwnerDetailClause = [ SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] ) | "setReferenceIconSetClause" ) ]
    "createOperatorOwnerDescendantsClause"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
createOperatorOwnerDescendantsClause = { ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) }
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
```

```

groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
attributeName = "QUOTED_STRING"
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { "," IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
attributeNameList = "(" "attributeName" { "," "attributeName" } ")"
bindableType = PLUGGABLE_MAPPING | OBJECT_TYPE | "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

createPluggableMappingCommand

Create a pluggable mapping in a project or a pluggable map folder.

pluggableMapName

Name of the pluggable map.

createOperatorOwnerDetailClause

Create the desired detail of a pluggable mapping.

setPropertyClause

Describe the keys of properties for the map or objects in the map.

createOperatorOwnerDescendantsClause

Create the desired child objects of a mapping or a pluggable mapping.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

addGroupClause

Add a mapping group to a mapping operator.

addAttributeClause

Add a mapping attribute to a mapping group.

addChildClause

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

The following is an example for creating a child object under a mapping

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

Note: Key word "OPERATOR" "GROUP" "ATTRIBUTE" are important for their respective ADD clauses. Without the key words, OMBPlus will interpret the ADD clause as an addChildClause. Here is an example:

```
OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'
```

```
OMB02932: Error getting child objects of type TABLE in M1
```

OMBPLUS interprets the ADD clause as one for creating a non-operator child object under the mapping object. Therefore, it tries to find type definition for non-operator child object "TABLE" and cannot find it.

addConnectionClause

Add connections between mapping groups or mapping attributes.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A
Default: 0
Length of the attribute.

Name: PRECISION
Type: NUMBER
Valid Values: N/A
Default: 0
Precision of the attribute.

Name: SCALE
Type: NUMBER
Valid Values: N/A
Default: 0
Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION
Type: NUMBER
Valid Values: 0 - 9
Default: 0
The precision of a timestamp or interval.

Properties for PLUGGABLE_MAPPING:

Name: TARGET_LOAD_ORDER
Type: STRING(65535)
Valid Values: N/A
Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract

Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier.

Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,

NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this

attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,
 NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_
 ABBREV,
 NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_
 COUNTRYCODE,
 NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_
 DELIVERYBEATCODE,
 NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1,
 NA_EXTRA_10,
 NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,
 NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
 NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_
 EXTRA_7,
 NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_
 CODE,
 NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_
 FIRSTNAMESTD,
 NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_
 INSTALLATIONTYPE,
 NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_
 ISGOODGROUP,
 NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
 NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME,
 NA_LOCALITY_2,
 NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_
 ORDER, NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_
 MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
 NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_
 NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_
 DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_
 PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
 NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
 POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
 PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
 ROUTENAME,

NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
 STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
 STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE of SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, -=, =||, || =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UUID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE
Type: STRING
Valid Values: N/A
Default: "
A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME
Type: STRING
Valid Values: N/A
Default: "
The name to be used by the code generator to identify this item. By default

it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORD_CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TARGET_FILTER_FOR_DELETE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER
Type: STRING(65535)
Valid Values: N/A
Default: "
The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only

used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCast, LCRSPplitter, MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION, TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

attributeName

Name of a mapping attribute.

groupBottomUpLocator

Location of a mapping group.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributeBottomUpLocator

Location of a mapping attribute.

attributesBottomUpLocator

Location of a list of mapping attributes.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBCREATE PLUGGABLE_MAP 'PLUGGABLE_MAP1'
```

```
OMBCREATE PLUGGABLE_MAP 'PLUGGABLE_MAP1'
```

```
ADD TABLE OPERATOR 'CUST_SRC'
```

```
BOUND TO TABLE '../SRC_MODULE/CUST_SRC'
```

See Also

OMBCREATE, OMBALTER PLUGGABLE_MAPPING, OMBRETRIEVE PLUGGABLE_MAPPING, OMBDROP PLUGGABLE_MAPPING

OMBCREATE PLUGGABLE_MAPPING_FOLDER

Purpose

Create a pluggable map folder in a project.

Prerequisites

1. The current context of scripting must be a project.
2. No concurrent user should be locking the project or any of its ancestors exclusively at the moment the map is being created.
3. The pluggable map folder name must not conflict with existing pluggable map folder names and the pluggable map folder names that concurrent user tries to use.

Syntax

```
createPluggableMappingFolderCommand = ( OMBCREATE PLUGGABLE_MAPPING_FOLDER
    "pluggableMapFolderName" [ SET ( ( "setPropertiesClause" [ SET
        "setReferenceIconSetClause" ] ) | "setReferenceIconSetClause" ) ] )
pluggableMapFolderName = "QUOTED_STRING"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createPluggableMappingFolderCommand

Create a pluggable map folder in a project.

pluggableMapFolderName

Name of the pluggable map folder.

setPropertiesClause

Describe the keys of properties for the map or objects in the map.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

Examples

```
OMBCREATE PLUGGABLE_MAP_FOLDER 'PLUGGABLE_MAP_FOLDER1'
```

See Also

OMBCREATE

OMBCREATE PRESENTATION_TEMPLATE

Purpose

Creates a presentation template that can be used in a business presentation module.

Prerequisites

Should be in the context of a business presentation module.

Syntax

```
createReportCommand = ( OMBCREATE PRESENTATION_TEMPLATE "QUOTED_STRING" [
    SET "setpropertiesClauseDelayed" ] [ SET "setReferenceIconSetClause" ]
    { "addReportItemClauses" } )
setpropertiesClauseDelayed = PROPERTIES "(" "propertyNameListVector" ")"
    VALUES "(" "propertyValueListVector" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addReportItemClauses = ( ADD DATA_ITEM "QUOTED_STRING" [ SET
    "setPropertiesClause" ] [ SET ( REF | REFERENCE )
    "ReportMeasureReferencesClause" ] ) | ( ADD EDGE_ITEM "QUOTED_STRING"
    [ SET "setPropertiesClause" ] [ SET ( REF | REFERENCE )
    "ReportEdgeReferencesClause" ] )
propertyNameListVector = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueListVector = "propertyValue" { ", " "propertyValue" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
ReportMeasureReferencesClause = MEASURE "QUOTED_STRING" OF CUBE
    "QUOTED_STRING"
ReportEdgeReferencesClause = [ ROLE "QUOTED_STRING" OF ] DIMENSION
    "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
```

Keywords And Parameters

`createReportCommand`

This command creates a presentation template.

`QUOTED_STRING`

Specify the name of the presentation template to be created.

`setpropertiesClauseDelayed`

This clause sets the properties.

`setReferenceIconSetClause`

Set specified Icon Set.

`addReportItemClauses`

This adds items to a presentation template.

propertyNameListVector

This clause holds the names of the properties.

propertyValueListVector

This clause holds the values of the properties.

setPropertyClause

Used to set properties (core, logical, physical, user-defined) for presentation templates. Valid properties are as shown:

Basic properties for presentation template:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the presentation template

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the presentation template

Name: PRESENTATION_TYPE

Type: STRING(40)

Valid Values: CROSSTAB, TABLE or a subtype of graph

Default: "

The type of the presentation template

Basic properties for EDGE_ITEM:

Name: PLACEMENT

Type: STRING(40)

Valid Values: TOP OR SIDE, TOP, SIDE, PAGE

Default: "

The placement of the edge item in the presentation template

Properties for PRESENTATION_TEMPLATE:

Name: CATALOG_FOLDER

Type: STRING

Valid Values: N/A

Default: "

Catalog Folder for deployed BI Beans presentation

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for referenced database objects

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

ReportMeasureReferencesClause

This clause references a measure from the item.

ReportEdgeReferencesClause

This clause references dimension or roles from the item.

propertyValue

This is a property value.

propertyNameList

This is the list of property names.

propertyValueList

This is the list of property values.

Examples

```
OMBCREATE PRESENTATION_TEMPLATE 'SALES_REPORT'
```

See Also

```
OMBALTER PRESENTATION_TEMPLATE, OMBRETRIEVE PRESENTATION_
TEMPLATE
```

OMBCREATE PROCEDURE

Purpose

To create a Procedure.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
createProcedureCommand = OMBCREATE ( PROCEDURE "QUOTED_STRING" ( [ SET (
    "setPropertyClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] ) { ADD ( "addFuncProcParameterClause"
    | "addRelationalDependentClause" ) } )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addFuncProcParameterClause = PARAMETER "QUOTED_STRING" [ SET
    "setPropertyClause" ]
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW | SEQUENCE | FUNCTION | PROCEDURE | PACKAGE )
    "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createProcedureCommand

This command creates a Procedure

QUOTED_STRING

Name of the Procedure to be created.

setPropertyClause

Used to set properties (core, user-defined) for procedurefunction. Valid properties are as shown:

Basic properties for PROCEDURE:

Name: BUSINESS_NAME

Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the Procedure

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the Procedure

Name: IMPLEMENTATION
Type: STRING
Valid Values: N/A
Default: "
Set the code for Procedure which is included global variable declaration
and code between BEGIN and END.

Basic properties for PARAMETER:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the Parameter

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the Parameter

Name: DATATYPE
Type: STRING
Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB,
BOOLEAN,
CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL
YEAR TO

MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR,

VARCHAR2, XMLTYPE

Default: NUMBER

Set the data type for Parameter

Name: IN_OUT

Type: STRING

Valid Values: IN, OUT, INOUT

Default: 'IN'

Set the parameter mode for Parameter

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Set the default value for Parameter

Properties for PROCEDURE:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

`addFuncProcParameterClause`

Adds one or more Parameters to this Procedure.

`addRelationalDependentClause`

This clause adds referential dependencies to other relational objects.

`propertyNameList`

Comma separated list of property names. Property names are unquoted.

`propertyValueList`

Comma separated list of property values.

`propertyValue`

Value of a property.

Examples

```
OMBCREATE PROCEDURE 'proc' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME,
IMPLEMENTATION) VALUES ('this is a Procedure', 'proc', 'BEGIN END proc \;')
```

```
ADD PARAMETER 'PARAM_1'
SET PROPERTIES (DEXSCRIPTION, BUSINESS_NAME, IN_OUT, DATATYPE,
DEFAULT_VALUE) VALUES ('param_1', 'this is a param_1','IN', 'VARCHAR2',
'this is a Varchar2')
ADD PARAMETER 'PARAM_2'
SET PROPERTIES (DEXSCRIPTION, BUSINESS_NAME, IN_OUT, DATATYPE,
DEFAULT_VALUE) VALUES ('param_2', 'this is a param_2','INOUT', 'DATE',
'this is a Date')
```

This will create a Procedure named "proc", its description is "this is a Procedure", and business name is "proc", return datatype NUMBER, and body of function as 'BEGIN END proc;'. It creates two parameters 'PARAM_1' and 'PARAM_2'

See Also

OMBCREATE, OMBALTER PROCEDURE, OMBDROP PROCEDURE

OMBCREATE PROCESS_FLOW

Purpose

To create a Process Flow.

Prerequisites

Should be in the context of a Process Flow Package.

Syntax

```
createProcessFlowCommand = OMBCREATE ( PROCESS_FLOW "QUOTED_STRING" [ SET
  ( "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] ( { ADD "addProcessParameterClause" }
  { ADD "addProcessVariableClause" } { ADD "addActivityClause" } { ADD
    "addTransitionClause" } { ADD "addUserDefinedParameterClause" } ) )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
  "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addProcessParameterClause = ( PARAMETER "QUOTED_STRING" ) [ SET
  "setPropertiesClause" ]
addProcessVariableClause = ( VARIABLE "QUOTED_STRING" ) [ SET
  "setPropertiesClause" ]
addActivityClause = ( "addStandardActivityClause" | "addMapActivityClause"
  | "addTemplateActivityClause" | "addDataAuditorActivityClause" |
  "addFunctionActivityClause" | "addSubProcessActivityClause" )
addTransitionClause = ( TRANSITION "QUOTED_STRING" ( FROM ACTIVITY
  "QUOTED_STRING" ) ( TO "QUOTED_STRING" ) ) [ SET "setPropertiesClause"
  ]
addUserDefinedParameterClause = ( PARAMETER "QUOTED_STRING" OF
  USER_DEFINED ACTIVITY "QUOTED_STRING" ) [ SET "setPropertiesClause" ]
propertyNameList = ( "UNQUOTED_STRING" | BINDING ) { ", " (
  "UNQUOTED_STRING" | BINDING ) }
propertyValueList = "propertyValue" { ", " "propertyValue" }
addStandardActivityClause = ( "UNQUOTED_STRING" | USER_DEFINED ) ACTIVITY
  "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ]
addMapActivityClause = ( MAPPING ACTIVITY "QUOTED_STRING" [ SET (
  PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF | REFERENCE )
    "setPropertiesAndReferencesMapClauses" ] [ "setReferenceIconSetClause"
    ] ) ] | ( REF | REFERENCE ) "setPropertiesAndReferencesMapClauses" [
    SET "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addTemplateActivityClause = ( ACTIVITY_TEMPLATE ACTIVITY "QUOTED_STRING" [
  SET ( PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF |
  REFERENCE ) "setPropertiesAndReferencesTemplateClause" ] [
    "setReferenceIconSetClause" ] ) ] | ( REF | REFERENCE )
    "setPropertiesAndReferencesTemplateClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addDataAuditorActivityClause = ( DATA_AUDITOR ACTIVITY "QUOTED_STRING" [
  SET ( PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF | REFERENCE
  ) "setPropertiesAndReferencesDataAuditorClauses" ] [
    "setReferenceIconSetClause" ] ) ] | ( REF | REFERENCE )
    "setPropertiesAndReferencesDataAuditorClauses" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addFunctionActivityClause = ( TRANSFORMATION ACTIVITY "QUOTED_STRING" [
  SET ( PROPERTIES "collectPropertiesClause" [ SET ( [ ( REF | REFERENCE
  ) "setPropertiesAndReferencesFunctionClauses" ] [
    "setReferenceIconSetClause" ] ) ] | ( REF | REFERENCE )
```

```

        "setPropertyAndReferencesFunctionClauses" [ SET
        "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
addSubProcessActivityClause = ( SUBPROCESS ACTIVITY "QUOTED_STRING" SET (
    PROPERTIES "collectPropertiesClause" SET ( REF | REFERENCE )
    "setPropertyAndReferencesSubProcessClauses" [ SET
    "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setPropertyAndReferencesSubProcessClauses" [ SET
    "setReferenceIconSetClause" ] ) )
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
collectPropertiesClause = (" "propertyNameList" ") VALUES "("
    "propertyValueList" ")"
setPropertyAndReferencesMapClauses = ( MAPPING "QUOTED_STRING" )
setPropertyAndReferencesTemplateClause = ( ACTIVITY_TEMPLATE
    "QUOTED_STRING" )
setPropertyAndReferencesDataAuditorClauses = ( DATA_AUDITOR
    "QUOTED_STRING" )
setPropertyAndReferencesFunctionClauses = ( TRANSFORMATION
    "QUOTED_STRING" )
setPropertyAndReferencesSubProcessClauses = ( PROCESS_FLOW
    "QUOTED_STRING" )

```

Keywords And Parameters

createProcessFlowCommand

Create a new process flow using quoted name.

setPropertyClause

Used to set properties (core, user-defined) for process flow. Note: For MAPPING, TRANSFORMATION and SUBPROCESS

activities the setPropertyAndReferencesMapClauses,

setPropertyAndReferencesFunctionClauses and

setPropertyAndReferencesSubProcessClauses respectively,

are mandatory.

For MAPPING or TRANSFORMATION activities and the REFERENCE property has to

be set to a

valid MAP or TRANSFORMATION within the current project.

For SUBPROCESS activities the REFERENCE property has to be set

to a SUBPROCESS within the same PROCESS_FLOW_PACKAGE.

Valid properties are as shown:

Base properties for PROCESS_FLOW:

Basic properties for Process Flow, Activity, Transition and Parameter:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Process Flow

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Process Flow Core properties for Transition :

Name: TRANSITION_CONDITION

Type: STRING

Valid Values: ", SUCCESS, ERROR, WARNING

Default: ", that is, Unconditional

Sets the Transition Condition of a Transition

Description of the Process Flow Core properties for Activity Parameter :

Name: DATATYPE

Type: STRING

Valid Values: INTEGER, FLOAT, DATE, STRING, BOOLEAN

Default: STRING

Sets the datatype of a Activity Parameter

Name: DIRECTION

Type: STRING

Valid Values: IN

Default: IN

Sets the direction of a Activity Parameter

Name: VALUE

Type: STRING

Valid Values: Examples '123', '123.456', 'Jan-08-2003', 'I am String',
'true'

Default: "

For Mapping activities representing PLSQL maps, the allowed value for the parameters:

OPERATING_MODE:'SET_BASED' 'ROW_BASED' 'ROW_BASED_TARGET_ONLY'
 'SET_BASED_FAIL_OVER_TO_ROW_BASED'
 'SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY'
 AUDIT_LEVEL:'NONE' 'STATISTICS' 'ERROR_DETAILS' 'COMPLETE'
 Sets the value of a Activity Parameter

Name: BINDING

Type: STRING

Valid Values: Examples 'PARAM_1', 'PARAM_2'

Default: "

Represents the parameter on the process flow that this parameter is bound to.

When setting users can specify the name of any PROCESS PARAMETER of same datatype.

This feature allows for parameterizing the process flow. If the parameter is bound

the VALUE property is ignored when generating the process flow.

To unbind a parameter, use an empty quoted string, that is "", and the parameter will be unbound.

Properties for PROCESS_FLOW:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

addProcessParameterClause

This clause adds the Parameters for a Process Flow.

addActivityClause

This clause adds the Activities for a Process Flow.

addTransitionClause

This clause adds the Transitions for a Process Flow.

addUserDefinedParameterClause

This clause adds the Parameters for a User Defined Activity of a Process Flow.

propertyNameList

A comma delimited set of property names to set.

propertyValueList

A comma delimited set of property values to set.

addStandardActivityClause

This clause adds standard activity types AND, EMAIL, END_ERROR, END_WARNING, END_SUCCESS, FILE_EXISTS, FORK, FTP, OR, ASSIGN, END_LOOP, FOR_LOOP, MANUAL, NOTIFICATION, ROUTE, SET_STATUS, SQLPLUS, WAIT, WHILE_LOOP OR USER_DEFINED to a Process Flow.

addMapActivityClause

This clause adds the MAP activity to a Process Flow.

addTemplateActivityClause

This clause adds an ACTIVITY_TEMPLATE as an activity to a Process Flow.

`addDataAuditorActivityClause`

This clause adds a DATA_AUDITOR activity to a Process Flow.

`addFunctionActivityClause`

This clause adds the Function or Procedure activity to a Process Flow.

`addSubProcessActivityClause`

This clause adds a Process as an activity to a Process Flow.

`propertyValue`

Integer value, float value or quoted string literal.

`collectPropertiesClause`

This clause collects core properties of Map, Function/Procedure and Subprocess activity.

`setPropertyAndReferencesMapClauses`

This clause sets reference to the existing Map.

`setPropertyAndReferencesFunctionClauses`

This clause sets a reference to existing Function or Procedure.

`setPropertyAndReferencesSubProcessClauses`

This clause sets a reference to existing Process Flow.

Examples

```
OMBCREATE PROCESS_FLOW 'TEST_PROCESS_FLOW'
SET PROPERTIES (DESCRIPTION, BUSINESS_NAME) VALUES ('this is a Process
Flow', 'process flow')
ADD PARAMETER 'PARAM_1'
ADD PARAMETER 'PARAM_2'
ADD FORK ACTIVITY 'FORK_ACTIVITY'
ADD USER_DEFINED ACTIVITY 'UD_ACTIVITY'
ADD FTP ACTIVITY 'FTP_ACTIVITY'
ADD OR ACTIVITY 'OR_ACTIVITY'
ADD AND ACTIVITY 'AND_ACTIVITY'
ADD MAPPING ACTIVITY 'ACTIVITY_MAP_1'
```



```

SET REFERENCE MAPPING '/PROCESS_FLOW_PROJECT/WAREHOUSE_P/MAP_
1'
ADD SUBPROCESS ACTIVITY 'SUBPROCESS_ACTIVITY'
SET REFERENCE PROCESS_FLOW 'REPORT_PROCESS_FLOW'
ADD MAPPING ACTIVITY 'ACTIVITY_MAP_2'
SET REFERENCE MAPPING '/PROCESS_FLOW_PROJECT/WAREHOUSE_P/MAP_
2'
ADD TRANSITION 'T1' FROM ACTIVITY 'START' TO 'FORK_ACTIVITY'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('SUCCESS')
ADD TRANSITION 'T2' FROM ACTIVITY 'FORK_ACTIVITY' TO 'FTP_ACTIVITY'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('SUCCESS')
ADD TRANSITION 'T3' FROM ACTIVITY 'FORK_ACTIVITY' TO 'ACTIVITY_MAP_
1'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('SUCCESS')
ADD TRANSITION 'T4' FROM ACTIVITY 'FTP_ACTIVITY' TO 'OR_ACTIVITY'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('SUCCESS')
ADD TRANSITION 'T5' FROM ACTIVITY 'FTP_ACTIVITY' TO 'UD_ACTIVITY'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('ERROR')
ADD TRANSITION 'T6' FROM ACTIVITY 'UD_ACTIVITY' TO 'OR_ACTIVITY'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('SUCCESS')
ADD TRANSITION 'T7' FROM ACTIVITY 'OR_ACTIVITY' TO 'AND_ACTIVITY'
ADD TRANSITION 'T8' FROM ACTIVITY 'ACTIVITY_MAP_1' TO 'AND_ACTIVITY'
ADD TRANSITION 'T9' FROM ACTIVITY 'ACTIVITY_MAP_1' TO 'SUBPROCESS_
ACTIVITY'

SET PROPERTIES (TRANSITION_CONDITION) VALUES ('WARNING')
ADD TRANSITION 'T10' FROM ACTIVITY 'AND_ACTIVITY' TO 'ACTIVITY_MAP_
2'
ADD TRANSITION 'T11' FROM ACTIVITY 'ACTIVITY_MAP_2' TO
'SUBPROCESS_ACTIVITY'
SET PROPERTIES (TRANSITION_CONDITION) VALUES ('WARNING')
ADD TRANSITION 'T12' FROM ACTIVITY 'ACTIVITY_MAP_2' TO 'END'
ADD TRANSITION 'T13' FROM ACTIVITY 'SUBPROCESS_ACTIVITY' TO 'END'
ADD PARAMETER 'PARAM_1' OF USER_DEFINED ACTIVITY 'UD_ACTIVITY'

```

Prerequisite for this examples are the existence of a MAP_1, MAP_2 and a Process Flow 'REPORT_PROCESS_FLOW'.

The example here will create a process flow and its parameters, it creates activity of types FTP, AND, OR, FORK, MAPPING, USER_DEFINED, and

SUBPROCESS. It creates various types of transitions among these activities.
At the end it creates a parameter for a user_defined activity type.

See Also

OMBCREATE, OMBALTER PROCESS_FLOW, OMBDROP PROCESS_FLOW

OMBCREATE PROCESS_FLOW_MODULE

Purpose

To create a Process Flow Module.

Prerequisites

Should be in the context of a project.

Syntax

```
createProcessFlowModuleCommand = OMBCREATE ( PROCESS_FLOW_MODULE
    "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
        "setReferenceClause" ] | "setReferenceClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClause = ( "setReferenceLocationClause" [ SET
    "setReferenceIconSetClause" ] | "setReferenceIconSetClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createProcessFlowModuleCommand

Create a new process flow module.

setPropertiesClause

Used to set properties (core, user-defined) for process flow module.

Base properties for PROCESS_FLOW_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Process Flow Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Process Flow Module

propertyNameList

Comma-delimited list of property names. Property names are not in quotation marks.

propertyValueList

Comma-delimited list of property values.

setReferenceLocationClause

Set a location to a supported workflow engine.

propertyValue

Value of a property.

Examples

```
OMBCREATE PROCESS_FLOW_MODULE 'process_Module' SET PROPERTIES
(DESCRIPTION,
BUSINESS_NAME) VALUES ('this is a Process Flow Module', 'process flow
module')
```

This will create a Process Flow Module named "process_Module", its description is "this is a Process Flow Module", and business name is "process flow module".

See Also

OMBCREATE, OMBALTER PROCESS_FLOW_MODULE, OMBDROP PROCESS_FLOW_MODULE

OMBCREATE PROCESS_FLOW_PACKAGE

Purpose

To create a Process Flow Package.

Prerequisites

Should be in the context of a Process Flow Module.

Syntax

```
createProcessFlowPackageCommand = OMBCREATE ( PROCESS_FLOW_PACKAGE
    "QUOTED_STRING" [ SET ( "setPropertiesClause" [ SET
        "setReferenceIconSetClause" ] | "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = ( "UNQUOTED_STRING" | BINDING ) { "," (
    "UNQUOTED_STRING" | BINDING ) }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createProcessFlowPackageCommand

Create a new process flow package.

setPropertiesClause

Used to set properties (core, user-defined) for process flow packages.

Valid properties are as shown:

Basic properties for PROCESS_FLOW_PACKAGE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Process Flow Package

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Process Flow Package

Properties for PROCESS_FLOW_PACKAGE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "OWF.PACKAGES.DEPLOYABLE:DESCRIPTION"

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

propertyNameList

A comma delimited set of property names to set.

propertyValueList

A comma delimited set of property values to set.

propertyValue

Integer value, float value or quoted string literal.

Examples

```
OMBCREATE PROCESS_FLOW_PACKAGE 'process_Package' SET PROPERTIES
(DESCRIPTION, BUSINESS_NAME) VALUES ('this is a Process Flow Package',
'process flow package')
```

This will create a Process Flow Package named "process_Package", its description is "this is a Process Flow Package", and business name is "process flow package".

See Also

OMBCREATE, OMBALTER PROCESS_FLOW_PACKAGE, OMBDROP PROCESS_FLOW_PACKAGE

OMBCREATE PROJECT

Purpose

To create a project.

Prerequisites

Should be in the top level context.

Syntax

```
createProjectCommand = OMBCREATE PROJECT "QUOTED_STRING" ( [ SET (
    "setPropertiesClause" [ SET "setReferenceIconSetClause" ] |
    "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createProjectCommand

Create a project.

setPropertiesClause

Associate a set of properties with a project.

Basic properties for PROJECT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Project

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Project

setReferenceIconSetClause

Set icon set for the project.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE PROJECT 'New Project' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME)
```

```
VALUES ('this is a project', 'payroll project')
```

This will create a project named "New Project", its description is "this is a project", and business name is "payroll project".

See Also

OMBCREATE, OMBALTER PROJECT, OMBDROP PROJECT

OMBCREATE QUEUE_PROPAGATION

Purpose

To create a Queue Propagation.

Prerequisites

Should be in the context of an Advanced Queue or Streams Queue. The Target Queue can exist in any Oracle Module.

Syntax

```
createQPCommand = OMBCREATE ( QUEUE_PROPAGATION "QUOTED_STRING" [ SET
    "setQPPropertiesClause" ] )
setQPPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createQPCommand

Creates a Queue Propagation with the given name.

setQPPropertiesClause

Sets properties (core, logical, physical, user-defined) for Queue Propagation. Valid properties are as shown:

Basic properties for QUEUE_PROPAGATION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Queue Propagation

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Queue Propagation

Name: TARGET_QUEUE

Type: STRING(4000)

Valid Values: N/A

Default: "

Target Queue for the Queue Propagation. This has to be the name of a Queue existing in any Oracle Module.

Properties for QUEUE_PROPAGATION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DURATION

Type: STRING

Valid Values: N/A

Default: "

The duration of proagation to be done. The default value is null.Applicable only for non-streams queue propagation.

Name: GENERATE_DBLINK

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Database Link which is used for propagation

Name: GENERATE_QUEUE_PROPAGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Queue Propagation

Name: GENERATE_REPLICATION_RULE

Type: BOOLEAN

Valid Values: true, false

Default: false

Generate Ruleset and Rule for Replication purpose in Streams queue propagation

Name: GENERATE_SCHEDULE_PROPAGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Scheduling propagation. Applicable only for non-streams queue propagation.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LATENCY

Type: STRING

Valid Values: N/A

Default: 60

The latency for the queue propagation. By default the value is 60. Applicable only for non-streams queue propagation.

Name: NEXT_TIME

Type: STRING

Valid Values: N/A

Default: "

Next time when the propagation to be done. The default value is null. Applicable only for non-streams queue propagation.

Name: NOT_PERMITTED_TAG_VALUES

Type: STRING

Valid Values: N/A

Default: "

List of comma separated Tag values (in Hex numbers) which are not allowed

for propagation

Name: PERMITTED_TAG_VALUES

Type: STRING

Valid Values: N/A

Default: "

List of comma separated Tag values (in Hex numbers) which are allowed for propagation

Name: START_TIME

Type: STRING

Valid Values: N/A

Default: SYSDATE

The start time for the propagation to happen. The default value is SYSDATE. Applicable only for non-streams queue propagation.

Name: SUBSCRIBER_TRANSFORMATION

Type: STRING

Valid Values: N/A

Default: "

A Transformation that will be applied before propagation to the target queue. Applicable only for non-streams queue propagation.

Name: SUBSCRIBER_RULE_CONDITION

Type: STRING

Valid Values: N/A

Default: "

A Rule condition to check whether the message can be propagated to the subscriber. Applicable only for non-streams queue propagation.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE QUEUE_PROPAGATION 'NEW_QUEUE_PROPAGATION' SET
PROPERTIES
```

```
(DESCRIPTION, TARGET_QUEUE ) VALUES ('this is a Queue Propagation',
'SOME_QUEUE')
```

This will create a Queue Propagation named "NEW_QUEUE_PROPAGATION", its description is "this is a Queue Propagation" and its Queue Table 'SOME_QUEUE'.

See Also

```
OMBCREATE QUEUE_PROPAGATION, OMBALTER QUEUE_PROPAGATION,
OMBDROP QUEUE_PROPAGATION
```

OMBCREATE QUEUE_TABLE

Purpose

To create a Queue Table.

Prerequisites

Should be in the context of an Oracle Module. The Object Type can exist in any Oracle Module.

Syntax

```
createQTCommand = OMBCREATE ( QUEUE_TABLE "QUOTED_STRING" [ SET
    "setQTPropertiesClause" ] )
setQTPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createQTCommand

Creates a Queue Table with the given name.

setQTPropertiesClause

Sets properties (core, logical, physical, user-defined) for Queue Table.

Valid properties are as shown:

Basic properties for QUEUE_TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Queue Table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Queue Table

Name: PAYLOAD_TYPE

Type: STRING(4000)

Valid Values: N/A

Default: "

Object Type for the Queue Table. This has to be the name of an Object Type (OBJECT_TYPE) existing in any Oracle Module.

Properties for QUEUE_TABLE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: GENERATE_QUEUE_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate code to create the queue table that will persist the messages of this Advanced Queue.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE QUEUE_TABLE 'NEW_QUEUE_TABLE' SET PROPERTIES  
(DESCRIPTION,  
PAYLOAD_TYPE ) VALUES ('this is a Queue Table', 'SOME_TYPE')
```

This will create a Queue Table named "NEW_QUEUE_TABLE", its description is "this is a Queue Table" and its Object Type 'SOME_TYPE'.

See Also

OMBCREATE QUEUE_TABLE, OMBALTER QUEUE_TABLE, OMBDROP QUEUE_TABLE

OMBCREATE REAL_TIME_MAPPING

Purpose

Create a Real Time mapping in an Oracle Module.

Prerequisites

1. The current context of scripting must be an Oracle Module.
2. No concurrent user should be locking the Oracle Module or any of its ancestors exclusively at the moment the map is being created.
3. The map name must not conflict with existing map names and the maps names that concurrent user tries to use.

Syntax

```
createRealTimeMappingCommand = OMBCREATE REAL_TIME_MAPPING "mappingName"
    "createOperatorOwnerDetailClause"
mappingName = "QUOTED_STRING"
createOperatorOwnerDetailClause = [ SET ( ( "setPropertiesClause" [ SET
    "setReferenceIconSetClause" ] ) | "setReferenceIconSetClause" ) ]
    "createOperatorOwnerDescendantsClause"
setPropertiesClause = PROPERTIES "propertyKeyList" VALUES
    "propertyValueList"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
createOperatorOwnerDescendantsClause = { ADD ( "addOperatorClause" |
    "addGroupClause" | "addAttributeClause" | "addChildClause" |
    "addConnectionClause" ) }
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
propertyValueList = "( " "propertyValue" { "," "propertyValue" } )"
addOperatorClause = "operatorType" OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ] [ SET "setPropertiesClause" ] [
    "setBindingClause" ]
addGroupClause = "groupDirection" GROUP "groupName" OF
    "operatorBottomUpLocator" [ SET "setPropertiesClause" ]
addAttributeClause = ATTRIBUTE "attributeName" OF "groupBottomUpLocator" [
    SET "setPropertiesClause" ]
addChildClause = "childType" "childName" "childOwnerBottomUpLocator" [ SET
    "setPropertiesClause" ]
addConnectionClause = CONNECTION FROM ( "groupBottomUpLocator" TO
    "groupBottomUpLocator" [ "groupToGroupConnectType" ] |
    "attributeBottomUpLocator" TO ( "attributeBottomUpLocator" |
    "attributesBottomUpLocator" | "groupBottomUpLocator" ) |
    "attributesBottomUpLocator" TO ( "attributesBottomUpLocator" |
    "groupBottomUpLocator" ) )
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
operatorName = "QUOTED_STRING"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
setBindingClause = BOUND TO "bindableLocator"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupName = "QUOTED_STRING"
operatorBottomUpLocator = OPERATOR "operatorName" [
```

```

"pluggableMapBottomUpLocator" ]
attributeName = "QUOTED_STRING"
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
childType = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
childOwnerBottomUpLocator = { OF "childType" "childName" } [ OF
    "mappableBottomUpLocator" ]
groupToGroupConnectType = COPY ALL | BY ( NAME [ IGNORE ( SPECIAL_CHARS
    "QUOTED_STRING" | SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX
    "QUOTED_STRING" | TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX
    "QUOTED_STRING" ) { ", " IGNORE ( SPECIAL_CHARS "QUOTED_STRING" |
    SOURCE_PREFIX "QUOTED_STRING" | SOURCE_SUFFIX "QUOTED_STRING" |
    TARGET_PREFIX "QUOTED_STRING" | TARGET_SUFFIX "QUOTED_STRING" ) } ] |
    POSITION )
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
attributesBottomUpLocator = ATTRIBUTES "attributeNameList" OF
    "groupBottomUpLocator"
pluggableMapName = "QUOTED_STRING"
bindableLocator = "bindableType" "bindableName" [ OF "bindableType"
    "bindableName" ]
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
attributeNameList = "(" "attributeName" { ", " "attributeName" } ")"
bindableType = PLUGGABLE_MAPPING | OBJECT_TYPE | "UNQUOTED_STRING"
bindableName = "QUOTED_STRING"

```

Keywords And Parameters

mappingName

Name of the mapping.

createOperatorOwnerDetailClause

Create the desired detail of a pluggable mapping.

setPropertyList

Describe the keys of properties for the map or objects in the map.

createOperatorOwnerDescendantsClause

Create the desired child objects of a mapping or a pluggable mapping.

propertyKeyList

The list of property keys.

propertyValueList

A list of property values.

addOperatorClause

Adds a mapping operator to a map. When you add an operator, Warehouse Builder creates default groups and parameters for the operator. Please see the appendix section of the Scripting Reference.

The following is an example for creating an operator:

```
OMBALTER MAPPING 'M1' ADD TABLE OPERATOR 'T1'
```

`addGroupClause`

Add a mapping group to a mapping operator.

`addAttributeClause`

Add a mapping attribute to a mapping group.

`addChildClause`

Add a child to a mapping, mapping operator, mapping group or mapping attribute.

The following is an example for creating a child object under a mapping

```
OMBALTER MAPPING 'M1' ADD SOURCE_DATA_FILE 'FILE1'
```

Note: Key word "OPERATOR" "GROUP" "ATTRIBUTE" are important for their respective ADD clauses. Without the key words, OMBPlus will interpret the ADD clause as an `addChildClause`. Here is an example:

```
OMB+> OMBALTER MAPPING 'M1' ADD TABLE 'T1'
```

```
OMB02932: Error getting child objects of type TABLE in M1
```

OMBPLUS interprets the ADD clause as one for creating a non-operator child object under the mapping object. Therefore, it tries to find type definition for non-operator child object "TABLE" and cannot find it.

`addConnectionClause`

Add connections between mapping groups or mapping attributes.

`propertyKey`

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the attribute

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the attribute

Name: DATATYPE
Type: STRING(20)
Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARHCAR, VARCHAR2, XMLTYPE
Default: "
Datatype of the Attribute

Name: LENGTH
Type: NUMBER
Valid Values: N/A
Default: 0
Length of the attribute.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Precision of the attribute.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for MAPPING:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_LANGUAGE

Type: STRING

Valid Values: ABAP, PLSQL, SQLLOADER, UNDEFINED

Default: UNDEFINED

The language used when generating code for the mapping.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE,
NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_
FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,

NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,

NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,

NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,

NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,

NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
 NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
 NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME,
 NA_LOCALITY_2,
 NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER,
 NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
 NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4, NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no

lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE of SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | | =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder

determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is

specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/append, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key

"MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key

"MATCHMERGE.GENERAL.MATCH_NEW_RECORD_CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data. Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key

"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this

target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT
Type: STRING
Valid Values: N/A
Default: "
Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT
Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this
operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name,
so all errors generated by a particular input record will be rolled up into
a single record with the error names concatenated in the error name
attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValue

A single property value. It can be a number, float, boolean or single-quoted string.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDuplicATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCast, LCRSPplitter, MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_PROCESS,
SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION,
TRANSFORMATION, UNPIVOT, VIEW.

operatorName

Name of a mapping operator.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

setBindingClause

Set the binding during the creation of a mapping operator or mapping attribute.

groupDirection

Direction of a mapping group.

groupName

Name of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

attributeName

Name of a mapping attribute.

groupBottomUpLocator

Location of a mapping group.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or

mapping attribute.

childOwnerBottomUpLocator

Location of a child owner. A child owner can be a map, mapping operator, mapping group, mapping attribute or a child.

groupToGroupConnectType

Connecting from a mapping group in one mapping operator to a mapping group in another mapping operator.

attributeBottomUpLocator

Location of a mapping attribute.

attributesBottomUpLocator

Location of a list of mapping attributes.

pluggableMapName

Name of the pluggable map.

bindableLocator

Location of the object to be bound to a mapping operator or mapping attribute.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

attributeNameList

A list of attribute names.

bindableType

Type of object bound to a mapping operator or mapping attribute.

bindableName

Name of the object bound to a mapping operator or mapping attribute.

Examples

```
OMBCREATE REAL_TIME_MAPPING 'MAP1'
```

```
OMBCREATE REAL_TIME_MAPPING 'MAP1'  
SET PROPERTIES (business_name, description)  
VALUES ('My map', 'Map to load customer look up table')  
ADD VARIABLE 'LAST_CUST' SET PROPERTIES (SCALE, PRECISION) VALUES  
(10,20)  
ADD TABLE OPERATOR 'CUST_SRC'  
BOUND TO TABLE '../SRC_MODULE/CUST_SRC'  
ADD TABLE OPERATOR 'CUST_LOOK_UP'  
ADD CONNECTION FROM GROUP 'INOUTGRP1' OF OPERATOR 'CUST_SRC'  
TO GROUP 'INOUTGRP1' OF OPERATOR 'CUST_LOOK_UP'
```

See Also

OMBCREATE, OMBALTER REAL_TIME_MAPPING, OMBRETRIEVE REAL_TIME_MAPPING, OMBDROP REAL_TIME_MAPPING

OMBCREATE REGISTERED_FUNCTION

Purpose

Creates a function that can be used in a query.

Prerequisites

Should be in the context of a Business Definition Module.

Syntax

```
createRegisteredFunctionCommand = OMBCREATE REGISTERED_FUNCTION
    "QUOTED_STRING" [ SET "setPropertiesClauseforRegFunforCreate" ] [ SET
    "setReferenceIconSetClause" ] { "addFunArgClauses" }
setPropertiesClauseforRegFunforCreate = PROPERTIES "("
    "propertyNameListforRegFun" ")" VALUES "(" "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addFunArgClauses = ADD PARAMETER "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
propertyNameListforRegFun = ( "UNQUOTED_STRING" | ( PACKAGE | SIGNATURE )
    ) { "," ( "UNQUOTED_STRING" | ( PACKAGE | SIGNATURE ) ) }
propertyValueList = "propertyValue" { "," "propertyValue" }
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

createRegisteredFunctionCommand

This command creates a registered function.

QUOTED_STRING

Specify the name of the function to be created.

setPropertiesClauseforRegFunforCreate

This clause sets the properties of the object.

setReferenceIconSetClause

Set specified Icon Set.

addFunArgClauses

This clause is for adding parameters to a function.

propertyNameListforRegFun

This is the list of property names.

propertyValueList

This is the list of property values.

setPropertyClause

Used to set properties (core, logical, physical, user-defined) for functions. Valid properties are as shown:

Basic properties for REGISTERED_FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the function

Name: AVAILABLE

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the Function is available for the user to use in calculations

Name: RETURN_TYPE

Type: STRING()

Valid Values: BINARY_INTEGER, BLOB, BOOLEAN, CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, SYS.ANYDATA, SYS.ROW_LCR, SYS_REFCURSOR, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE, SYS.XMLFORMAT,

BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
 SYS.XMLSEQUENCETYPE,
 BLAST_MATCH_PLSQLRECORDTYPE

Default: 'NUMBER'

Return type of the function

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the parameter

Name: DATATYPE

Type: STRING()

Valid Values: BINARY_INTEGER, BLOB, BOOLEAN, CHAR, CLOB, DATE, FLOAT,
 INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH
 NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, SYS.ANYDATA,
 SYS.ROW_LCR, SYS_REFCURSOR, TIMESTAMP, TIMESTAMP WITH LOCAL TIME
 ZONE

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE,
 SYS.XMLFORMAT,

BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
 SYS.XMLSEQUENCETYPE,

BLAST_MATCH_PLSQLRECORDTYPE

Default: 'NUMBER'

Datatype of the parameter

Properties for REGISTERED_FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for the referenced Function

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: PACKAGE

Type: STRING

Valid Values: N/A

Default: "

May be used to identify the name of a Package that contains the Function

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

propertyValue

This is a property value.

propertyNameList

This is the list of property names.

Examples

```
OMBCREATE REGISTERED_FUNCTION 'My_Sum'
```

See Also

OMBALTER REGISTERED_FUNCTION, OMBRETRIEVE REGISTERED_FUNCTION

OMBCREATE ROLE

Purpose

To create a Warehouse Builder role.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseCreateRoleCommand = OMBCREATE ( ROLE "QUOTED_STRING" [ SET
    "setPropertiesClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`parseCreateRoleCommand`

This clause creates a Warehouse Builder role.

`setPropertiesClause`

Used to set properties of a Warehouse Builder role. Valid properties are as shown.

Basic properties for ROLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the role

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the role

Examples

```
OMBCREATE ROLE 'DEVELOPMENT_ROLE'
```

will create a role named 'DEVELOPMENT_ROLE'.

See Also

OMBALTER ROLE, OMBDROP ROLE, OMBRETRIEVE ROLE

OMBCREATE SAP_MODULE

Purpose

To create a SAP module. It is not supported in the current release.

Prerequisites

You must open a project to create a SAP module.

Syntax

```
createSAPModuleCommand = OMBCREATE ( SAP_MODULE "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET "setReferenceClauseForDataMetadataModule"
    ] | "setReferenceClauseForDataMetadataModule" ) ] [
    "addModuleReferenceLocationClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceClauseForDataMetadataModule = ( "setReferenceLocationClause" [
    SET "setReferenceMetadataLocationOrIconSetClause" ] |
    "setReferenceMetadataLocationOrIconSetClause" )
addModuleReferenceLocationClause = "addReferenceLocationClause" {
    "addReferenceLocationClause" }
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
setReferenceLocationClause = ( REFERENCE | REF ) LOCATION "QUOTED_STRING"
setReferenceMetadataLocationOrIconSetClause = (
    "setReferenceMetadataLocationClause" [ SET "setReferenceIconSetClause"
    ] | "setReferenceIconSetClause" )
addReferenceLocationClause = ADD ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" [ SET AS DEFAULT ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
setReferenceMetadataLocationClause = ( REFERENCE | REF ) METADATA_LOCATION
    "QUOTED_STRING"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
```

Keywords And Parameters

createSAPModuleCommand

Create a SAP module

setPropertiesClause

Associate a set of properties with an SAP module.

Base properties for SAP_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a SAP Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of an SAP Module

setReferenceClauseForDataMetadataModule

Set location and/or icon set for the SAP module.

addModuleReferenceLocationClause

Add runtime locations to the SAP module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

setReferenceLocationClause

Set a runtime location to the SAP module.

setReferenceMetadataLocationOrIconSetClause

Set metadata location and/or icon set for the SAP module.

addReferenceLocationClause

Add a runtime location to the SAP module.

propertyValue

Value of a property.

setReferenceMetadataLocationClause

Set metadata location for the SAP module.

setReferenceIconSetClause

Set icon set for the SAP module.

Examples

```
OMBCREATE SAP_MODULE 'src_module' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('this is an SAP module', 'source module')
```

This will create an SAP module named "src_module", its description is "this is an SAP module", and business name is "source module".

See Also

OMBCREATE, OMBALTER SAP_MODULE, OMBDROP SAP_MODULE

OMBCREATE SEQUENCE

Purpose

To create a sequence.

Prerequisites

In the context of an Oracle Module.

Syntax

```
createSequenceCommand = OMBCREATE ( SEQUENCE "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createSequenceCommand

This command creates a sequence.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for sequences., columns, unique keys, foreign keys, primary keys, and check constraints.

Basic properties for SEQUENCE:

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the sequence.

Name: CURRVAL

Type: NUMBER

Valid Values: N/A

Default: 1

current increment value.

Name: NEXTVAL

Type: NUMBER

Valid Values: N/A

Default: 1

next increment value. next increment value.

Properties for SEQUENCE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: INCREMENT_BY

Type: NUMBER

Valid Values: -2147483648 - 2147483647

Default: 1

Sequence Incremented By

Name: START_WITH

Type: NUMBER

Valid Values: -2147483648 - 2147483647

Default: 1

Sequence Starts With

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE SEQUENCE 'new_sequence' SET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME) VALUES ('this is a new sequence', 'New Sequence')
```

This will create a sequence named "NEW_SEQUENCE", its description is "this is a new sequence", and business name is "New Sequence".

See Also

OMBCREATE, OMBALTER SEQUENCE, OMBDROP SEQUENCE, OMBRETRIEVE SEQUENCE

OMBCREATE SNAPSHOT

Purpose

To create a snapshot of a component.

Prerequisites

Component on which snapshot is to be created should already exist. This command can be executed for any component regardless of current context.

Syntax

```
parseCreateCommand = OMBCREATE "createSnapshotCommand"
createSnapshotCommand = ( SNAPSHOT "QUOTED_STRING" [ WITH DEPENDEE_DEPTH (
    MAX | "INTEGER_LITERAL" ) ] [ SET "setPropertyClause" ]
    "addSnapshotFCOClauses" )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
addSnapshotFCOClauses = ( ADD "objectClause" )+
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
objectClause = "UNQUOTED_STRING" "QUOTED_STRING" [ CASCADE | NO CASCADE ]
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parseCreateCommand

Root production for OMBCREATE SNAPSHOT.

createSnapshotCommand

Clause for creating snapshots.

QUOTED_STRING

Name of the snapshot to be created.

DEPENDEE_DEPTH

Use this optional clause to include in the snapshot all dependees for each component in the command.

setPropertyClause

Optional clause to set the properties of a snapshot.

Basic properties for SNAPSHOT:

Name: TYPE

Type: STRING(200)
 Valid Values: FULL,SIGNATURE
 Default: FULL
 This is the type of snapshot

Name: DESCRIPTION
 Type: STRING(4000)
 Valid Values: N/A
 Default: "
 Description of the snapshot

PROPERTIES

Valid set of properties are DESCRIPTION and TYPE

VALUES

Values for the corresponding properties. Default values of properties for DESCRIPTION is null and valid values for TYPE are FULL or SIGNATURE

addSnapshotFCOClauses

Components to be added to the snapshot

propertyNameList

Property names for SNAPSHOT.

propertyValueList

List of property values for SNAPSHOT.

propertyValue

Allowable value types for a snapshot property.

Examples

```
OMBCREATE SNAPSHOT 'S1' SET PROPERTIES (DESCRIPTION, TYPE) VALUES ('this is
```

```
snapshot', 'FULL') ADD TABLE '/Project1/WH1/T1'
```

This will create a snapshot named S1 with the T1 table component in it.

```
OMBCREATE SNAPSHOT 'S1' WITH DEPENDEE_DEPTH 1 SET PROPERTIES  
(DESCRIPTION,  
TYPE) VALUES('this is snapshot with dependees', 'FULL') ADD MAPPING  
'/Project1/WH1/MAP1'
```

This will create snapshot named S1 with the MAP1 mapping component. This command will find all the dependee components, which in this case would be all the components which the map references. For example, if MAP1 contains T1,T2 and T3, then this WITH DEPENDEE_DEPTH 1 option will take a snapshot of the map MAP1 and tables T1,T2, and T3.

See Also

OMBALTER SNAPSHOT, OMBDROP SNAPSHOT, OMBRESTORE SNAPSHOT,
OMBCOMPARE SNAPSHOT, OMBLIST SNAPSHOT, OMBRETRIEVE SNAPSHOT

OMBCREATE STREAMS_CAPTURE_PROCESS

Purpose

To create a Streams Capture Process.

Prerequisites

Should be in the context of a Streams Queue.

Syntax

```
createCaptureCommand = OMBCREATE ( STREAMS_CAPTURE_PROCESS "QUOTED_STRING"
    [ "addTableClause"+ ] [ SET "setCapturePropertiesClause" ] )
addTableClause = ADD TABLE "QUOTED_STRING"
setCapturePropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES
    "(" "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createCaptureCommand

Create a Streams Capture Process. This Streams Capture Process will enqueue the changes it captures into the containing Streams Queue

addTableClause

Add a table to the set of tables whose changes are to be captured by this Streams Capture Process

setCapturePropertiesClause

Sets properties (core, logical, physical, user-defined) for Streams Capture Process.

propertyNameList

The list of properties.

Basic properties for STREAMS_CAPTURE_PROCESS:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Streams Capture Process

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Streams Capture

Properties for STREAMS_CAPTURE_PROCESS:

Name: CAPTURE_START_PARAMETER

Type: STRING

Valid Values: START_DATE, START_SCN

Default: START_SCN

This specifies whether the Streams Capture Process should start capturing changes based on the Start Date or the Start SCN.

Name: CAPTURE_TAGGED_LCR

Type: BOOLEAN

Valid Values: true, false

Default: false

If TRUE, then a redo entry is always considered for capture and an LCR is always considered for apply, regardless of whether redo entry or LCR has a non-NULL tag. If FALSE, then a redo entry is considered for capture and an LCR is considered for apply only when the redo entry or the LCR contains a NULL tag.

Name: CAPTURE_TIMEOUT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The maximum number of seconds to wait for another instance of the same capture process to finish.

Name: DBA_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location of the DBA user who should create the supplemental logs.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether the Object is deployable or not.

Name: DISABLE_ON_LIMIT

Type: BOOLEAN

Valid Values: true, false

Default: true

If set to true, the capture process will be disabled once the message/time limit is reached.

Name: MAXIMUM_SCN

Type: NUMBER

Valid Values: 0 - 1000000000

Default: 0

This is the Maximum SCN value whose corresponding changes will be captured by the Streams Capture Process.

Name: MESSAGE_COUNT_LIMIT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

If Disable On Limit is set to true, then the Streams Capture Process is disabled once the specified number of messages have been captured.

Name: PARALLELISM_DEGREE

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The number of parallel server process that will mine the redo logs.

Name: START_DATE

Type: STRING

Valid Values: N/A

Default: 1970-01-01

The user specified date from which the Streams Capture Process should start capturing changes.

Name: START_SCN

Type: NUMBER

Valid Values: N/A

Default: 0

The user specified SCN from which the Streams Capture Process should start capturing changes.

Name: TIME_LIMIT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

If Disable On Limit is set to true, then the Streams Capture Process is disabled once the specified number of seconds elapse.

Name: WRITE_ALERT_LOG

Type: BOOLEAN

Valid Values: true, false

Default: true

If set to true, then the Streams Capture Process writes a message to the alert log on exit.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE STREAMS_CAPTURE_PROCESS 'NEW_CAPTURE_PROCESS' ADD  
TABLE 'TABLE_1'
```

```
SET PROPERTIES (DESCRIPTION) VALUES ('this is a Capture Process')
```

This will create a Streams Capture Process named "NEW_CAPTURE_PROCESS", its description is "this is a Capture Process" and it captures changes to table 'TABLE_1'.

See Also

OMBCREATE, OMBALTER STREAMS_CAPTURE_PROCESS, OMBDROP
STREAMS_CAPTURE_PROCESS

OMBCREATE STREAMS_QUEUE

Purpose

To create an Streams Queue.

Prerequisites

Should be in the context of an Oracle Module. The Queue Table should exist in the same Oracle Module.

Syntax

```
createANYQCommand = OMBCREATE ( STREAMS_QUEUE "QUOTED_STRING" [ SET
    "setPropertyClause" ] )
setPropertyClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createANYQCommand

Creates a Streams Queue with the given name.

setPropertyClause

Sets properties (core, logical, physical, user-defined) for Streams Queue.

Valid properties are as shown:

Basic properties for STREAMS_QUEUE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Streams Queue

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description for the Streams Queue

Name: QTABLE

Type: STRING(4000)

Valid Values: N/A

Default: "

Queue Table for the Streams Queue. This has to be the name of a Queue Table(QUEUE_TABLE) existing in the same Oracle Module.

Properties for STREAMS_QUEUE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DEQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Dequeue Enabled for AQ

Name: ENQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enqueue enabled for AQ

Name: GENERATE_ADVANCED_QUEUE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the Advanced Queue.

Name: GENERATE_TEMPORARY_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the temporary table.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MAX_RETRIES

Type: NUMBER

Valid Values: N/A

Default: 5

Max. number of Retries

Name: RETENTION_TIME

Type: NUMBER

Valid Values: N/A

Default: 0

Message retention time

Name: RETRY_DELAY

Type: NUMBER

Valid Values: N/A

Default: 0

Delay period before retry

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE STREAMS_QUEUE 'NEW_STREAMS_QUEUE' SET PROPERTIES  
(DESCRIPTION,
```

```
QTABLE) VALUES ('this is a Streams Queue', 'SOME_QUEUE_TABLE')
```

This will create a Streams Queue named "NEW_STREAMS_QUEUE", its

description is "this is a Streams Queue" and its Queue Table

```
'SOME_QUEUE_TABLE'.
```

See Also

```
OMBCREATE STREAMS_QUEUE, OMBALTER STREAMS_QUEUE, OMBDROP  
STREAMS_QUEUE
```

OMBCREATE TABLE

Purpose

To create a table.

Prerequisites

In the context of an Oracle Module.

Syntax

```
createTableCommand = OMBCREATE ( TABLE "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] [ "addTableSCOClauses" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
addTableSCOClauses = ADD ( "addColumnClause" | "addConstraintClause" |
    "addSCOClauses" ) [ "addTableSCOClauses" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addColumnClause = COLUMN "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ]
    [ SET "setPropertiesClause" ]
addConstraintClause = "addUkPkClause" | "addFkClause" |
    "addCheckConstraintClause"
addSCOClauses = "addIndexClause" | "addIndexPartitionClause" |
    "addIndexPartitionKeyClause" | "addPartitionClause" |
    "addPartitionKeyClause" | "addSubpartitionClause" |
    "addaddMaterializedViewSCOandDependentClauseClause" |
    "addSubPartitionKeyClause" | "addIndexColumnClause"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
addUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" [ SET
    "setUkPkPropertiesAndReferencesColumnsClauses" ]
addFkClause = FOREIGN_KEY "QUOTED_STRING" [ SET "setFkSubClauses" ]
addCheckConstraintClause = CHECK_CONSTRAINT "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
addIndexClause = INDEX "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addIndexPartitionClause = INDEX_PARTITION "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] (
    "renameSCOConfigurationClause" [ SET
    "setSCOConfigurationPropertiesClauses" ] | [ SET
    "setSCOConfigurationPropertiesClauses" ] )
addIndexPartitionKeyClause = INDEX_PARTITION_KEY "QUOTED_STRING" OF INDEX
    "QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionClause = PARTITION "QUOTED_STRING" [ AT POSITION
    "INTEGER_LITERAL" ] [ SET "setSCOConfigurationPropertiesClauses" ]
addPartitionKeyClause = PARTITION_KEY "QUOTED_STRING" [ SET
    "setSCOConfigurationPropertiesClauses" ]
addSubpartitionClause = SUBPARTITION "QUOTED_STRING" OF PARTITION
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
    "setSCOConfigurationPropertiesClauses" ]
addaddMaterializedViewSCOandDependentClauseClause = TEMPLATE_SUBPARTITION
    "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ] [ SET
    "setSCOConfigurationPropertiesClauses" ]
addSubPartitionKeyClause = SUBPARTITION_KEY "QUOTED_STRING" [ SET
```

```

"setSCOConfigurationPropertiesClauses" ]
addIndexColumnClause = INDEX_COLUMN "QUOTED_STRING" OF INDEX
"QUOTED_STRING" [ SET "setSCOConfigurationPropertiesClauses" ]
setUkPkPropertiesAndReferencesColumnsClauses = "setPropertiesClause" [ SET
( REF | REFERENCE ) "constraintColumnReferencesClause" ] | ( REF |
REFERENCE ) "constraintColumnReferencesClause"
setFkSubClauses = "setPropertiesClause" [ SET ( REF | REFERENCE )
"setFkReferencesClauses" ] | ( REF | REFERENCE )
"setFkReferencesClauses"
setSCOConfigurationPropertiesClauses = PROPERTIES "(" "propertyNameList"
)" VALUES "(" "propertyValueList" ")"
renameSCOConfigurationClause = RENAME TO "QUOTED_STRING"
constraintColumnReferencesClause = COLUMNS "(" "quotedNameList" ")"
setFkReferencesClauses = "constraintColumnReferencesClause" [ SET ( REF |
REFERENCE ) "constraintUkReferencesClause" ] |
"constraintUkReferencesClause" [ SET ( REF | REFERENCE )
"constraintColumnReferencesClause" ]
quotedNameList = "QUOTED_STRING" { ", " "QUOTED_STRING" }
constraintUkReferencesClause = ( UNIQUE_KEY | PRIMARY_KEY )
"QUOTED_STRING" [ OF ( TABLE | VIEW ) "QUOTED_STRING" ]

```

Keywords And Parameters

createTableCommand

This command creates a table.

QUOTED_STRING

Specify the name of the table to be created.

setPropertiesClause

Used to set properties (core, logical, physical, user-defined) for tables (including partitions and subpartitons), and their columns, indexes (including index partitions), unique keys, foreign keys, primary keys, and check constraints.

Basic properties for TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the table

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Default value of the column

Name: NOT_NULL

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify "true" to enforce Not Null restriction on a column.

Basic properties for PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the table rows according to a Hash Algorithm, lists of values, or specified ranges.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH partitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of partitions that is a power of 2. If you have multiple Partition Keys, you only have to specify once.

Basic properties for PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Commas can be escaped using "" (for example, '1,2,3'). Always specify DEFAULT as the value of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH customized subpartitions the database should create for a particular main RANGE partition (RANGE-HASH BY QUANTITY partitioning). For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once. Set it to 0 to reverse to the use of generic template HASH_QUANTITY specified in SUBPARTITION_KEY.

Basic properties for SUBPARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

For partition-level partitioning according to a Hash Algorithm or lists of values. Each partition is further sorted into subpartitions.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH subpartitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once.

Basic properties for SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST subpartition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any SUBPARTITION. Each LIST subpartition must have at least one value. No value, including NULL, can appear in more than one subpartition.

Basic properties for TEMPLATE_SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

In composite partitioning, template subpartitions are automatically applied to those partitions without their subpartitions specified. Here for LIST subpartitions only, specify a comma-delimited, ordered list of literal values corresponding to the LIST subpartitioning column. Always specify DEFAULT as the value of the last template LIST subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any TEMPLATE_SUBPARTITION. Each LIST template subpartition must have at least one value. No value, including NULL, can appear in more than one template subpartition.

Basic properties for INDEX:

Name: INDEX_TYPE

Type: STRING

Valid Values: UNIQUE, NON-UNIQUE, BITMAP, FUNCTION-BASED

Default: (No default, must be one of the preceding choices)

Specify the type of an index. NORMAL can be used in place of NON-UNIQUE.

Name: LOCAL_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify if an index is Global or Local. The default is Global.

Specify Local so that the index is partitioned on the same columns, with the same number of partitions and the same partition bounds as table.

Oracle Database automatically maintains local index partitioning as the underlying table is repartitioned.

Name: COLUMN_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

Specify an expression built from columns of table, constants, SQL functions, and user-defined functions to create a FUNCTION-BASED index.

Basic properties for INDEX_PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the index rows according to a Hash Algorithm, lists of values, or specified ranges. Hash index partitioning is supported starting with Oracle Database 10g version.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH index partitions the database should create based on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of index partitions that is a power of 2. If you have multiple index Partition Keys, you only have to specify once.

Basic properties for INDEX_PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition. No need to specify VALUES_LESS_THAN for Local index.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last partition. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition. No need to specify VALUES_EQUAL_TO for Local index.

Basic properties for CHECK_CONSTRAINTS:

Name: CHECK_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Specify a condition that each row in the table must satisfy.

Properties for TABLE:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: CACHE_MODE

Type: STRING

Valid Values: , CACHE, NOCACHE

Default: "

Indicate how Oracle should store blocks in the buffer cache.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: MONITORING_MODE

Type: STRING

Valid Values: , MONITORING, NOMONITORING

Default: "

Specify MONITORING if you want modification statistics to be collected on this table.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow tablespaces. The number of tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of tablespaces, then Oracle cycles through the names of the

tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. For composite-partitioned tables, it is used for subpartition template to store a list of tablespaces.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: ROWDEPENDENCIES_MODE

Type: STRING

Valid Values: , NOROWDEPENDENCIES, ROWDEPENDENCIES

Default: "

Specify ROWDEPENDENCIES to use row-level dependency tracking.

Name: ROW_MOVEMENT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify whether Oracle can move a table row.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PRIMARY_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result

only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to

defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the

table. Specify `DISABLE` to disable the integrity constraint. The default is `ENABLE`.

Name: `EXCEPTIONSINTO`

Type: `STRING`

Valid Values: `N/A`

Default: `"`

Specify an exceptions table (`[schema.]table`). The `EXCEPTIONS` table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with `NOVALIDATE` option.

Name: `INITIALLY`

Type: `STRING`

Valid Values: `, DEFERRED, IMMEDIATE`

Default: `"`

Specify `(INITIALLY) IMMEDIATE` to indicate that Oracle should check a `DEFERRABLE` constraint at the end of each subsequent SQL statement. Specify `(INITIALLY) DEFERRED` to indicate that Oracle should check a `DEFERRABLE` constraint at the end of subsequent transactions. The default is `(INITIALLY) IMMEDIATE`.

Name: `ONDELETE`

Type: `STRING`

Valid Values: `, CASCADE, SET NULL`

Default: `"`

Specify `CASCADE` if you want Oracle to remove dependent foreign key values. Specify `SET NULL` if you want Oracle to convert dependent foreign key values to `NULL`.

Name: `RELY`

Type: `STRING`

Valid Values: `, NORELY, RELY`

Default: `"`

Specify `RELY` to activate an existing constraint in `NOVALIDATE` mode for query rewrite in an unenforced query rewrite integrity mode. The default is

NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for CHECK_CONSTRAINT:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The

default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for PARTITION:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size

in kilobytes or megabytes.

Name: INITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1 for Table and 2 for Index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used

space that Oracle maintains for each data block of the database object. The default is 40.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for INDEX:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: COMPUTESTATISTICS

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to collect statistics at relatively little cost during the creation of an index.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: INDEXORDER

Type: STRING

Valid Values: , ASC, DESC

Default: "

Use ASC or DESC to indicate whether the index should be created in ascending or descending order. The Oracle default is ASC.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2 for Index.

Name: KEYCOMPRESS

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Specify COMPRESS to enable key compression.

Name: KEYCOMPRESSPREFIXLENGTH

Type: STRING

Valid Values: N/A

Default: "

Specify the prefix length (number of prefix columns to compress). For unique indexes, the valid range of prefix length values is from 1 to the number of key columns minus 1. The default prefix length is the number of key columns minus 1. For nonunique indexes, the valid range of prefix length values is from 1 to the number of key columns. The default prefix length is the number of key columns. Oracle compresses only nonpartitioned indexes that are nonunique or unique indexes of at least two columns. You cannot specify COMPRESS for a bitmap index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (2-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: ONLINE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to indicate that DML operations on the table will be allowed during creation of the index.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of Index tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow Index tablespaces. The number of Index tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of Index tablespaces, then Oracle cycles through the names

of the Index tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for a locally partitioned index. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. If specified, then individual local Hash index partitions specified will be ignored for Local Hash or Range-Hash Index partitioning.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: SORT

Type: STRING

Valid Values: , NOSORT, REVERSE, SORT

Default: "

Specify NOSORT to indicate to Oracle that the rows are already stored in the database in ascending order. Specify REVERSE to store the bytes of the index block in reverse order, excluding the rowid.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addTableSCOClauses

This clause adds the secon class objects.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

addColumnClause

This clause adds a column.

When you create a table or alter a table to add a set of columns, the position that you specify for a column must be either less than or equal to the number of columns that you have listed up to that point in the command.

For example, the following OMBCREATE command does not add the specified columns to the table:

```
OMBCREATE TABLE 'MY_TABLE' \
SET PROPERTIES (DESCRIPTION) VALUES ('TRIAL TABLE') \
ADD COLUMN 'C1' \
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \
ADD COLUMN 'C2' AT POSITION 3 \
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \
ADD COLUMN 'C3' AT POSITION 2 \
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2')
```

This is because at the point when you specify the position of the column C2 as 3, you have added just two columns to the table. But the following OMBALTER command adds the specified columns to the table. This is because at the point when you specify the position of the column C2 as 2, you are adding the second column to the table.

```
OMBCREATE TABLE 'MY_TABLE' \
SET PROPERTIES (DESCRIPTION) VALUES ('TRIAL TABLE') \
ADD COLUMN 'C1' \
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \
ADD COLUMN 'C2' AT POSITION 2 \
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2') \
ADD COLUMN 'C3' AT POSITION 1 \
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10)
```

In the preceding example, the order in which the columns are added are as follows:

C1

C1, C2

C3, C1, C2

`addConstraintClause`

Adds primary and unique key, and add check constraints.

`addSCOClause`

This clause will add SCOs.

`propertyValue`

This clause adds the property values.

`addUkPkClause`

This clause adds the adds unique key and primary keys.

QUOTED_STRING

name of the unique key or primary key.

`addFkClause`

This clause adds foreign key.

QUOTED_STRING

Name of the foreign key.

`addCheckConstraintClause`

add a check constraint.

QUOTED_STRING

Name of the CheckConstraint.

`addIndexClause`

This clause adds an index.

QUOTED_STRING

Name of the index.

addPartitionClause

This clause adds a partition.

QUOTED_STRING

Name of the partition.

addPartitionKeyClause

This clause adds a partition key.

QUOTED_STRING

Name of the partition key. This should be a column identifier.

addIndexColumnClause

This clause will add index column to a specified index.

QUOTED_STRING

This should be a column identifier of owning object (such as a table) of the index.

setUkPkPropertiesAndReferencesColumnsClauses

This clause adds properties and references to columns.

setFkSubClauses

This clause set references to a foreign key.

setSCOConfigurationPropertiesClauses

Set the configuration properties for the following:

- Partition, Subpartition, and Template Subpartition: All refer to configuration properties of Partition.
- Index, and Index Partition: For Index Partition, refer to configuration properties of Partition.

renameSCOConfigurationClause

This clause renames configuration objects.

constraintColumnReferencesClause

This clause provides names of all columns.

setFkReferencesClauses

This clause sets foreign key references.

quotedNameList

This clause gives column names.

constraintUkReferencesClause

The first QUOTED_STRING denotes the UniqueKey or Primary key name, and the latter denotes the table's or view's name.

Examples

```
OMBCREATE TABLE 'new_table' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME)
```

```
VALUES ('this is a new table', 'New Table')
```

This will create a table named "NEW_TABLE", its description is "this is a new table", and business name is "New Table".

See Also

OMBCREATE, OMBALTER TABLE, OMBDROP TABLE, OMBRETRIEVE TABLE

OMBCREATE TABLE_FUNCTION

Purpose

To create a Table Function.

Prerequisites

Should be in the context of Oracle Module or Package. The REFCursorType and PLSQLTableType which are set as Datatype for parameters should preexist.

Syntax

```
createTableFunctionCommand = OMBCREATE ( TABLE_FUNCTION "QUOTED_STRING"
    SET "setPropertiesClause" [ SET "setReferenceIconSetClause" ] { ADD
    "addFuncProcParameterClause" } )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ( REFERENCE | REF ) ICONSET "QUOTED_STRING"
addFuncProcParameterClause = PARAMETER "QUOTED_STRING" [ SET
    "setPropertiesClause" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createTableFunctionCommand

Creates a table function

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Table Function and its Parameters. Valid properties are as shown:

Properties for TABLE_FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Table Function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Table Function

Name: PARALLEL_EXECUTION

Type: STRING(5)

Valid Values: TRUE, FALSE

Default: "

Enables Parallel Execution of the Table Function

Name: PIPELINED_EXECUTION

Type: STRING(5)

Valid Values: TRUE, FALSE

Default: "

Enables Partitioned Execution of the Table Function

Name: ORDER_METHOD

Type: STRING(9)

Valid Values: ORDERBY, CLUSTERBY

Default: ORDERBY

Order Method for the Table Function

Name: PARTITION_METHOD

Type: STRING(5)

Valid Values: NONE, ANY, HASH, RANGE

Default: NONE

Partition Method for the Table Function

Name: RETURN_TYPE

Type: STRING(4000)

Valid Values: N/A

Default: "

Name of the Return Type of this Table Function. For this release, this has to be a PLSQL Table Type whose datatype has to be PLSQLRecordType. Also, it should already be defined in USER_TYPES Package in this Module.

Name: IMPLEMENTATION

Type: STRING(4000)

Valid Values: N/A

Default: "

Implementation code for this Table Function.

Name: IS_DETERMINISTIC

Type: STRING(5)

Valid Values: TRUE, FALSE

Default: "

Whether this Table Function is Deterministic.

Properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING(4000)

Valid Values: Any valid REF cursor type

Default: "

Datatype of the Parameter. Parameter will always be IN type for Table Function. For this release, it has to be a REF Cursor type. Also, this REF Cursor should already be defined in USER_TYPES Package in this Module.

Properties for TABLE_FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be

executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addFuncProcParameterClause

Adds a Parameter with the given name and datatype to the Table Function.
The datatype of the Parameter is set as its DATATYPE property.

propertyNameList

Comma separated list of property names. Property names are unquoted.

propertyValueList

Comma separated list of property values.

propertyValue

Value of a property.

Examples

```
OMBCREATE TABLE_FUNCTION 'table_function' SET PROPERTIES
(DESCRIPTION,
RETURN_TYPE) VALUES ('this is a Table Function',
```

'my_module.my_package.plsql_table_type')

This will create a Table Function named "table_function" , its description is "this is a Table Function" and its return type will be PLSQL Table Type "plsql_table_type" existing under 'my_module.my_package'.

See Also

OMBCREATE, OMBALTER TABLE_FUNCTION, OMBDROP TABLE_FUNCTION

OMBCREATE TIME_DIMENSION

Purpose

This command creates a time dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
createTimeDimensionCommand = OMBCREATE TIME_DIMENSION "TimeDimensionName"
  ( ( "setFiscalPropertyClause" "addFiscalCalendarHierarchyClause"+ |
    "addCalendarHierarchyClause"+ [ "setFiscalPropertyClause"
    "addFiscalCalendarHierarchyClause"+ ] ) | ( "setPropertiesClause" (
    "setFiscalPropertyClause" "addFiscalCalendarHierarchyClause"+ | (
    "addCalendarHierarchyClause"+ [ "setFiscalPropertyClause"
    "addFiscalCalendarHierarchyClause"+ ] ) ) ) ) {
    "addDimensionRoleClause" } [ "implementationClause"
    "addSequenceClause" ] "populationClause"
TimeDimensionName = "QUOTED_STRING"
setFiscalPropertyClause = "setFiscalPropertiesClause"
addFiscalCalendarHierarchyClause = ADD FISCAL_CALENDAR HIERARCHY
  "hierarchyName" [ "setPropertiesClause" ] SET ( REF | REFERENCE )
  "fiscalCalendarLevelList"
addCalendarHierarchyClause = ADD ( ( NORMAL_CALENDAR HIERARCHY
  "hierarchyName" [ "setPropertiesClause" ] SET ( REF | REFERENCE )
  "normalCalendarLevelList" ) | ( WEEK_CALENDAR HIERARCHY
  "hierarchyName" [ "setPropertiesClause" ] SET ( REF | REFERENCE )
  "weekCalendarLevelList" ) )
setPropertiesClause = SET PROPERTIES "propertyKeyList" VALUES
  "propertyValueList"
addDimensionRoleClause = ADD DIMENSION_ROLE "roleName" [
  "setPropertiesClause" ]
implementationClause = IMPLEMENTED BY ( STAR | SNOWFLAKE ) [ USING
  COMPOSITE_UNIQUE_KEY ]
addSequenceClause = ( ( ADD ( REF | REFERENCE ) SEQUENCE "QUOTED_STRING" )
  | ( SET ( REF | REFERENCE ) SEQUENCE "QUOTED_STRING" ) )
populationClause = POPULATE DATA FROM "calendarYear" FOR "yearCount" YEARS
setFiscalPropertiesClause = SET FISCAL_CALENDAR PROPERTIES
  "propertyKeyList" VALUES "propertyValueList"
hierarchyName = "QUOTED_STRING"
fiscalCalendarLevelList = "( ( "fiscalCalendarLevelType" LEVEL [
  "levelName" ] [ "setPropertiesClause" ] ) { ", "
  "fiscalCalendarLevelType" LEVEL [ "levelName" ] [
  "setPropertiesClause" ] } )"
normalCalendarLevelList = "( ( "normalCalendarLevelType" LEVEL [
  "levelName" ] [ "setPropertiesClause" ] ) { ", "
  "normalCalendarLevelType" LEVEL [ "levelName" ] [
  "setPropertiesClause" ] } )"
weekCalendarLevelList = "( ( "weekCalendarLevelType" LEVEL [ "levelName"
  ] [ "setPropertiesClause" ] ) { ", " "weekCalendarLevelType" LEVEL [
  "levelName" ] [ "setPropertiesClause" ] } )"
propertyKeyList = "( "propertyKey" { ", " "propertyKey" } )"
propertyValueList = "( "propertyValue" { ", " "propertyValue" } )"
roleName = "QUOTED_STRING"
calendarYear = "INTEGER_LITERAL"
yearCount = "INTEGER_LITERAL"
```



```

fiscalCalendarLevelType = ( DAY | FISCAL_WEEK | FISCAL_MONTH |
    FISCAL_QUARTER | FISCAL_YEAR )
levelName = "QUOTED_STRING"
normalCalendarLevelType = ( DAY | CALENDAR_MONTH | CALENDAR_QUARTER |
    CALENDAR_YEAR )
weekCalendarLevelType = ( DAY | CALENDAR_WEEK )
propertyKey = "UNQUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )

```

Keywords And Parameters

TimeDimensionName

The name of the time dimension.

setFiscalPropertyClause

set the fiscal properties.

addFiscalCalendarHierarchyClause

This clause adds an fiscal hierarchy to the time dimension.

addCalendarHierarchyClause

This clause adds a new hierarchy to the time dimension by: renaming the hierarchy, setting of hierarchy properties, or setting level references.

setPropertyClause

This clause sets the following properties:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

addDimensionRoleClause

This clause adds a dimension role.

implementationClause

Time Dimension is implemented as STAR or as SNOWFLAKE.

addSequenceClause

Either create a new sequence and use it in the time dimension using ADD REF SEQUENCE ... statement, otherwise use SET REF SEQUENCE ... statement use to an existing sequence.

populationClause

This clause specifies the starting year and the number of years for which data will be populated.

setFiscalPropertiesClause

This clause sets the following properties:

Fiscal types allowed in OWB time dimension. Name: FISCAL_TYPE

Type: STRING

Valid Values: '544', '445'

Default: '544'

Fiscal calendar year start date, it could be any date of a year.

Name: FISCAL_CALENDAR_START_YEAR

Type: STRING

Valid Values: Dates in these format 'DD-MON-YYYY' or 'DD-MM-YYYY'

Default: '01-JAN-2000'

The day of the week when the fiscal year begins.

Name: FISCAL_CALENDAR_START_DAY_OF_WEEK

Type: STRING

Valid Values: 'MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY', 'SUNDAY'

Default: 'false'

hierarchyName

The name of a hierarchy.

fiscalCalendarLevelList

This clause creates a fiscal hierarchy and sets reference fiscal levels.

propertyKeyList

A list of time dimension properties.

propertyValueList

A list of time dimension property values.

roleName

A role name.

propertyKey

Basic properties for TIME DIMENSION, TIME DIMENSION MAP, DIMENSION_ATTRIBUTE, LEVEL, LEVEL_ATTRIBUTE and HIERARCHY:

Basic properties for TIME DIMENSION :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Time Dimension

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Time Dimension

Name: STORAGE

Type: STRING

Valid Values: 'RELATIONAL', 'AW'

Default: 'RELATIONAL'

The storage of a dimension can be AW or relational

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the analytical workspace name where the dimension is implemented

Name: AW_DIMENSION_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace dimension physical object name

Basic properties for TIME MAP :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Time Dimension Map

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Time Dimension Map

Basic properties for DIMENSION_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension_Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Properties for DIMENSION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS

Type: STRING

Valid Values: DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY, DEPLOY_TO_CATALOG_ONLY

Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for Dimension, they are DDL Scripts for Relational Dimensional or Scripts for ROLAP or or Scripts for AW.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: VIEW_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Name of the view that is generated to hide the control rows on the dimension implementation table of a star schema. If this field is left blank, the view name will default to '<Name of Dimension>_v'

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Dimension is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

Examples

```
OMBCREATE TIME_DIMENSION 'STAR1'
SET PROPERTIES ( DESCRIPTION, BUSINESS_NAME )
VALUES ('TimeSeries Star Description 1', 'TIME DIMENSION STAR1')
ADD NORMAL_CALENDAR HIERARCHY 'H1'
SET PROPERTIES ( DESCRIPTION, BUSINESS_NAME )
VALUES ('TimeSeries Hierarchy Description 1', 'TIME DIMENSION HIERARCHY
STAR1')
SET REF ( DAY LEVEL 'MYDAY' , CALENDAR_YEAR LEVEL 'MYCALYEAR' )
IMPLEMENTED BY STAR
ADD REF SEQUENCE 'STAR1'
POPULATE DATA FROM 2000 FOR 2 YEARS
```

```
OMBCREATE TIME_DIMENSION 'FYR2005'
SET FISCAL_CALENDAR PROPERTIES ( FISCAL_TYPE, FISCAL_CALENDAR_
START_YEAR,
FISCAL_CALENDAR_START_DAY_OF_WEEK ) VALUES ( '544', '01-01-2000',
'MONDAY'
)
ADD FISCAL_CALENDAR HIERARCHY 'FCALH1'
SET PROPERTIES ( DESCRIPTION, BUSINESS_NAME )
VALUES ('TimeSeries Hierarchy Description 1', 'TIME DIMENSION HIERARCHY
FSTAR1 FCALH1')
SET REF ( DAY LEVEL 'MY_DAY' , FISCAL_YEAR LEVEL 'MY_FISCAL_YEAR' )
IMPLEMENTED BY STAR
ADD REF SEQUENCE 'YR2005'
POPULATE DATA FROM 2000 FOR 2 YEARS.
```

and so on

See Also

OMBALTER TIME_DIMENSION, OMBDROP TIME_DIMENSION, OMBRETRIEVE TIME_DIMENSION

OMBCREATE TRANSPORTABLE_MODULE

Purpose

To create a transportable module.

Prerequisites

In the context of a project.

Syntax

```
createTMCommand = ( OMBCREATE TRANSPORTABLE_MODULE "QUOTED_STRING" {
    "setPropertiesAndLocationsAndIconSet" } )
setPropertiesAndLocationsAndIconSet = SET ( "setPropertiesClause" |
    "setSourceLocationClause" | "setTargetLocationClause" |
    "setReferenceIconSetClause" )
setPropertiesClause = ( PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")" )
setSourceLocationClause = SOURCE_LOCATION "QUOTED_STRING"
setTargetLocationClause = TARGET_LOCATION "QUOTED_STRING"
setReferenceIconSetClause = ( REF | REFERENCE ) ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { ", " "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createTMCommand

This command is for creating a transportable module.

QUOTED_STRING

The name of the transportable module to be created.

setPropertiesAndLocationsAndIconSet

Set properties for the transportable module, and/or specify source and target locations, and/or specify icon set for the newly create transportable module.

setPropertiesClause

Set properties for the transportable module.

Basic properties for TRANSPORTABLE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of the transportable module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description for the transportable module

Properties for TRANSPORTABLE_MODULE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: TARGET_OS_TYPE

Type: STRING

Valid Values: Linux, Unix, Windows

Default: Unix

The operating system type of the target machine. This is needed for generating shell scripts in correct style required by the operating system.

Name: TRANSPORT_TABLESPACE

Type: BOOLEAN

Valid Values: true, false

Default: true

Specifies whether transportable tablespace (TTS) feature is to be used for deploying tables in the transportable module. If set to true, tablespaces are copied from source to target using the server TTS mechanism. If set to false, tables are individually extracted and deployed using Oracle Data Pump available in Oracle Database 10g or later; but tablespaces are not transported. Since Oracle Data Pump is new in Oracle Database 10g, setting this parameter to false is only allowed if both source and target databases are with Oracle 10g or a higher versions.

Name: WHAT_TO_DEPLOY

Type: STRING

Valid Values: ALL_OBJECTS, TABLES_ONLY

Default: ALL_OBJECTS

Specifies whether only tables in the transportable module are deployed or everything in it is deployed.

Name: WORK_DIRECTORY

Type: STRING

Valid Values: N/A

Default: "

The full path of work directory on target machine, where temporary files, logs and tablespace datafiles may be stored. If left unspecified, OWB's runtime home directory is used as the work directory. It is highly recommended that users specify dedicated directory for transportable module deployment.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

setSourceLocationClause

Specify the source location name.

QUOTED_STRING

The name of an already created transportable module source location.

setTargetLocationClause

Specify the target location name.

QUOTED_STRING

The name of an already created transportable module target location.

setReferenceIconSetClause

Set the icon set for the new transportable module.

QUOTED_STRING

The name of the icon set.

propertyNameList

The list of unquoted property names.

propertyValueList

The list of property values.

propertyValue

A property value can be a single-quoted string, an integer, or a floating point number.

Examples

```
OMBCREATE TRANSPORTABLE_MODULE 'TM101'
SET PROPERTIES (WORK_DIRECTORY, TARGET_OS_TYPE, WHAT_TO_DEPLOY,
TRANSPORT_TABLESPACE)
VALUES ('/private/tgfiledir', 'Unix', 'ALL_OBJECTS', 'false')
SET SOURCE_LOCATION 'TM_SRC_LOC'
SET TARGET_LOCATION 'TM_TGT_LOC'
```

This command will create a transportable module named 'TM101' in the current project. The transportable module will be used to transport objects from source location TM_SRC_LOC to target location TM_TGT_LOC.

Once created using OMBCREATE TRANSPORTABLE_MODULE command, the transportable module is only an empty container. Use OMBIMPORT ORACLE_DATABASE command to specify what source objects should be included in it. Then use OMBDEPLOY command to deploy the transportable module to target. It is only at the deployment time when data and metadata movement actually happen.

See Also

OMBCREATE, OMBALTER TRANSPORTABLE_MODULE, OMBRETRIEVE TRANSPORTABLE_MODULE, OMBDROP TRANSPORTABLE_MODULE, OMBCREATE TRANSIENT_IMPORT_ACTION_PLAN, OMBIMPORT ORACLE_DATABASE, OMBDEPLOY

OMBCREATE VARYING_ARRAY

Purpose

To create an Varying Array (or Varray)

Prerequisites

Should be in the context of an Oracle Module

Syntax

```
createVaryingArrayCommand = OMBCREATE ( VARYING_ARRAY "QUOTED_STRING" [
    SET ( "setPropertiesClause" [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

createVaryingArrayCommand

Creates a Varying Array with the given name.

setPropertiesClause

Sets properties (core, logical, physical, user-defined) for Varying Array .

Valid properties are as shown:

Basic properties for VARYING_ARRAY:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Varying Array

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Varying Array

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Base Element of the Varying Array

Properties for VARYING_ARRAY:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

propertyValueList

The list of property values.

propertyValue

This clause adds the property values.

Examples

```
OMBCREATE VARYING_ARRAY 'NEW_VARRAY' SET PROPERTIES  
(DATATYPE,ARRAY_LENGTH)
```

```
VALUES ('NUMBER',10)
```

This will create a Varray named 'NEW_VARRAY' with its base element type as 'NUMBER' and array length (size) 10.

See Also

OMBCREATE, OMBALTER VARYING_ARRAY, OMBDROP VARYING_ARRAY

OMBCREATE VIEW

Purpose

To create a view.

Prerequisites

In the context of an Oracle Module.

Syntax

```
createViewCommand = OMBCREATE ( VIEW "QUOTED_STRING" [ SET (
    "setPropertiesClause" [ SET ( REF | REFERENCE )
    "setReferenceIconSetClause" ] | ( REF | REFERENCE )
    "setReferenceIconSetClause" ) ] [ "addViewSCOandDependentClause" ] )
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
setReferenceIconSetClause = ICONSET "QUOTED_STRING"
addViewSCOandDependentClause = ADD ( "addColumnClause" |
    "addViewConstraintClause" | "addRelationalDependentClause" ) [
    "addViewSCOandDependentClause" ]
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
addColumnClause = COLUMN "QUOTED_STRING" [ AT POSITION "INTEGER_LITERAL" ]
    [ SET "setPropertiesClause" ]
addViewConstraintClause = "addUkPkClause" | "addFkClause"
addRelationalDependentClause = ( REFERENCE | REF ) ( TABLE | VIEW |
    MATERIALIZED_VIEW ) "QUOTED_STRING"
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
addUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" [ SET
    "setUkPkPropertiesAndReferencesColumnsClauses" ]
addFkClause = FOREIGN_KEY "QUOTED_STRING" [ SET "setFkSubClauses" ]
setUkPkPropertiesAndReferencesColumnsClauses = "setPropertiesClause" [ SET
    ( REF | REFERENCE ) "constraintColumnReferencesClause" ] | ( REF |
    REFERENCE ) "constraintColumnReferencesClause"
setFkSubClauses = "setPropertiesClause" [ SET ( REF | REFERENCE )
    "setFkReferencesClauses" ] | ( REF | REFERENCE )
    "setFkReferencesClauses"
constraintColumnReferencesClause = COLUMNS "(" "quotedNameList" ")"
setFkReferencesClauses = "constraintColumnReferencesClause" [ SET ( REF |
    REFERENCE ) "constraintUkReferencesClause" ] |
    "constraintUkReferencesClause" [ SET ( REF | REFERENCE )
    "constraintColumnReferencesClause" ]
quotedNameList = "QUOTED_STRING" { "," "QUOTED_STRING" }
constraintUkReferencesClause = ( UNIQUE_KEY | PRIMARY_KEY )
    "QUOTED_STRING" [ OF ( TABLE | VIEW ) "QUOTED_STRING" ]
```

Keywords And Parameters

`createViewCommand`

This command creates a view.

`QUOTED_STRING`

Specify the name of the view to be created.

setPropertyClause

Used to set properties (core, logical, physical, user-defined) for views and their columns, unique keys, foreign keys, and primary keys.

Note:

Constraints can be specified but will not be generated for either View or Materialized View in this release.

Basic properties for VIEW, MATERIALIZED_VIEW:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the View, MaterializedView

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the View, MaterializedView

Name: VIEW_QUERY

Type: STRING(4000)

Valid Values: N/A

Default: "

Sets the query definition in View and MaterializedView.

Properties for VIEW:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the

constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

addViewSCOandDependentClause

This clause adds components like columns and constraints, as well as dependencies to some other relational objects.

propertyNameList

The list of properties.

propertyValueList

The list of property values.

addColumnClause

This clause adds a column.

When you create a table or alter a table to add a set of columns, the position that you specify for a column must be either less than or equal to the number of columns that you have listed up to that point in the command.

For example, the following OMBCREATE command does not add the specified columns to the table:

```
OMBCREATE TABLE 'MY_TABLE' \  
SET PROPERTIES (DESCRIPTION) VALUES ('TRIAL TABLE') \  
ADD COLUMN 'C1' \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \  
ADD COLUMN 'C2' AT POSITION 3 \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \  
ADD COLUMN 'C3' AT POSITION 2 \  
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2')
```

This is because at the point when you specify the position of the column C2 as 3, you have added just two columns to the table. But the following OMBALTER command adds the specified columns to the table. This is because at the point when you specify the position of the column C2 as 2, you are adding the second column to the table.

```
OMBCREATE TABLE 'MY_TABLE' \  
SET PROPERTIES (DESCRIPTION) VALUES ('TRIAL TABLE') \  
ADD COLUMN 'C1' \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10) \  
ADD COLUMN 'C2' AT POSITION 2 \  
SET PROPERTIES (DATATYPE) VALUES ('VARCHAR2') \  
ADD COLUMN 'C3' AT POSITION 1 \  
SET PROPERTIES (DATATYPE, PRECISION) VALUES ('NUMBER',10)
```

In the preceding example, the order in which the columns are added are as follows:

C1

C1, C2

C3, C1, C2

`addViewConstraintClause`

This clause adds the view's configuration clause.

`addRelationalDependentClause`

This clause adds referential dependencies to other relational objects.

`propertyValue`

This clause adds the property values.

`addUkPkClause`

This clause adds the adds unique key and primary keys.

QUOTED_STRING

name of the unique key or primary key.

`addFkClause`

This clause adds foreign key.

QUOTED_STRING

Name of the foreign key.

`setUkPkPropertiesAndReferencesColumnsClauses`

This clause adds properties and references to columns.

`setFkSubClauses`

This clause set references to a foreign key.

`constraintColumnReferencesClause`

This clause provides names of all columns.

`setFkReferencesClauses`

This clause sets foreign key references.

quotedNameList

This clause gives column names.

constraintUkReferencesClause

The first QUOTED_STRING denotes the UniqueKey or Primary key name, and the latter denotes the table's or view's name.

Examples

```
OMBCREATE VIEW 'NEW_VIEW' SET PROPERTIES (DESCRIPTION, BUSINESS_
NAME)
```

```
VALUES ('this is a new view', 'New View')
```

This will create a view named "NEW_VIEW", its description is "this is a new view", and business name is "New View".

See Also

OMBCREATE, OMBALTER VIEW, OMBDROP VIEW, OMBRETRIEVE VIEW

11

OMBRETRIEVE to OMBRETRIEVE LOCATION

This chapter lists commands associated with OMBRETRIEVE in alphabetical order, concluding with the command OMBRETRIEVE LOCATION. Subsequent commands associated with OMBRETRIEVE are contained in the next chapter.

OMBRETRIEVE

Purpose

Retrieve metadata of a component.

Prerequisites

Can be in any context. Component to retrieve can be specified by either absolute or relative path.

Syntax

```
retrieveCommand = OMBRETRIEVE "fco_type" "fco_name" { "sco_type"  
    "sco_name" } ( "getPropertiesClause" | "getSCOClauses" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getSCOClauses = GET [ ( REF | REFERENCE ) [ "qualifier" ] ] "sco_type" [ AT  
    POSITION "pos" ]  
propertyNameList = "propertyName" { "," "propertyName" }  
qualifier = "UNQUOTED_STRING"
```

Keywords And Parameters

retrieveCommand

Specify the component from which to retrieve metadata.

fco_type

The type of the component.

fco_name

The physical name of the component in single quotes.

getPropertiesClause

Retrieve properties of an object.

getSCOClauses

Retrieve a list of child object names of a given type.

qualifier

Specify which reference to set, if there are more than one pointing to the same type.

propertyNameList

A list of property names.

propertyName

An unquoted string representing the name of a property.

Examples

This is an example for retrieving the description a table:

```
OMBRETRIEVE TABLE 'T1' GET PROPERTIES (DESCRIPTION)
```

The following statement retrieves the datatype and length of a view column:

```
OMBRETRIEVE VIEW 'V1' COLUMN 'COL1'  
GET PROPERTIES (DATATYPE, LENGTH)
```

See Also

OMBCREATE, OMBALTER

OMBRETRIEVE ACTIVITY_TEMPLATE

Purpose

To retrieve the details of an activity template.

Prerequisites

Should be in the context of an Activity Template Folder.

Syntax

```
parseRetrieveTemplateCommand = OMBRETRIEVE ACTIVITY_TEMPLATE
    "QUOTED_STRING" ( "getParameterPropertiesClause" |
    "getTemplatePropertiesClause" )
getParameterPropertiesClause = PARAMETER "QUOTED_STRING" GET PROPERTIES
    "propertyKeyList"
getTemplatePropertiesClause = ( GET ( ( PROPERTIES "propertyKeyList" ) |
    PARAMETERS | ( ( REFERENCE | REF ) [ "getReferenceIconSetClause" ] ) ) )
)
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ICONSET
propertyKey = "UNQUOTED_STRING"
```

Examples

retrieveActivityTemplateCommandExampleTag??

See Also

OMBDROP, OMBCREATE ACTIVITY_TEMPLATE, OMBALTER ACTIVITY_TEMPLATE

OMBRETRIEVE ACTIVITY_TEMPLATE_FOLDER

Purpose

To retrieve the details of an activity template folder.

Prerequisites

Should be in the context of a Project.

Syntax

```
parseRetrieveTemplateFolderCommand = OMBRETRIEVE ACTIVITY_TEMPLATE_FOLDER
    "QUOTED_STRING" "getTemplateFolderPropertiesClause"
getTemplateFolderPropertiesClause = ( GET ( ( PROPERTIES "propertyKeyList"
    ) | ( ( REFERENCE | REF ) "getReferenceIconSetClause" ) ) )
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ICONSET
propertyKey = "UNQUOTED_STRING"
```

Examples

```
OMBRETRIEVE ACTIVITY_TEMPLATE 'FOLDR1' GET PROPERTIES (BUSINESS_
NAME)
```

See Also

OMBDROP, OMBCREATE ACTIVITY_TEMPLATE_FOLDER, OMBALTER
ACTIVITY_TEMPLATE_FOLDER

OMBRETRIEVE ADVANCED_QUEUE

Purpose

Retrieve details of the Advanced Queue.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
retrieveAQCommand = OMBRETRIEVE ADVANCED_QUEUE "QUOTED_STRING" GET (
    "getPropertiesClause" | "getReferenceIconSetClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveAQCommand

Retrieves the details of the Advanced Queue with the given name.

getPropertiesClause

Retrieves the values of the given Properties for the Advanced Queue with the given name.

Basic properties for ADVANCED_QUEUE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Advanced Queue

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Advanced Queue

Name: QTABLE

Type: STRING(4000)

Valid Values: N/A

Default: "

Queue Table for the Advanced Queue. This has to be the name of a Queue Table(QUEUE_TABLE) existing in the same Oracle Module.

Properties for ADVANCED_QUEUE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DEQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Dequeue Enabled for AQ

Name: ENQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enqueue enabled for AQ

Name: GENERATE_ADVANCED_QUEUE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the Advanced Queue.

Name: GENERATE_TEMPORARY_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the temporary table.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MAX_RETRIES

Type: NUMBER

Valid Values: N/A

Default: 5

Max. number of Retries

Name: RETENTION_TIME

Type: NUMBER

Valid Values: N/A

Default: 0

Message retention time

Name: RETRY_DELAY

Type: NUMBER

Valid Values: N/A

Default: 0

Delay period before retry

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

getReferenceIconSetClause

Get the Icon Set.

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE ADVANCED_QUEUE 'SOME_ADVANCED_QUEUE' GET
PROPERTIES
```


(MAX_RETRIES, RETRY_DELAY, RETENTION_TIME, ENQUEUE_ENABLED,
DEQUEUE_ENABLED, QTABLE)

This will retrieve the Advanced Queue "SOME_ADVANCED_QUEUE"'s properties.

See Also

OMBRETRIEVE, OMBALTER ADVANCED_QUEUE, OMBCREATE ADVANCED_
QUEUE, OMBDROP ADVANCED_QUEUE

OMBRETRIEVE ALTERNATIVE_SORT_ORDER

Purpose

Retrieve details of an Alternative Sort Order.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveAlternativeSortOrderCommand = OMBRETRIEVE ALTERNATIVE_SORT_ORDER
    "QUOTED_STRING" "retrieveAlternativeSortOrderClauseDetails"
retrieveAlternativeSortOrderClauseDetails = GET (
    "getPropertiesClauseforLOVandD2D" | "getReferenceIconSetClause" | (
        REF | REFERENCE ) DEFINING ITEM | ( REF | REFERENCE ) ORDERED ITEM | (
            REF | REFERENCE ) ITEMS | DEPENDENTS )
getPropertiesClauseforLOVandD2D = PROPERTIES "(
    "propertyNameListforLOVandD2D" )"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyNameListforLOVandD2D = ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) {
    ", " ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) }
```

Keywords And Parameters

`retrieveAlternativeSortOrderCommand`

Retrieves the alternative sort order.

`QUOTED_STRING`

name of the alternative sort order.

`retrieveAlternativeSortOrderClauseDetails`

This clause retrieves the contents of an alternative sort order.

`GET`

This clause retrieves the following

`REF DEFINING ITEM` retrieves the Item that holds the individual values for this alternative sort order.

`REF ORDERED ITEM` retrieves the Item that holds the order of the values for this alternative sort order.

`REF ITEMS` retrieves the list of Items that use this alternative sort order.

`DEPENDENTS` retrieves a list of Item Folders that the alternative sort order depends on.

(This will return the Item Folder containing any Values and Order Items).

getPropertiesClauseforLOVandD2D

This clause gets the properties of the object.

Basic properties for ALTERNATIVE_SORT_ORDER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the alternative sort order

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the alternative sort order

Name: DRILL_TO_DETAIL

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the alternative sort order enables drilling between the item folders containing the items that use the alternative sort order

Name: RETRIEVE_VALUES_GROUP_SIZE

Type: Number

Valid Values: N/A

Default: '100'

The number of rows to be fetched from the database at a time

Name: CACHE_VALUES

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the list of values should be cached in memory

Name: REQUIRE_SEARCH

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether search criteria should be requested

Name: SHOW_IN_NAVIGATOR

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be shown in the item navigator

Name: SORTED_DISTINCT

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be displayed sorted with duplicates hidden

Properties for ALTERNATIVE_SORT_ORDER:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.

2. " " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

propertyNameListforLOVandD2D

This is the list of property names.

Examples

OMBRETRIEVE ALTERNATIVE_SORT_ORDER 'COLORS' GET
PROPERTIES(DESCRIPTION)

See Also

OMBALTER ALTERNATIVE_SORT_ORDER, OMBCREATE ALTERNATIVE_SORT_
ORDER

OMBRETRIEVE ANALYZE_ACTION_PLAN

Purpose

To retrieve a profile action plan.

Prerequisites

An action plan for profiling must already exist.

Syntax

```
RetrieveActionPlanCommand = ( OMBRETRIEVE ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ) ( "getActionsClause"
    | "retrieveActionClause" )
getActionsClause = GET ACTIONS
retrieveActionClause = ACTION "QUOTED_STRING" ( GET (
    "getPropertiesClause" | "getReferenceClause" ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceClause = ( REF | REFERENCE )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

RetrieveActionPlanCommand

Retrieve a profile action plan

getActionsClause

Get a list of actions from an action plan.

retrieveActionClause

Retrieve a set of properties or the associated object of an action.

QUOTED_STRING

Action name

getPropertiesClause

Retrieve a set of properties that is associated with an action.

getReferenceClause

Retrieve the object associated with an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

OMBRETRIEVE ANALYZE_ACTION_PLAN 'ANALYZE_PLAN' GET ACTIONS

See Also

OMBCREATE ANALYZE_ACTION_PLAN, OMBPROFILE

OMBRETRIEVE BUSINESS_AREA

Purpose

Retrieve details of a Business Area.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveBusinessAreaCommand = OMBRETRIEVE BUSINESS_AREA "QUOTED_STRING"  
    "retrieveBusinessAreaClause"  
retrieveBusinessAreaClause = GET ( "getPropertiesClause" |  
    "getReferenceIconSetClause" | "getBASCOClause" )  
getPropertiesClause = PROPERTIES "( "propertyNameList" )"  
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET  
getBASCOClause = ITEM_FOLDERS  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveBusinessAreaCommand

To retrieve a business area.

QUOTED_STRING

name of the business area.

retrieveBusinessAreaClause

Retrieves the contents of the business area.

getPropertiesClause

Retrieves the properties of the object.

Basic properties for BUSINESS_AREA:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the business area

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the business area

Properties for BUSINESS_AREA:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

getBASCOClause

Retrieves the shortcuts from the business area.

propertyNameList

This is the list of property names.

Examples

```
OMBRETRIEVE BUSINESS_AREA 'SALES' GET PROPERTIES(DESCRIPTION)
```

See Also

OMBALTER BUSINESS_AREA, OMBCREATE BUSINESS_AREA

OMBRETRIEVE BUSINESS_DEFINITION_MODULE

Purpose

Retrieve details of the business definition module.

Prerequisites

Should be in the context of a project or use the full path.

Syntax

```
retrieveEULModuleCommand = OMBRETRIEVE BUSINESS_DEFINITION_MODULE
    "QUOTED_STRING" ( "getPropertiesClause" | "getReferenceLocationClause"
    | "getReferenceDefaultLocationClause" | "getReferenceIconSetClause" |
    "getReferenceLocationsClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT
    LOCATION
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveEULModuleCommand

This command retrieves the details of a business definition module.

QUOTED_STRING

Name of the existing business definition module or path to the business definition module.

getPropertiesClause

Retrieve a set of properties associated with a business definition module.

Basic properties for BUSINESS_DEFINITION_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a business definition module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a business definition module

Properties for BUSINESS_DEFINITION_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Discoverer Location for Business Definition Module

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Name: MLS_DEPLOYMENT_LANGUAGE

Type: STRING

Valid Values: N/A

Default: MLS_BASE_LANGUAGE

MLS Language to be used for deployment

Name: OBJECT_MATCHING

Type: STRING

Valid Values: BY_IDENTIFIER, BY_NAME

Default: BY_IDENTIFIER

Whether import should match up objects by identifier or by name

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceLocationClause

Retrieve the location set for the business definition module.

getReferenceDefaultLocationClause

Retrieve the default runtime location referenced by this business definition module.

getReferenceIconSetClause

Retrieve the icon set referenced by this business definition module.

getReferenceLocationsClause

Retrieve the runtime locations referenced by this business definition module.

propertyNameList

Comma-delimited list of property names. Property names are not in quotes.

Examples

```
OMBRETRIEVE BUSINESS_DEFINITION_MODULE 'src_module' GET PROPERTIES  
(DESCRIPTION, BUSINESS_NAME)
```

This will retrieve the business definition module "src_module"s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE BUSINESS_PRESENTATION_MODULE

Purpose

Retrieve details of the presentation module.

Prerequisites

Should be in the context of a project or use the full path.

Syntax

```
retrieveReportModuleCommand = OMBRETRIEVE BUSINESS_PRESENTATION_MODULE
    "QUOTED_STRING" ( "getPropertiesClause" | "getReferenceLocationClause"
    | "getReferenceDefaultLocationClause" | "getReferenceIconSetClause" |
    "getReferenceLocationsClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT
    LOCATION
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveReportModuleCommand

This command retrieves the details of a presentation module.

QUOTED_STRING

Name of the existing presentation module or path to the presentation module.

getPropertiesClause

Retrieve a set of properties associated with a presentation module.

Basic properties for BUSINESS_PRESENTATION_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a presentation module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a presentation module

Properties for BUSINESS_PRESENTATION_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

BI Beans Location for Business Presentation Module

Name: DEFAULT_CATALOG_FOLDER

Type: STRING

Valid Values: N/A

Default: "

Default Catalog Folder for deployed BI Beans presentations

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceLocationClause

Retrieve the location set for the presentation module.

getReferenceDefaultLocationClause

Retrieve the default runtime location referenced by this business

presentation module.

getReferenceIconSetClause

Retrieve the icon set referenced by this business presentation module.

getReferenceLocationsClause

Retrieve the runtime locations referenced by this business presentation module.

propertyNameList

Comma-delimited list of property names. Property names are not in quotes.

Examples

```
OMBRETRIEVE BUSINESS_PRESENTATION_MODULE 'salesrep_module' GET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME)
```

This will retrieve the presentation module "salesrep_module"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE CALENDAR

Purpose

Retrieve details of a Calendar.

Prerequisites

Should be in the context of a CalendarModule

Syntax

```
retrieveCalendarClause = OMBRETRIEVE CALENDAR "QUOTED_STRING" ( ( GET (
    "getCalendarPropertiesClause" | "getReferenceIconSetClause" ) ) |
    "getSchedulePropertiesClause" )
getCalendarPropertiesClause = ( PROPERTIES "propertyKeyList" ) | SCHEDULES
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getSchedulePropertiesClause = SCHEDULE "QUOTED_STRING" GET PROPERTIES
    "propertyKeyList"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyKey = "UNQUOTED_STRING"
```

Examples

```
OMBRETRIEVE CALENDAR 'CAL1' GET PROPERTIES (BUSINESS_NAME,
DESCRIPTION)
```

```
OMBRETRIEVE CALENDAR 'CAL1' SCHEDULE 'LOCALWINDOW' GET
PROPERTIES
(TIMEZONE)
```

```
OMBRETRIEVE CALENDAR 'CAL1' SCHEDULE 'LOCALWINDOW' GET
PROPERTIES
(PREVIEW_DATES)
```

See Also

OMBDROP, OMBCREATE CALENDAR_MODULE, OMBALTER CALENDAR_MODULE

OMBRETRIEVE CALENDAR_MODULE

Purpose

Retrieve details of a Calendar Module.

Prerequisites

Should be in the context of a Project

Syntax

```
retrieveCalendarClause = OMBRETRIEVE CALENDAR "QUOTED_STRING" ( ( GET (
    "getCalendarPropertiesClause" | "getReferenceIconSetClause" ) ) |
    "getSchedulePropertiesClause" )
getCalendarPropertiesClause = ( PROPERTIES "propertyKeyList" ) | SCHEDULES
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getSchedulePropertiesClause = SCHEDULE "QUOTED_STRING" GET PROPERTIES
    "propertyKeyList"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
propertyKey = "UNQUOTED_STRING"
```

Examples

```
OMBRETRIEVE CALENDAR_MODULE 'CALMOD1' GET PROPERTIES
(BUSINESS_NAME,
DESCRIPTION)
```

See Also

OMBDROP, OMBCREATE CALENDAR_MODULE, OMBALTER CALENDAR_MODULE

OMBRETRIEVE CHANGE_DATA_CAPTURE

Purpose

This command is used to retrieve details about the change data capture.

Prerequisites

This command can only be executed in the context of a change data capture and operates only on already existing change data captures.

Syntax

```
retrieveChangeSetCommand = OMBRETRIEVE ( CHANGE_DATA_CAPTURE
    "QUOTED_STRING" ) GET ( CHANGESET_TYPE | PROPERTIES "("
    "propertyNameList" ")" | CAPTURE ( OBJECTS | TABLES | VIEWS |
    MATERIALIZED_VIEWS ) | CAPTURE ( TABLE | VIEW | MATERIALIZED_VIEW )
    "QUOTED_STRING" "getCaptureObjectDetailsClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
getCaptureObjectDetailsClause = ( "getCaptureColumnsClause" |
    "getChangeSpecClause" | "getRowIdentifierClause" |
    "getTxnIdentifierClause" | "getSpecialCaptureColumnsClause" )
getCaptureColumnsClause = COLUMNS
getChangeSpecClause = ( CAPTURED OPERATIONS ) | ( INSERT CHANGE_COLUMN ) |
    ( UPDATE CHANGE_COLUMN ) | ( DELETE CHANGE_COLUMN ) | ( INSERT
    CHANGE_EXPRESSION ) | ( UPDATE CHANGE_EXPRESSION ) | ( DELETE
    CHANGE_EXPRESSION )
getRowIdentifierClause = ROW_IDENTIFIER_COLUMNS
getTxnIdentifierClause = TXN_IDENTIFIER_COLUMNS
getSpecialCaptureColumnsClause = CAPTURE ( OLD_VALUES | USER_NAME )
```

Examples

```
OMBRETRIEVE CHANGE_DATA_CAPTURE 'EMPLOYEE_CHANGES' GET
CAPTURE TABLES
```

This retrieves a list of the tables whose changes are captured by this change data capture.

See Also

OMBRETRIEVE

OMBRETRIEVE CMI_DEFINITION

Purpose

Retrieve details of the CMI definition.

Prerequisites

Should be in the root context.

Syntax

```
retrieveMIVDefinitionCommand = OMBRETRIEVE CMI_DEFINITION "QUOTED_STRING"
    "getPropertiesClause"
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveMIVDefinitionCommand

This command retrieves the details of an CMI Definition

QUOTED_STRING

Name of the existing CMI definition or path to the CMI definition.

getPropertiesClause

Retrieve a set of properties that is associated with an CMI Definition.

Properties for CMI_DEFINITION:

Name: LOCATION_UOID

Type: STRING

Valid Values: N/A

Default: "

Location Warehouse Builder should use to retrieve the data.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE CMI_DEFINITION 'src_definition' GET PROPERTIES  
(DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the CMI definition "src_definition"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE CMI_MODULE

Purpose

Retrieve details of the CMI module.

Prerequisites

Should be in the context of project.

Syntax

```
retrieveMIVModuleCommand = OMBRETRIEVE CMI_MODULE "QUOTED_STRING" (
    "getPropertiesClause" | "getReferenceLocationClause" |
    "getReferenceDefaultLocationClause" |
    "getReferenceMetadataLocationClause" | "getReferenceIconSetClause" |
    "getReferenceLocationsClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT
    LOCATION
getReferenceMetadataLocationClause = GET ( REF | REFERENCE )
    METADATA_LOCATION
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveMIVModuleCommand

This command retrieves the details of an CMI Module

QUOTED_STRING

Name of the existing CMI module or path to the CMI module.

getPropertiesClause

Retrieve a set of properties that is associated with an CMI Module.

Basic properties for CMI_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of an CMI Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of an CMI Module

Properties for CMI_MODULE:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

If this is a source module, this value indicates the location from which data will be read. If this is a target warehouse module, this value indicates the location where generated code will be deployed to and/or where data will be written to.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceLocationClause

Retrieve the name of the runtime location referenced by this CMI module.

getReferenceDefaultLocationClause

Retrieve the default runtime location referenced by this CMI module.

getReferenceMetadataLocationClause

Retrieve the metadata location referenced by this CMI module.

getReferenceIconSetClause

Retrieve the icon set referenced by this CMI module.

getReferenceLocationsClause

Retrieve the runtime locations referenced by this CMI module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE CMI_MODULE 'src_module' GET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the CMI module "src_module"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE COLLECTION

Purpose

Retrieve details of the collection, including its shortcuts.

Prerequisites

Should be in the context of a project, before retrieving a collection.

Syntax

```
retrieveCollectionCommand = OMBRETRIEVE COLLECTION "QUOTED_STRING" ( GET (
  ( "referencesClause" | "propertiesClause" ) ) |
  "getReferenceIconSetClause" )+
referencesClause = ( ALL | TABLE | VIEW | PACKAGE | DIMENSION |
  MATERIALIZED_VIEW | SEQUENCE | CUBE | ADVANCED_QUEUE | STREAMS_QUEUE |
  QUEUE_TABLE | EXTERNAL_TABLE | VARYING_ARRAY | OBJECT_TYPE |
  NESTED_TABLE | MAPPING | COLLECTION | ORACLE_MODULE | PROCESS_FLOW |
  PROCESS_FLOW_PACKAGE | PROCESS_FLOW_MODULE | SAP_MODULE | CMI_MODULE |
  FUNCTION | PROCEDURE | FLAT_FILE_MODULE | FLAT_FILE |
  BUSINESS_DEFINITION_MODULE | BUSINESS_PRESENTATION_MODULE |
  PRESENTATION_TEMPLATE | ITEM_FOLDER | LIST_OF_VALUES | DRILL_TO_DETAIL
  | ALTERNATIVE_SORT_ORDER | DRILL_PATH | BUSINESS_AREA |
  GATEWAY_MODULE | CONFIGURATION | REGISTERED_FUNCTION |
  PLUGGABLE_MAPPING | PLUGGABLE_MAPPING_FOLDER | DATA_AUDITOR |
  TRANSPORTABLE_MODULE | EXPERT_MODULE | EXPERT | CALENDAR_MODULE |
  CALENDAR | DATA_PROFILE | DATA_RULE_MODULE | DATA_RULE ) REFERENCES
propertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveCollectionCommand

Retrieve details regarding a collection of objects.

referencesClause

Specify the type of references to retrieve from the collection.

propertiesClause

Retrieve values for a number of properties.

Basic properties for COLLECTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the collection

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the collection

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

Comma separated list of property names to retrieve values. Property names are unquoted.

Examples

OMBRETRIEVE COLLECTION 'Purchasing Warehouse' GET TABLE REFERENCES

See Also

OMBLIST

OMBRETRIEVE CONFIGURATION

Purpose

Retrieve details of the Configuration.

Prerequisites

Should be in the context of a project.

Syntax

```
retrieveConfigurationCommand = OMBRETRIEVE CONFIGURATION "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferenceIconSetClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveConfigurationCommand

This command retrieves the details of a Configuration.

getPropertiesClause

Retrieve a set of properties that is associated with a Configuration.

Basic properties for CONFIGURATION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Configuration

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Configuration.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceIconSetClause

Retrieve the Icon Set associated with a Configuration.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE CONFIGURATION 'QA_CONFIGURATION' GET PROPERTIES  
(DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the Configuration description and business name.

See Also

OMBLIST

OMBRETRIEVE CONNECTOR

Purpose

Retrieve details from a connector.

Prerequisites

Can be in any context; the name is a name of the connector's owning location and a connector name separated by slash.

Syntax

```
retrieveConnectorCommand = OMBRETRIEVE CONNECTOR "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferencedLocationClause" |  
    "getReferenceIconSetClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferencedLocationClause = GET ( REF | REFERENCE ) LOCATION  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveConnectorCommand

Retrieve details from the named connector.

getPropertiesClause

Get properties of the connector.

getReferencedLocationClause

Get the name of the location which the connector references.

getReferenceIconSetClause

Get the referenced icon set

propertyNameList

The names of the properties whose values you want to set.

Properties for CONNECTOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the connector.

Name: DESCRIPTION
 Type: STRING(4000)
 Valid Values: N/A
 Default: "
 Description of the connector.

Name: DATABASE_LINK_NAME
 Type: STRING(30)
 Valid Values: N/A
 Default: "
 Database Link name.

Properties for CONNECTOR:

Name: DEPLOYABLE
 Type: BOOLEAN
 Valid Values: true, false
 Default: true
 Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBRETRIEVE CONNECTOR 'A_LOCATION/A_CONNECTOR' GET PROPERTIES
(DESCRIPTION,
BUSINESS_NAME)
```

This will retrieve the connector "A_CONNECTOR"'s description, and a business name.

```
OMBRETRIEVE CONNECTOR 'A_LOCATION/A_CONNECTOR' GET REF
LOCATION
```

This will retrieve the A_CONNECTOR referenced location.

See Also

OMBRETRIEVE

OMBRETRIEVE CONTROL_CENTER

Purpose

Retrieve details from a control center.

Prerequisites

Can be in any context.

Syntax

```
retrieveControlCenterCommand = OMBRETRIEVE CONTROL_CENTER "QUOTED_STRING"
    ( "retrieveControlCenterClause" | "getReferenceIconSetClause" |
      "retrieveReferenceLocationClause" )
retrieveControlCenterClause = "getPropertiesClause" | GET
    "getReferenceLocationsClause"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
retrieveReferenceLocationClause = GET ( REFERENCE | REF ) LOCATION
    "QUOTED_STRING" "getPropertiesClause"
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceLocationsClause = ( REFERENCE | REF ) LOCATIONS
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveControlCenterCommand

Retrieve details from a control center.

getReferenceIconSetClause

Retrieve the icon set from a control center.

retrieveReferenceLocationClause

Get the specified properties of the referenced location.

getPropertiesClause

Get the specified properties of the control center.

getReferenceLocationsClause

Get the list of all referenced locations of this control center.

propertyNameList

The names of the properties whose values you want to retrieve.

Properties for CONTROL_CENTER:

Name: HOST

Type: STRING

Valid Values: N/A

Default: N/A

The host machine the control center is installed on.

Name: PORT

Type: NUMBER

Valid Values: 1 - 65535

Default: 1521

The port number of the database in which the control center is installed.

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The service name of the database in which the control center is installed.

Name: USER

Synonym: USER_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The name of the database user you wish to connect to the control center as.

Name: PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: N/A

The name of the schema in which the control center is installed.

All of the preceding properties (except of PASSWORD) are mandatory for OMBCREATE CONTROL_CENTER.

Basic properties for CONTROL_CENTER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the control center.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the control center.

Properties for a referenced location of the control center:

Name: IS_SOURCE

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, then a refernced location is a source location.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, then a refernced location is a target location.

Properties for CONTROL_CENTER:

Name: HOST

Type: STRING

Valid Values: N/A

Default: "

Host of the location

Name: NET_SERVICE_NAME

Type: STRING(2000)

Valid Values: N/A

Default: "

Net Service Name of the location

Name: PASSWORD

Type: STRING(30)

Valid Values: N/A

Default: "

Password for the location

Name: PORT

Type: NUMBER

Valid Values: N/A

Default: 1521

Port of the location

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: "

Schema name for the location

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: "

Service Name of the location

Name: USER

Type: STRING

Valid Values: N/A

Default: "

User name for the location

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

Examples

```
OMBRETRIEVE CONTROL_CENTER 'MY_CONNECTION' GET PROPERTIES  
(DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the control center "MY_CONNECTION"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE CORRECTION_MAPS_ACTION_PLAN

Purpose

To retrieve properties of a correction map action plan.

Prerequisites

In the context of a data profile.

Syntax

```
RetrieveActionPlanCommand = ( OMBRETRIEVE ( ( DEPLOYMENT_ACTION_PLAN |  
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |  
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ) ( "getActionsClause"  
    | "retrieveActionClause" )  
getActionsClause = GET ACTIONS  
retrieveActionClause = ACTION "QUOTED_STRING" ( GET (  
    "getPropertiesClause" | "getReferenceClause" ) )  
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"  
getReferenceClause = ( REF | REFERENCE )  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

RetrieveActionPlanCommand

Retrieve correction maps action plan.

QUOTED_STRING

Action plan name.

getActionsClause

Get a list of actions from an action plan.

retrieveActionClause

Retrieve a set of properties or the associated object of an action.

getPropertiesClause

Retrieve a set of properties that is associated with an action.

getReferenceClause

Retrieve the object associated with an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE CORRECTION_MAPS_ACTION_PLAN 'CORRECT_INV_LOC_MAP' GET ACTIONS
```

```
OMBRETRIEVE CORRECTION_MAPS_ACTION_PLAN 'CORRECT_INV_LOC_MAP' ACTION
```

```
'UK_LOC_U' GET REFERENCE
```

```
OMBRETRIEVE CORRECTION_MAPS_ACTION_PLAN 'CORRECT_INV_LOC_MAP' ACTION
```

```
'UK_LOC_U' GET PROPERTIES (DATA_RULE_USAGE_NAME, ERROR_HANDLING_STRATEGY,
```

```
CORRECTION_STRATEGY)
```

See Also

```
OMBCREATE CORRECTION_MAPS_ACTION_PLAN, OMBPROFILE
```

OMBRETRIEVE CORRECTION_SCHEMA_ACTION_PLAN

Purpose

To retrieve properties of a correction schema action plan.

Prerequisites

In the context of a data profile.

Syntax

```
RetrieveActionPlanCommand = ( OMBRETRIEVE ( ( DEPLOYMENT_ACTION_PLAN |  
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |  
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ) ( "getActionsClause"  
    | "retrieveActionClause" )  
getActionsClause = GET ACTIONS  
retrieveActionClause = ACTION "QUOTED_STRING" ( GET (  
    "getPropertiesClause" | "getReferenceClause" ) )  
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"  
getReferenceClause = ( REF | REFERENCE )  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

getActionsClause

Get a list of actions from an action plan.

retrieveActionClause

Retrieve a set of properties or the associated object of an action.

QUOTED_STRING

Action name

getPropertiesClause

Retrieve a set of properties that is associated with an action.

getReferenceClause

Retrieve the object associated with an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE CORRECTION_SCHEMA_ACTION_PLAN 'CORRECT_INV_LOC'  
GET ACTIONS
```

OMBRETRIEVE CORRECTION_SCHEMA_ACTION_PLAN 'CORRECT_INV_LOC'
ACTION
'GEN_INV' GET REFERENCE

See Also

OMBCREATE CORRECTION_SCHEMA_ACTION_PLAN, OMBPROFILE

OMBRETRIEVE CUBE

Purpose

This command retrieves a Cube.

Prerequisites

Should be in Oracle Module context.

Syntax

```
retrieveCubeCommand = OMBRETRIEVE CUBE "cubeName" ( GET ( PROPERTIES
    "propertyKeyList" | "getReferenceIconSetClause" | MEASURES |
    COMPOSITE_DIMENSION | DIMENSION_USES | IMPLEMENTED_OBJECT | PARTITION
    ( LEVEL | HIERARCHY ) ) | GET AGGREGATE_FUNCTION | "measureLocator"
    GET ( PROPERTIES "propertyKeyList" | IMPLEMENTED_OBJECT |
    AGGREGATE_FUNCTIONS FOR DIMENSIONS "dimensionList" | PRECOMPUTE_LEVELS
    FOR DIMENSIONS "dimensionList" | CALCULATED_EXPRESSION ) |
    "dimensionUseLocator" GET ( PROPERTIES "propertyKeyList" | ( REF |
    REFERENCE ) ( DIMENSION | LEVEL | ROLE ) | IMPLEMENTED_OBJECT |
    DIMENSION_USE_POSITION ) | "compositeDimensionLocator" GET ( REF |
    REFERENCE ) DIMENSIONS )
cubeName = "QUOTED_STRING"
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
measureLocator = MEASURE "measureName"
dimensionList = "( " "dimension" { "," "dimension" } )"
dimensionUseLocator = DIMENSION_USE "dimensionUseName"
compositeDimensionLocator = COMPOSITE_DIMENSION "compositeDimensionName"
propertyKey = "UNQUOTED_STRING"
measureName = "QUOTED_STRING"
dimension = "QUOTED_STRING"
dimensionUseName = "QUOTED_STRING"
compositeDimensionName = "QUOTED_STRING"
```

Examples

See Also

OMBRETRIEVE, OMBCREATE CUBE, OMBALTER CUBE, OMBDROP CUBE

OMBRETRIEVE DATA_AUDITOR

Purpose

Retrieve data auditor details such as the number of operators.

Prerequisites

The current context must be in an Oracle Module.

Syntax

```
retrieveDataAuditorCommand = OMBRETRIEVE DATA_AUDITOR "dataAuditorName"
    "getDataAuditorDetailClause"
dataAuditorName = "QUOTED_STRING"
getDataAuditorDetailClause = GET ( PROPERTIES "propertyKeyList" |
    "getReferenceIconSetClause" | [ "operatorType" ] OPERATORS |
    "childType" ) | "getChildDetailClause" | "operatorLocator"
    "getOperatorDetailClause"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
operatorType = "UNQUOTED_STRING"
childType = "UNQUOTED_STRING"
getChildDetailClause = ( "childType" "childName" )+ GET ( PROPERTIES
    "propertyKeyList" | "childType" )
operatorLocator = OPERATOR "operatorName"
getOperatorDetailClause = GET ( PROPERTIES "propertyKeyList" |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause"
propertyKey = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
operatorName = "QUOTED_STRING"
```

Keywords And Parameters

retrieveDataAuditorCommand

Retrieve the detail of a data auditor such as how many operators are there.

dataAuditorName

Name of data auditor.

getDataAuditorDetailClause

Get properties or operators from the data auditor.

propertyKeyList

The list of property keys.

getReferenceIconSetClause

Retrieve the icon set referenced by the data auditor.

operatorType

Type of a mapping operator. The following operator types are available:
CUBE, DIMENSION, MATERIALIZED_VIEW, TABLE, VIEW.

childType

Type of a child that belongs to data auditor or operator

getChildDetailClause

Get the desired detail of a child object that belongs to the data auditor
or operator.

operatorLocator

Locate operator.

getOperatorDetailClause

Get the desired detail of a data auditor operator.

propertyKey

A property key for an object.

Basic properties for DATA_AUDITOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the data auditor

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the daa auditor

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

#

Properties for DATA_AUDITOR:

Name: ANALYZE_TABLE_SAMPLE_PERCENTAGE

Type: NUMBER

Valid Values: N/A

Default: 90

The default percentage of rows to be sampled when the target tables are analyzed for statistics to improve performance during insertion.

Name: ANALYZE_TABLE_STATEMENTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate statistics collection statement if this is true.

Name: ANSI_SQL_SYNTAX

Type: BOOLEAN

Valid Values: true, false

Default: true

A switch between ANSI and Oracle SQL Syntax.

Name: BULK_PROCESSING_CODE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate bulk processing code if this is true.

Name: BULK_SIZE

Type: NUMBER

Valid Values: N/A

Default: 50

The default number of rows to be fetched in batch during cursor processing.

Name: COMMIT_CONTROL

Type: STRING

Valid Values: AUTO_COMMIT, AUTO_CORR_COMMIT, MANUAL_COMMIT

Default: AUTO_COMMIT

Options for how commit is performed

Name: COMMIT_FREQUENCY

Type: NUMBER

Valid Values: N/A

Default: 1000

The default number of rows processed before a commit is issued.

Name: DEFAULT_AUDIT_LEVEL

Type: STRING

Valid Values: COMPLETE, ERROR_DETAILS, NONE, STATISTICS

Default: ERROR_DETAILS

The default audit level when the step is executed.

Name: DEFAULT_OPERATING_MODE

Type: STRING

Valid Values: ROW_BASED, ROW_BASED_TARGET_ONLY, SET_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY

Default: SET_BASED_FAIL_OVER_TO_ROW_BASED

The default operating mode.

Name: DEFAULT_PURGE_GROUP

Type: STRING

Valid Values: N/A

Default: WB

The default purge group to be used when the step is executed.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: ENABLE_PARALLEL_DML

Type: BOOLEAN

Valid Values: true, false

Default: true

Determine if PDML is enabled at runtime.

Name: ERROR_TRIGGER

Type: STRING

Valid Values: N/A

Default: "

Error trigger procedure name.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_MODE

Type: STRING

Valid Values: ALL_MODES, ROW_BASED, ROW_BASED_TARGET_ONLY, SET_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED,

SET_BASED_FAIL_OVER_TO_ROW_BASED_TARGET_ONLY

Default: ALL_MODES

The operating modes for which code should be generated.

Name: MAXIMUM_NUMBER_OF_ERRORS

Type: NUMBER

Valid Values: N/A

Default: 50

The default maximum number of errors encountered before terminating the step execution.

Name: OPTIMIZED_CODE

Type: BOOLEAN

Valid Values: true, false

Default: true

Attempt to generate optimized code if this is true.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TARGET_LOAD_ORDERING

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate target load ordering code.

Name: THRESHOLD_MODE

Type: STRING

Valid Values: PERCENTAGE, SIX_SIGMA

Default: PERCENTAGE

Use six sigma or percentage for failure thresholds.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

childName

Name of a child that belongs to data auditor, operator.

operatorName

Name of an operator.

Examples

```
OMBRETRIEVE DATA_AUDITOR 'MAP1' GET OPERATORS
```

```
OMBRETRIEVE DATA_AUDITOR 'MAP1' OPERATOR 'SRC1'  
GET PROPERTIES (BUSINESS_NAME, DESCRIPTION)
```

See Also

OMBRETRIEVE, OMBCREATE DATA_AUDITOR, OMBALTER DATA_AUDITOR,
OMBDROP DATA_AUDITOR

OMBRETRIEVE DATA_PROFILE

Purpose

Retrieve details of the Data Profile.

Prerequisites

Should be in the context of project.

Syntax

```
retrieveDataProfileCommand = OMBRETRIEVE DATA_PROFILE "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferenceLocationClause" |  
    "getReferenceIconSetClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

`retrieveDataProfileCommand`

This command retrieves the details of a Data Profile

`QUOTED_STRING`

Name of the existing Data Profile or path to the Data Profile.

`getPropertiesClause`

Retrieve a set of properties that is associated with a Data Profile.

Properties for `DATA_PROFILE`:

Name: `ABAP_DIRECTORY`

Type: `STRING`

Valid Values: `N/A`

Default: `/tmp`

Location where SAP data is dumped as flat files

Name: `CALCULATE_COMMON_FORMATS`

Type: `BOOLEAN`

Valid Values: `true, false`

Default: `false`

Setting this to true will enable common format discovery for all the columns in this profile.

Name: CALCULATE_DATATYPES

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable data type discovery for all the columns in this profile.

Name: CALCULATE_DATA_RULES

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable data rule profiling for the selected table.

Name: CALCULATE_DOMAINS

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable domain discovery.

Name: CALCULATE_FD

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable functional dependency discovery.

Name: CALCULATE_FK

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable foreign key discovery.

Name: CALCULATE_PATTERNS

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable pattern discovery.

Name: CALCULATE_REDUNDANT_COLUMNS

Type: BOOLEAN

Valid Values: true, false

Default: false

Setting this to true will enable redundant column discovery.

Name: CALCULATE_UK

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable unique key discovery.

Name: COPY_DATA

Type: BOOLEAN

Valid Values: true, false

Default: true

Setting this to true will enable copying of data from source to profile workspace.

Name: DOMAIN_MAX_COUNT

Type: NUMBER

Valid Values: N/A

Default: 100

The maximum number of distinct values in a column in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MAX_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 50

The maximum number of distinct values in a column, expressed as a

percentage of the total number of rows in the table, in order for that column to be discovered as possibly being defined by a domain. Domain Discovery of a column occurs if the number of distinct values in that column is at or below the Max Distinct Values Count property, AND, the number of distinct values as a percentage of total rows is at or below the Max Distinct Values Percent property.

Name: DOMAIN_MIN_COUNT

Type: NUMBER

Valid Values: N/A

Default: 2

The minimum number of rows for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: DOMAIN_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 1

The minimum number of rows, expressed as a percentage of the total number of rows, for the given distinct value in order for that distinct value to be considered as compliant with the domain. Domain Value Compliance for a value occurs if the number of rows with that value is at or above the Min Rows Count property, AND, the number of rows with that value as a percentage of total rows is at or above the Min Rows Percent property.

Name: FD_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 100

This is the minimum percentage of rows that need to satisfy a functional dependency relationship.

Name: FD_UK_FK_LHS_COUNT

Type: NUMBER

Valid Values: N/A

Default: 1

This is the maximum number of attributes for unique key and functional dependency profiling.

Name: FK_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 75

This is the minimum percentage of rows that need to satisfy a foreign key relationship.

Name: MAX_NUM_PATTERNS

Type: NUMBER

Valid Values: N/A

Default: 10

This tells the profiler to get the top-N patterns for the attribute.

Name: NULL_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 10

If the percentage of null values in a column is less than this threshold percent, then that column will be discovered as a possible Not Null column.

Name: NULL_VALUE

Type: STRING

Valid Values: N/A

Default: null

This value will be considered as the null value when profiling. Please enclose the value in single quotes. An unquoted null (the current default value) will be considered a database null.

Name: REDUNDANT_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 75

This is the minimum percentage of rows that are redundant with respect to a foreign key-unique key pair.

Name: SAMPLE_RATE

Type: NUMBER

Valid Values: 0 - 100

Default: 100

This value will be the percent of total rows that will be randomly selected during loading.

Name: UK_MIN_PERCENT

Type: NUMBER

Valid Values: 0 - 100

Default: 75

This is the minimum percentage of rows that need to satisfy a unique key relationship.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

getReferenceLocationClause

Retrieve the location that is set to the Data Profile.

getReferenceIconSetClause

Retrieve the icon set referenced by this Data Profile.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE DATA_PROFILE 'src_profile' GET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the Data Profile "src_profile"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE DATA_RULE

Purpose

Retrieve data rule details such as the data rule type.

Prerequisites

The current context of scripting must be a data rule module.

Syntax

```
retrieveDataRuleCommand = OMBRETRIEVE DATA_RULE "QUOTED_STRING"
    "retrieveDataRuleClause"
retrieveDataRuleClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getGroupsClause" |
    "getDomainValuesClause" ) | GROUP "retrieveGroupClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getGroupsClause = GROUPS
getDomainValuesClause = DOMAIN_VALUES
retrieveGroupClause = "QUOTED_STRING" ( GET ( "getGroupPropertiesClause" |
    "getAttributesClause" ) | "retrieveAttributeClause" )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
getGroupPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getAttributesClause = ATTRIBUTES
retrieveAttributeClause = ATTRIBUTE "QUOTED_STRING"
    "getAttributePropertiesClause"
getAttributePropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
```

Keywords And Parameters

retrieveDataRuleCommand

Retrieve a data rule

QUOTED_STRING

Data rule name.

retrieveDataRuleClause

Gets the groups, domain values or properties of a data rule.

getPropertiesClause

This clause retrieves all the properties.

Properties for DATA_RULE:

Name: ATTR_VALUE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

Description not available.

Name: FUNCTIONAL_DEP_THRESHOLD

Type: NUMBER

Valid Values: N/A

Default: 0

Description not available.

Name: IGNORE_NULLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LOCAL_MAX_COUNT

Type: STRING

Valid Values: N/A

Default: n

Description not available.

Name: LOCAL_MIN_COUNT

Type: STRING

Valid Values: N/A

Default: 1

Description not available.

Name: MAX_VALUE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: MIN_VALUE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: NAMEADDR_PASS_CONDITION

Type: STRING

Valid Values: PASS_PARSED, PASS_POSTALMATCHED_AVAIL

Default: PASS_PARSED

Description not available.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Description not available.

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Description not available.

Name: REMOTE_MAX_COUNT

Type: STRING

Valid Values: N/A

Default: n

Description not available.

Name: REMOTE_MIN_COUNT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: RULE_TYPE

Type: STRING

Valid Values: , ATTR_VALUE_RULE, DOMAIN_FORMAT_DATE_RULE,
DOMAIN_FORMAT_EMAIL_RULE, DOMAIN_FORMAT_IP_RULE, DOMAIN_
FORMAT_NUMBER_RULE,
DOMAIN_FORMAT_SSN_RULE, DOMAIN_FORMAT_TELEPHONE_RULE,
DOMAIN_FORMAT_URL_RULE, DOMAIN_LIST_RULE, DOMAIN_NO_NULL_
RULE,
DOMAIN_PATTERN_LIST_RULE, DOMAIN_RANGE_RULE, FUNCTIONAL_DEP_
RULE,
IDENTITY_RULE, NAMEADDRESS_RULE, REFERENCE_RULE, SET_RULE

Default: "

Description not available.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceIconSetClause

Retrieve the icon set referenced by the data rule.

getGroupsClause

Retrieves all the groups of a data rule.

getDomainValuesClause

Retreives all the domain values.

retrieveGroupClause

Retrieve a data rule group.

QUOTED_STRING

Group name.

propertyNameList

Comma separated list of property names. Property names are unquoted.

getGroupPropertiesClause

This clause retrieves all the properties of the data rule group.

getAttributesClause

Retrieves all the attributes of a data rule group.

retrieveAttributeClause

Retrieve a data rule group attribute.

QUOTED_STRING

Attribute name.

getAttributePropertiesClause

This clause retrieves all the properties of the data rule group attribute.

Examples

```
OMBRETRIEVE DATA_RULE 'STATE_NAME' GET PROPERTIES(RULE_TYPE)
```

```
OMBRETRIEVE DATA_RULE 'STATE_NAME' GET GROUPS
```

```
OMBRETRIEVE DATA_RULE 'STATE_NAME' GROUP 'THIS' GET ATTRIBUTES
```

```
OMBRETRIEVE DATA_RULE 'STATE_NAME' GET DOMAIN_VALUES
```

```
OMBRETRIEVE DATA_RULE 'REFERENCE' GET PROPERTIES (RULE_TYPE,  
LOCAL_MIN_COUNT, REMOTE_MIN_COUNT)
```

```
OMBRETRIEVE DATA_RULE 'STATE_NAME' GROUP 'THIS' ATTRIBUTE 'VALUE'  
GET
```

```
PROPERTIES(DATATYPE)
```

See Also

OMBRETRIEVE

OMBRETRIEVE DATA_RULE_MODULE

Purpose

Retrieve details of the data rule module.

Prerequisites

Should be in the context of project.

Syntax

```
retrieveDataRuleModuleCommand = OMBRETRIEVE DATA_RULE_MODULE
    "QUOTED_STRING" ( "getPropertiesClause" | "getReferenceIconSetClause"
    )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveDataRuleModuleCommand

Retrieve the data rule module.

QUOTED_STRING

Data rule module name.

getPropertiesClause

This clause retrieves all the properties.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceIconSetClause

Retrieve the icon set referenced by the data rule module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE DATA_RULE_MODULE 'br_module' GET PROPERTIES
(DESCRIPTION)
```

See Also

OMBRETRIEVE

OMBRETRIEVE DEPLOYMENT

Purpose

Retrieve details of the Deployment.

Prerequisites

Should be in the context of a Configuration.

Syntax

```
retrieveDeploymentCommand = OMBRETRIEVE DEPLOYMENT "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferencedControlCenterClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferencedControlCenterClause = GET ( REF | REFERENCE ) CONTROL_CENTER  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveDeploymentCommand

This command retrieves the details of a Deployment.

getPropertiesClause

Retrieve a set of properties that is associated with a Deployment.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE DEPLOYMENT 'QA_DEPLOYMENT' GET REF CONTROL_CENTER
```

This will retrieve a name of the referenced Control Center.

See Also

OMBLIST

OMBRETRIEVE DEPLOYMENT_ACTION_PLAN

Purpose

Retrieve the details of an existing deployment action plan.

Prerequisites

There must be a current working project.

Syntax

```
RetrieveActionPlanCommand = ( OMBRETRIEVE ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" ) ( "getActionsClause"
    | "retrieveActionClause" )
getActionsClause = GET ACTIONS
retrieveActionClause = ACTION "QUOTED_STRING" ( GET (
    "getPropertiesClause" | "getReferenceClause" ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceClause = ( REF | REFERENCE )
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

RetrieveActionPlanCommand

Retrieve the details of an existing deployment action plan.

getActionsClause

Get a list of actions from an action plan.

retrieveActionClause

Retrieve a set of properties or the associated object of an action.

getPropertiesClause

Retrieve a set of properties that is associated with an action.

PROPERTIES

The only valid property is OPERATION, which specifies the type of action to be taken.

getReferenceClause

Retrieve the object associated with an action.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

OMBRETRIEVE DEPLOYMENT_ACTION_PLAN 'MY_PLAN' GET ACTIONS

OMBRETRIEVE DEPLOYMENT_ACTION_PLAN 'MY_PLAN' ACTION 'MY_VIEW_CREATE'

GET PROPERTIES (OPERATION)

OMBRETRIEVE DEPLOYMENT_ACTION_PLAN 'MY_PLAN' ACTION 'MY_TABLE_DEPLOY'

GET REFERENCE

See Also

OMBCREATE DEPLOYMENT_ACTION_PLAN, OMBDEPLOY

OMBRETRIEVE DIMENSION

Purpose

This command retrieves a dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
retrieveDimensionCommand = OMBRETRIEVE DIMENSION "dimensionName" ( GET (
    PROPERTIES "propertyKeyList" | "getReferenceIconSetClause" |
    DIMENSION_ATTRIBUTES | LEVELS | HIERARCHIES | DIMENSION_ROLES | ( REF
    | REFERENCE ) SEQUENCE | DIMENSION_KEY COLUMN | SURROGATE_KEY |
    PARENT_KEY | BUSINESS_KEYS ) | "dimensionAttributeDetailClause" |
    "levelDetailClause" | "hierarchyDetailClause" | "roleDetailClause" )
dimensionName = "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ( REFERENCE | REF ) ICONSET
dimensionAttributeDetailClause = "dimensionAttributeLocator" GET
    PROPERTIES "propertyKeyList"
levelDetailClause = "levelLocator" ( GET ( PROPERTIES "propertyKeyList" |
    LEVEL_ATTRIBUTES | IMPLEMENTED_OBJECT ) | "levelAttributeDetailClause"
    | "levelRelationshipDetailClause" |
    "skipLevelRelationshipDetailClause" )
hierarchyDetailClause = "hierarchyLocator" GET ( PROPERTIES
    "propertyKeyList" | ( REF | REFERENCE ) LEVELS )
roleDetailClause = "roleLocator" ( GET ( PROPERTIES "propertyKeyList" ) )
propertyKey = "UNQUOTED_STRING"
dimensionAttributeLocator = DIMENSION_ATTRIBUTE "dimensionAttributeName"
levelLocator = LEVEL "levelName"
levelAttributeDetailClause = "levelAttributeLocator" GET ( PROPERTIES
    "propertyKeyList" | ( REF | REFERENCE ) ( DIMENSION_ATTRIBUTE |
    TYPE_THREE_SCD_PREVIOUS LEVEL_ATTRIBUTE | TYPE_THREE_SCD_CURRENT
    LEVEL_ATTRIBUTE ) | IMPLEMENTED COLUMN )
levelRelationshipDetailClause = LEVEL_RELATIONSHIP IN "hierarchyLocator"
    GET IMPLEMENTED COLUMN
skipLevelRelationshipDetailClause = SKIP_LEVEL_RELATIONSHIP IN
    "hierarchyLocator" GET ( PARENT LEVEL | IMPLEMENTED COLUMN )
hierarchyLocator = HIERARCHY "hierarchyName"
roleLocator = DIMENSION_ROLE "roleName"
dimensionAttributeName = "QUOTED_STRING"
levelName = "QUOTED_STRING"
levelAttributeLocator = LEVEL_ATTRIBUTE "levelAttributeName"
hierarchyName = "QUOTED_STRING"
roleName = "QUOTED_STRING"
levelAttributeName = "QUOTED_STRING"
```

Examples

OMBRETRIEVE DIMENSION 'DIM' GET LEVELS

See Also

OMBRETRIEVE, OMBCREATE DIMENSION, OMBALTER DIMENSION, OMBDROP DIMENSION

OMBRETRIEVE DRILL_PATH

Purpose

Retrieve details of a Drill Path.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveDrillPathCommand = OMBRETRIEVE DRILL_PATH "QUOTED_STRING"  
    "retrieveDrillPathClauseDetails"  
retrieveDrillPathClauseDetails = ( "retrieveDrillLevelClause" | GET (   
    "getPropertiesClause" | "getReferenceIconSetClause" | DRILL_LEVELS | (   
        REF | REFERENCE ) HIERARCHY | ( REF | REFERENCE ) DIMENSION | ( REF |   
        REFERENCE ) ROLE | DEPENDENTS ) )  
retrieveDrillLevelClause = DRILL_LEVEL "QUOTED_STRING" ( GET (   
    "getPropertiesClause" | ( REF | REFERENCE ) ITEMS | ( REF | REFERENCE   
    ) USING JOINS | ( REF | REFERENCE ) LEVEL | CHILDREN | PARENT ) )  
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"  
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveDrillPathCommand

Retrieves the drill path.

QUOTED_STRING

name of the drill path.

retrieveDrillPathClauseDetails

This clause retrieves the contents of a drill path.

retrieveDrillLevelClause

Retrieves a specific drill level.

QUOTED_STRING

name of the drill level.

GET

For drill level this clause retrieves the following

REF ITEMS retrieves the list of Items referenced by this Drill Level.

REF USING JOINS retrieves the list of Joins used to join the Item Folder

containing Items in this Drill Level to the Item Folder containing Items in the parent Drill Level.

REF LEVEL retrieves the dimensional Level corresponding to the Drill Level.

CHILDREN retrieves the child Drill Levels.

getPropertiesClause

Retrieves the properties of the object.

Basic properties for DRILL_PATH:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill path

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill path

Basic properties for DRILL_LEVEL:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill level

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill level

Properties for DRILL_PATH:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.

2. " " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

propertyNameList

This is the list of property names.

Examples

```
OMBRETRIEVE DRILL_PATH 'PRODUCT_ROLLUP' GET  
PROPERTIES(DESCRIPTION)
```

See Also

OMBALTER DRILL_PATH, OMBCREATE DRILL_PATH

OMBRETRIEVE DRILL_TO_DETAIL

Purpose

Retrieve details of a Drill to Detail.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveDrillToDetailCommand = OMBRETRIEVE DRILL_TO_DETAIL "QUOTED_STRING"
    "retrieveDrillToDetailClauseDetails"
retrieveDrillToDetailClauseDetails = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | ( REF | REFERENCE ) ITEMS )
getPropertiesClause = PROPERTIES "( " "propertyNameList" )"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveDrillToDetailCommand

Retrieves the drill to detail.

QUOTED_STRING

name of the drill to detail.

retrieveDrillToDetailClauseDetails

This clause retrieves the contents of a drill to detail.

GET

This clause retrieves the following

REF ITEMS retrieves the list of Items that use this drill to detail.

getPropertiesClause

Retrieves the properties of the object.

Basic properties for DRILL_TO_DETAIL:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the drill to detail

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the drill to detail

Properties for DRILL_TO_DETAIL:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts
to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

propertyNameList

This is the list of property names.

Examples

```
OMBRETRIEVE DRILL_TO_DETAIL 'COLORS' GET PROPERTIES(DESCRIPTION)
```

See Also

```
OMBALTER DRILL_TO_DETAIL, OMBCREATE DRILL_TO_DETAIL
```

OMBRETRIEVE EXPERT

Purpose

To retrieve metadata on an expert.

Prerequisites

In the context of an expert module.

Syntax

```

retrieveExpertCommand = OMBRETRIEVE EXPERT "QUOTED_STRING" (
    "retrieveExpertClause" | "retrieveTaskClause" |
    "retrieveTransitionClause" | "retrieveParameterClause" |
    "retrieveVariableClause" )
retrieveExpertClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getExpertSCOClauses" )
retrieveTaskClause = TASK "QUOTED_STRING" ( GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | PARAMETERS | INCOMING TRANSITIONS |
    OUTGOING TRANSITIONS | ( REF | REFERENCE ) EXPERT ) |
    "retrieveParameterClause" )
retrieveTransitionClause = TRANSITION "QUOTED_STRING" GET (
    "getPropertiesClause" | SOURCE TASK | DESTINATION TASK )
retrieveParameterClause = PARAMETER "QUOTED_STRING" GET (
    "getPropertiesClause" | "getBindingClause" )
retrieveVariableClause = VARIABLE "QUOTED_STRING" GET
    "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getExpertSCOClauses = ( TASKS | "TASK_TYPE" TASKS | NESTED_EXPERT TASKS |
    TRANSITIONS | PARAMETERS | VARIABLES )
getBindingClause = BOUND ( VARIABLE | PARAMETER )
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }

```

Keywords And Parameters

retrieveExpertCommand

Retrieve definition of the expert.

retrieveExpertClause

Retrieve either the properties or the child object definitions for the expert.

retrieveTaskClause

Retrieve definition of a task.

retrieveTransitionClause

Retrieve definition of a transition.

retrieveParameterClause

Retrieve definition of an expert or a task parameter.

retrieveVariableClause

Retrieve definition of a variable.

getPropertiesClause

Get the values of the specified properties.

Basic properties for EXPERT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the expert

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the expert

Basic properties for TASK:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the task

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the task. This is equivalent to the Goal of task in the expert editor.

Name: INSTRUCTION
Type: STRING(4000)
Valid Values: N/A
Default: "
The instruction for the task

Name: PREPROCESSING
Type: STRING
Valid Values: N/A
Default: N/A
The pre-processing script for the task

Name: MAIN
Type: STRING
Valid Values: N/A
Default: N/A
The main script for the task

Name: POSTPROCESSING
Type: STRING
Valid Values: N/A
Default: N/A
The post-processing script for the task

Basic properties for START TASK:

Name: PROC_DECL
Type: STRING(4000)
Valid Values: N/A
Default: "
The procedure declaration for the expert.

Basic properties for TRANSITION:

Name: BUSINESS_NAME

Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the transition

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the transition

Name: TRANSITION_CONDITION
Type: STRING(4000)
Valid Values: N/A
Default: "
Condition of the transition

Name: TRANSITION_ORDER
Type: NUMBER
Valid Values: N/A
Default: N/A
Order of the transition

Basic properties for PARAMETER:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the parameter

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the parameter

Name: DATATYPE
Type: STRING
Valid Values: STRING, NUMBER, BOOLEAN, ARRAY
Default: STRING
Datatype of the parameter

Name: DIRECTION
Type: STRING
Valid Values: IN, OUT, INOUT
Default: IN
Direction of the parameter

Name: VALUE
Type: Same as datatype of the parameter
Valid Values: N/A
Default: N/A
The static value of the parameter

Basic properties for VARIABLE:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the variable

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the variable

Name: DATATYPE
Type: STRING
Valid Values: STRING, NUMBER, BOOLEAN, ARRAY
Default: STRING

Datatype of the variable

Name: VALUE

Type: Same as datatype of the variable

Valid Values: N/A

Default: N/A

The static value of the variable

Properties for EXPERT:

Name: CLOSE_ASSISTANT_ON_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the task assistant should be automatically closed after the expert has been run.

Name: CLOSE_WINDOWS_ON_EXECUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Close all open windows when this expert is run.

Name: FINISH_DIALOG_ON_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: true

Shows the finish dialog upon completion of expert.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LOGGING

Type: BOOLEAN

Valid Values: true, false

Default: false

Log output to file when this expert is being run. A log file will be created in <shiphome>/owb/log directory whenever this expert is run.

Name: MENU_ITEM_DISPLAY_STRING

Type: STRING

Valid Values: N/A

Default: "

The display string when this expert is added as a menu item.

Name: ONLY_RUN_FROM_MENU

Type: BOOLEAN

Valid Values: true, false

Default: false

Only allow this expert to be run when it is attached to a menu item.

Name: REVERT_TO_SAVED_ON_ERROR

Type: BOOLEAN

Valid Values: true, false

Default: false

Revert to saved metadata if error occurs when the expert is run.

Name: RUN_STANDALONE

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the expert should be run as a standalone in expert assistant mode or not.

Name: SAVE_ALL_BEFORE_START

Type: BOOLEAN

Valid Values: true, false

Default: false

Save all metadata before running the expert.

Name: SHOW_BUSY_DIALOG

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether busy dialog should be shown when OMB or Java tasks are executed in non-standalone mode.

Name: SHOW_LOG_WINDOW

Type: BOOLEAN

Valid Values: true, false

Default: false

Sets whether the log window should be shown when running the expert.

Name: SHOW_PROGRESS_GRAPH

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the progress graph dialog should be shown when running the expert.

Name: SHOW_TASK_ASSISTANT

Type: BOOLEAN

Valid Values: true, false

Default: true

Sets whether the task assistant should be shown when running the expert.

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

getReferenceIconSetClause

Retrieve the icon set referenced by this expert module.

getExpertSCOClauses

Get a list of child object of the specified type for the expert.

getBindingClause

Get the bound variable or parameter.

propertyNameList

The list of property names.

Examples

This command will retrieve the properties for the expert EXP1:

```
OMBRETRIEVE EXPERT 'EXP1' GET PROPERTIES (BUSINESS_NAME,  
DESCRIPTION)
```

This command will retrieve the description properties on task 'MY_TASK':

```
OMBRETRIEVE EXPERT 'EXP1' TASK 'MY_TASK' GET PROPERTIES  
(DESCRIPTION)
```

See Also

OMBRETRIEVE, OMBCREATE EXPERT, OMBALTER EXPERT, OMBDROP EXPERT

OMBRETRIEVE EXPERT_MODULE

Purpose

To retrieve metadata on an expert module.

Prerequisites

In the context of a project.

Syntax

```
retrieveExpertModuleCommand = OMBRETRIEVE EXPERT_MODULE "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferenceIconSetClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveExpertModuleCommand

Retrieve definition of an expert module.

getPropertiesClause

Get the values of the specified properties.

Basic properties for EXPERT_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the expert module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the expert module

getReferenceIconSetClause

Retrieve the icon set referenced by this expert module.

propertyNameList

The list of property names to get.

Examples

This command will retrieve the properties for the expert module EM1:

```
OMBRETRIEVE EXPERT_MODULE 'EM1' GET PROPERTIES (BUSINESS_NAME,  
DESCRIPTION)
```

See Also

OMBRETRIEVE

OMBRETRIEVE EXTERNAL_TABLE

Purpose

Retrieve details of an external table.

Prerequisites

Should be in the context of an Oracle module.

Syntax

```
retrieveExternalTableCommand = OMBRETRIEVE EXTERNAL_TABLE "QUOTED_STRING"
    ( "retrieveExternalTableClause" | "retrieveExternalTableColumnClause"
      | "retrieveExternalTableDatafileClause" |
      "retrieveDataRuleUsageClause" )
retrieveExternalTableClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getExternalTableObjectsClause" )
retrieveExternalTableColumnClause = COLUMN "QUOTED_STRING" GET (
    "getPropertiesClause" | FIELD )
retrieveExternalTableDatafileClause = DATA_FILE "QUOTED_STRING" GET
    "getPropertiesClause"
retrieveDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" ( GET (
    "getPropertiesClause" | GROUPS ) | GROUP "QUOTED_STRING" ( GET (
    "getPropertiesClause" | ATTRIBUTES | REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) ) ) | ATTRIBUTE "QUOTED_STRING" (
    GET ( "getPropertiesClause" | REF COLUMN ) ) ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getExternalTableObjectsClause = COLUMNS | FLAT_FILE | RECORD |
    DEFAULT_LOCATION | DATA_FILES | COLUMN AT POSITION "INTEGER_LITERAL" |
    DATA_RULE_USAGES
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveExternalTableCommand

Retrieve details from an external table.

QUOTED_STRING

The name of the external table.

retrieveExternalTableClause

Retrieve details from an external table.

retrieveExternalTableColumnClause

Retrieve details from an external table column.

retrieveExternalTableDatafileClause

Retrieve details from one of the external table's data files.

retrieveDataRuleUsageClause

This clause retrieves the data rule usages.

QUOTED_STRING

Name of data rule usage, group or attribute.

GROUPS

Retrieve the names of all relation groups in the data rule usage.

ATTRIBUTES

Retrieve the names of all attributes in a data rule usage group.

TABLE

Table name associated with the data rule usage group.

VIEW

View name associated with the data rule usage group.

MATERIALIZED_VIEW

Materialized view name associated with the data rule usage group.

EXTERNAL_TABLE

External table name associated with the data rule usage group.

COLUMN

Column name associated with the data rule usage group attribute.

getPropertiesClause

Retrieve specified properties.

getExternalTableObjectsClause

Retrieve a record reference, flat file reference, columns, or data files.

propertyNameList

The names of the properties whose values you want to retrieve.

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARHCAR, VARCHAR2

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for EXTERNAL_TABLE:

Name: BAD_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the bad file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: BAD_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the bad file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

When deployable is set to true, a script to create an External Table is generated.

Name: DISCARD_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the discard file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: DISCARD_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the discard file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: ENDIAN

Type: STRING

Valid Values: BIG, LITTLE, PLATFORM

Default: PLATFORM

Data endian should be platform default, little or big. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LOAD_NULLS_WHEN_MISSING_VALUES

Type: BOOLEAN

Valid Values: true, false

Default: false

If TRUE, then NULLs are loaded for any missing values in the record. If FALSE, then records with missing values are rejected. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: LOG_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: USE_DEFAULT_LOCATION

Location to use when creating the log file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: LOG_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name to use when creating the log file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: NLS_CHARACTERSET

Type: STRING

Valid Values: N/A

Default: "

NLS Characterset of the file. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: NUMBER_OF_REJECTS_ALLOWED

Type: NUMBER

Valid Values: 0 - 2147483647

Default: 0

The number of rejects allowed before processing is terminated.

Name: PARALLEL_ACCESS_DRIVERS

Type: NUMBER

Valid Values: 1 - 63999

Default: 1

The number of parallel access drivers to enable.

Name: PARALLEL_ACCESS_MODE

Type: BOOLEAN

Valid Values: true, false

Default: false

Enable or disable parallel processing.

Name: REJECTS_ARE_UNLIMITED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable or disable limiting the number of rejected records.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: STRING_SIZES_IN

Type: STRING

Valid Values: BYTES, CHARACTERS

Default: BYTES

String sizes are in bytes or characters. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Name: TRIM

Type: STRING

Valid Values: BOTH, LEFT, NONE, RIGHT, SQL*LOADER

Default: NONE

Specification from trim option on input fields. When the Access Parameters property is specified for the External Table, this configuration parameter is ignored.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBRETRIEVE EXTERNAL_TABLE 'SRC_TABLE' GET PROPERTIES  
(DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the external table "SRC_TABLE"'s description and business name.

See Also

OMBRETRIEVE, OMBCREATE EXTERNAL_TABLE, OMBALTER EXTERNAL_TABLE, OMBDROP EXTERNAL_TABLE

OMBRETRIEVE FLAT_FILE

Purpose

Retrieve details of a flat file.

Prerequisites

Create and change context to a flat file module.

Syntax

```
retrieveFlatFileCommand = OMBRETRIEVE FLAT_FILE "QUOTED_STRING" (
    "retrieveFlatFileClause" | "retrieveFlatFileObjectsClause" )
retrieveFlatFileClause = GET ( "getPropertiesClause" | "getRecordsClause"
    | "getReferenceIconSetClause" )
retrieveFlatFileObjectsClause = RECORD "QUOTED_STRING" ( ( GET
    "getPropertiesClause" ) | "getFieldsClause" | "retrieveFieldClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getRecordsClause = RECORDS
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getFieldsClause = GET FIELDS
retrieveFieldClause = "getFieldByNameClause" | "getFieldAtPositionClause"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
getFieldByNameClause = FIELD "QUOTED_STRING" GET "getPropertiesClause"
getFieldAtPositionClause = GET FIELD AT POSITION "INTEGER_LITERAL"
```

Keywords And Parameters

retrieveFlatFileCommand

Retrieve the details of a flat file.

QUOTED_STRING

The name of a flat file in quotes.

retrieveFlatFileClause

Retrieve properties of the flat file.

retrieveFlatFileObjectsClause

Retrieve details of the flat file's records and fields.

getPropertiesClause

Retrieve specified properties from the flat file, record, or field.

getRecordsClause

Retrieve a list of records from the flat file.

`getFieldsClause`

Retrieve a list of fields from the record.

`retrieveFieldClause`

Identify a specific field to retrieve properties from.

`propertyNameList`

The names of the properties whose values you want to retrieve.

Properties for FLAT_FILE:

Name: DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default "

The name of the "sampled" file. Also the default data file value used in SQL*Loader maps and External Tables.

Name: IS_DELIMITED

Type: BOOLEAN

Valid Values: true, false, 1, 0

Default: true

True indicates that this flat file is delimited. False indicates that its fields are defined by fixed lengths

Name: CHARACTERSET

Type: STRING

Valid Values:

AL24UTFFSS,AR8ARABICMAC,AR8ARABICMACS,AR8ISO8859P6,AR8MSAWIN,A
R8MSWIN1256,BLT8CP921,BLT8EBCDIC1112,BLT8MSWIN1257,BLT8PC775,CDN8PC
863,CL8EBCDIC1025,CL8EBCDIC1025X,CL8ISO8859P5,CL8KOI8R,CL8MACCYRILLI
C,CL8MACCYRILLICS,CL8MSWIN1251,D8EBCDIC273,DK8EBCDIC277,EE8EBCDIC
870,EE8ISO8859P2,EE8MACCE,EE8MACCES,EE8MACCROATIAN,EE8MACCROATI
ANS,EE8MSWIN1250,EE8PC852,EL8EBCDIC875,EL8ISO8859P7,EL8MACGREEK,EL8
MACGREEKS,EL8MSWIN1253,EL8PC437S,EL8PC737,EL8PC869,F8EBCDIC297,I8EBC
DIC280,IS8MACICELANDIC,IS8MACICELANDICS,IS8PC861,IW8EBCDIC424,IW8IS
O8859P8,IW8MACHEBREW,IW8MACHEBREWS,IW8MSWIN1255,JA16EBCDIC930,J
A16EUC,JA16EUCYEN,JA16MACSJIS,JA16SJIS,JA16SJISYEN,JA16VMS,KO16KSC560
1,LT8MSWIN921,N8PC865,NEE8ISO8859P4,RU8PC855,RU8PC866,S8EBCDIC278,SE8I
SO8859P3,TH8MACTHAI,TH8MACTHAIS,TH8TISASCII,TR8EBCDIC1026,TR8MAC
TURKISH,TR8MACTURKISHS,TR8MSWIN1254,TR8PC857,US7ASCII,US8PC437,UTF

8,WE8EBCDIC284,WE8EBCDIC285,WE8EBCDIC37,WE8EBCDIC37C,WE8EBCDIC500,WE8EBCDIC500C,WE8EBCDIC871,WE8ISO8859P1,WE8ISO8859P9,WE8MACROMAN8,WE8MACROMAN8S,WE8MSWIN1252,WE8PC850,WE8PC860,ZHS16CGB231280,ZHS16GBK,ZHS16MACCGB231280,ZHT16BIG5,ZHT16MSWIN950,ZHT32EUC

Default: WE8MSWIN1252

The character set of the data file.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

The character(s) which denote the end of a physical record in a data file.

A hex value may be entered by entering embedded single quotes twice as:

'x"0f"' (all are single quotes). The outside single quote indicates a quoted string and the inside single quotes single-quote x single-quote single-quote 0f single-quote single-quote single-quote. (Please note that this is not the FIELD_DELIMITER.

Name: RECORD_LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0 (Records are delimited by default)

The length (in characters) of the records in the data file.

Name: RECORD_TYPE_COLUMN_NUMBER

Type: NUMBER

Valid Values: 0+

Default: 0

The column which contains the record type values for a delimited, multi-record type file.

Name: RECORD_TYPE_START_POSITION

Type: NUMBER

Valid Values: 0+

Default: 0

The starting position of the field (relative to 0) which contains the record type values for a fixed-length, multi-record type file.

Name: RECORD_TYPE_END_POSITION

Type: NUMBER

Valid Values: 0+

Default: 0

The ending position of the field (relative to 0) which contains the record type values for a fixed-length, multi-record type file.

Name: NUMBER_OF_RECORDS_TO_SKIP

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The default number of records to skip when loading this file.

Name: FIELD_DELIMITER

Type: STRING

Valid Values: Any single character

Default: ',' (Comma)

The character to divide the fields in a delimited file.

Name: FIELD_LEFT_ENCLOSURE

Type: STRING

Valid Values: Any single character

Default: None

A character to enclose fields which may contain the field delimiter.

Name: FIELD_RIGHT_ENCLOSURE

Type: STRING

Valid Values: Any single character

Default: None

A character to enclose fields which may contain the field delimiter.

Name: NUMBER_OF_PHYSICAL_RECORDS_PER_LOGICAL

Type: Number

Valid Values: 0+

Default: 0

Set this value if you wish to concatenate a fixed number of physical

records to form a single logical record.

Name: CONTINUE_IF_ENDS_WITH

Type: STRING

Valid Values: Any single character

Default: None

Set this value if you wish to concatenate a variable number of physical records to form a single logical record, determined by records ending with this character.

Name: CONTINUE_IF_STARTS_WITH

Type: STRING

Valid Values: N/A

Default: None

Set this value if you wish to concatenate a variable number of physical records to form a single logical record, determined by records beginning with this character.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the record

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the record

Properties for RECORD:

Name: RECORD_TYPE_VALUE

Type: STRING

Valid Values: N/A

Default: None

This is a mandatory property for each record of a multi-record type file.

It is the string which will identify this record type in the data file.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the record

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the record

Properties for FIELD:

Name: DATATYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARRAW,

VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

This is the SQL*Loader data type for the field.

Name: MAXIMUM_LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0

This is the maximum length of the field.

Name: LENGTH

Type: NUMBER

Valid Values: 0+

Default: 0

Deprecated. This is the length of the field in a fixed length file. This is the max length of the field in a delimited file.

Name: PRECISION

Type: NUMBER

Valid Values: Depends on data type

Default: 0

Precision of the field.

Name SCALE

Type: NUMBER

Valid Values: Depends on data type

Default: 0

Scale of the field

Name: START_POSITION

Type: NUMBER

Valid Values: 1+

Default: 1

The starting position of a field for a fixed length file.

Name: END_POSITION

Type: NUMBER

Valid Values: 1+

Default: 1

The ending position of a field for a fixed length file.

Name: SQL_DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: DEFAULT. This will derive the SQL_DATATYPE from the value of DATATYPE.

The data type which the field will be treated as in mapping and for External Tables.

Name: SQL_LENGTH

Type: NUMBER

Valid Values: 1 - 4000

Default: 0

Name: SQL_PRECISION

Type: NUMBER

Valid Values: 1 - 38

Default: 1

Name: SQL_SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 0

Name: MASK

Type: STRING

Valid Values: N/A

Default: None

This is the mask used to define the format of DATE fields in the data file.

Name: NULL_IF

Type: STRING

Valid Values: Either = or != followed by either 'BLANKS', a single quoted string, or a hexadecimal string

Default: None

If this condition is true for a field, the value loaded will be NULL.

Name: DEFAULT_IF

Type: STRING

Valid Values: Either = or != followed by either 'BLANKS', a single quoted string, or a hexadecimal string

Default: None

If this condition is true for a field, the value loaded will be either NULL or 0, dependent on data type.

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the field

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the field

getFieldByNameClause

Identify a specific field by its name.

getFieldAtPositionClause

Identify a specific field by its position in the record.

Examples

```
OMBRETRIEVE FLAT_FILE 'SRC_FILE' GET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the flat file "SRC_FILE"'s description and business name.

```
OMBRETRIEVE FLAT_FILE 'SRC_FILE_2' GET RECORDS
```

This will retrieve the records of the flat file "SRC_FILE_2".

```
OMBRETRIEVE FLAT_FILE 'SRC_FILE_3' RECORD 'EMPLOYEE' GET FIELDS
```

This will retrieve the fields of the "EMPLOYEE" record.

OMBRETRIEVE FLAT_FILE 'TARGET_FILE' RECORD 'TARGET_FILE' GET FIELD
AT

POSITION 3 GET PROPERTIES (UOID)

This will retrieve the UOID of the third field in the single record flat
file "TARGET_FILE".

See Also

OMBRETRIEVE

OMBRETRIEVE FLAT_FILE_MODULE

Purpose

Retrieve details from a flat file module.

Prerequisites

Should be in the context of a project.

Syntax

```
retrieveFlatFileModuleCommand = OMBRETRIEVE FLAT_FILE_MODULE
    "QUOTED_STRING" ( "getPropertiesClause" | "getReferenceLocationClause"
    | "getReferenceDefaultLocationClause" |
    "getReferenceMetadataLocationClause" | "getReferenceIconSetClause" |
    "getReferenceLocationsClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT
    LOCATION
getReferenceMetadataLocationClause = GET ( REF | REFERENCE )
    METADATA_LOCATION
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

`retrieveFlatFileModuleCommand`

Retrieve details from a flat file module.

`QUOTED_STRING`

The name of the flat file module to retrieve details from.

`getPropertiesClause`

Retrieve specified property values from the flat file module.

`getReferenceLocationClause`

Retrieve the name of the runtime location referenced by this flat file module.

`getReferenceDefaultLocationClause`

Retrieve the default runtime location referenced by this flat file module.

`getReferenceIconSetClause`

Retrieve the icon set referenced by this flat file module.

getReferenceLocationsClause

Retrieve the runtime locations referenced by this flat file module.

propertyNameList

The names of the properties whose values you want to retrieve.

Basic properties for FLAT_FILE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of the flat file module.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the flat file module.

Examples

```
OMBRETRIEVE FLAT_FILE_MODULE 'src_module' GET PROPERTIES  
(DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the flat file module "src_module"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE FUNCTION

Purpose

Retrieve details of the Function.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations or WB_PREDEFINED_TRANS for Predefined Transformations.

WB_CUSTOM_TRANS may be modified by an administrator. WB_PREDEFINED_TRANS

may not be modified.

WB_CUSTOM_TRANS and WB_PREDEFINED_TRANS are not dependent on any project.

Syntax

```
retrieveFunctionCommand = OMBRETRIEVE FUNCTION "QUOTED_STRING" (  
    "getPropertiesClause" | "getFuncProcParameterClause" |  
    "getFuncProcParameterPositionClause" | "getFuncProcSignatureClause" |  
    "retrieveFuncProcParameterClause" | "getRelationalDependentsClause" |  
    "getReferenceIconSetClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getFuncProcParameterClause = GET PARAMETERS  
getFuncProcParameterPositionClause = GET PARAMETER AT POSITION  
    "INTEGER_LITERAL"  
getFuncProcSignatureClause = GET SIGNATURE  
retrieveFuncProcParameterClause = PARAMETER "QUOTED_STRING"  
    "getPropertiesClause"  
getRelationalDependentsClause = GET ( REF | REFERENCE ) ( TABLES | VIEWS |  
    MATERIALIZED_VIEWS | SEQUENCES | FUNCTIONS | PROCEDURES | PACKAGES )  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveFunctionCommand

This command retrieves the details of a Function

QUOTED_STRING

Name of the existing Function or path to the Function.

getPropertiesClause

Used to get properties (core, user-defined) for function. Valid properties are as shown:

Basic properties for FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Function

Name: RETURN_TYPE

Type: STRING

Valid Values: PLS_INTEGER, BINARY_INTEGER, BOOLEAN, NUMBER, FLOAT, CHAR,

VARCHAR, VARCHAR2, DATE

Default: NUMBER

Set the Return Type for Function

Name: IMPLEMENTATION

Type: STRING

Valid Values: N/A

Default: "

Set the code for Function which is included global variable declaration and code between BEGIN and END.

Name: IS_DETERMINISTIC

Type: BOOLEAN

Valid Values: true, false

Default: false

This setting helps the optimizer avoid redundant function calls.

Name: IS_PARALLEL_ENABLE

Type: BOOLEAN

Valid Values: true, false

Default: false

This option sets flag to a stored function can be used safely in the slave sessions of parallel DML evaluations.

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB, BOOLEAN,

CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO

MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARHCAR,

VARCHAR2, XMLTYPE

Default: NUMBER

Set the data type for Parameter

Name: IN_OUT

Type: STRING

Valid Values: IN, OUT, INOUT

Default: 'IN'

Set the parameter mode for Parameter

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Set the default value for Parameter

Properties for FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getFuncProcParameterClause

Get all the parameter names of the Function

`getFuncProcParameterPositionClause`

Get the parameter position of Function

`getFuncProcSignatureClause`

Get the complete signature of the Function which includes parameter names, datatype, in/out type and default values

`retrieveFuncProcParameterClause`

Get the parameter information such as datatype, default value, in/out type and position

`QUOTED_STRING`

Name of the existing Parameter

`getRelationalDependentsClause`

This clause retrieves referential dependencies to other relational objects.

`propertyNameList`

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE FUNCTION 'func' GET PROPERTIES (DESCRIPTION, UID,
BUSINESS_NAME, RETURN_TYPE, IMPLEMENTATION, IS_DETERMINISTIC,
IS_PARALLEL_ENABLE, IS_IMPORTED)
```

This will retrieve the Function "func's description, uid, business name, return type, implementation, and boolean values of deterministic parallel_enable and imported.

If Packaged Function is overloaded, first find the Signature by using OMBLIST command, and then use OMBRETRIEVE command using appropriate signature.

Example, if OMBLIST FUNCTIONS gives following two signatures,

```
FUNC_1 (NUMBER) RETURN NUMBER
```

```
FUNC_1 (VARCHAR2, NUMBER) RETURN NUMBER
```

The OMBRETRIEVE Syntax to retrieve the first one will be as follows

```
OMBRETRIEVE FUNCTION 'FUNC_1 \ (NUMBER\ ) RETURN NUMBER' GET
PROPERTIES
```

(DESCRIPTION, BUSINESS_NAME)

See Also

OMBRETRIEVE

OMBRETRIEVE GATEWAY_MODULE

Purpose

Retrieve details of a gateway module.

Prerequisites

Should be in the context of project.

Syntax

```
retrieveGatewayModuleCommand = OMBRETRIEVE GATEWAY_MODULE "QUOTED_STRING"  
    ( "getPropertiesClause" | "getReferenceLocationClause" |  
      "getReferenceDefaultLocationClause" |  
      "getReferenceMetadataLocationClause" | "getReferenceIconSetClause" |  
      "getReferenceLocationsClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION  
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT  
    LOCATION  
getReferenceMetadataLocationClause = GET ( REF | REFERENCE )  
    METADATA_LOCATION  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveGatewayModuleCommand

Specify the gateway module from which to retrieve details.

getPropertiesClause

Retrieve the properties of a gateway module.

Basic properties for GATEWAY_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the gateway module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the gateway module

Name: GATEWAY_TYPE

Type: STRING

Valid Values: N/A

Default: N/A

Type of gateway module

getReferenceLocationClause

Retrieve the name of the runtime location referenced by this gateway module.

getReferenceDefaultLocationClause

Retrieve the default runtime location referenced by this gateway module.

getReferenceIconSetClause

Retrieve the icon set referenced by this gateway module.

getReferenceLocationsClause

Retrieve the runtime locations referenced by this gateway module.

propertyNameList

A list of property names.

Examples

The following line retrieves the description of a gateway module:

```
OMBRETRIEVE GATEWAY_MODULE 'db2_module' GET  
PROPERTIES(DESCRIPTION)
```

See Also

OMBRETRIEVE

OMBRETRIEVE ICONSET

Purpose

To retrieve the properties of an iconset.

Prerequisites

Any context.

Syntax

```
retrieveIconSetCommand = OMBRETRIEVE ICONSET "QUOTED_STRING" GET
    "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveIconSetCommand

This command retrieves the properties of an iconset.

QUOTED_STRING

The name of the iconset to retrieve.

getPropertiesClause

This clause gets the values for a list of properties.

Basic properties for ICONSET:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the iconset

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the iconset

Name: BELONGS_TO_GROUP

Type: STRING

Valid Values: N/A

Default: "

Name of the Group to which the iconset belongs

Name: CANVAS_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the canvas icon (36x36)

Name: PALETTE_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the palette icon (18x18)

Name: TREE_ICON

Type: STRING

Valid Values: N/A

Default: "

URL of the tree icon (16x16)

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE ICONSET 'ICON1' GET PROPERTIES (DESCRIPTION, BELONGS_
TO_GROUP)
```

See Also

OMBCREATE ICONSET, OMBALTER ICONSET, OMBDROP ICONSET, OMBLIST
ICONSETS

OMBRETRIEVE_IMPORT_ACTION_PLAN

Purpose

To display the details of a transient import action plan.

Prerequisites

In the context of a project.

Syntax

```
retrieveImportActionPlanCommand = OMBRETRIEVE ( IMPORT_ACTION_PLAN )  
    "QUOTED_STRING" ( "getActionsClause" | "retrieveActionClause" )  
getActionsClause = GET ACTIONS  
retrieveActionClause = ACTION "QUOTED_STRING" GET ( "getPropertiesClause"  
    | "getRefSourceOrTargetClause" )  
getPropertiesClause = PROPERTIES "( "propertyNameList" )"  
getRefSourceOrTargetClause = ( REF | REFERENCE ) ( SOURCE | TARGET )  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveImportActionPlanCommand

This command is for displaying details in an import action plan.

QUOTED_STRING

The name of the import action plan to be accessed.

getActionsClause

For listing actions in the import action plan.

retrieveActionClause

For displaying information of an action in the import action plan.

QUOTED_STRING

The name of the action to be accessed.

getPropertiesClause

For retrieving properties associated with the action. For the current release, there are no predefined properties for import actions.

Examples

```
OMBRETRIEVE_IMPORT_ACTION_PLAN 'PLAN1' GET ACTIONS
```

This command will list all the actions defined in action plan PLAN1.

OMBRETRIEVE IMPORT_ACTION_PLAN 'PLAN1' ACTION 'A1' GET REF SOURCE

This command will retrieve the set of items to be imported from source. A list is returned, which contains alternating type names and item names, for example, {TABLE SCOTT.T1 VIEW SCOTT2.V1}.

OMBRETRIEVE IMPORT_ACTION_PLAN 'PLAN1' ACTION 'A1' GET REF TARGET

The path of the target for import action A1 is retrieved. An example return value is TRANSPORTABLE_MODULE: /MY_PROJECT/TM101.

See Also

OMBCREATE IMPORT_ACTION_PLAN, OMBIMPORT

OMBRETRIEVE ITEM_FOLDER

Purpose

Retrieve details of an item folder.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveItemFolderCommand = OMBRETRIEVE ITEM_FOLDER "QUOTED_STRING" (
    "retrieveItemFolderClause" | "retrieveItemFolderItemClause" |
    "retrieveItemFolderJoinClause" | "retrieveItemFolderConditionClause" )
retrieveItemFolderClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getItemFolderSCOClaue" )
retrieveItemFolderItemClause = ITEM "QUOTED_STRING" GET (
    "getPropertiesClause" | SOURCE_OBJECTS | ( REF | REFERENCE ) DEFINING
    LISTS_OF_VALUES | ( REF | REFERENCE ) DEFINING ALTERNATIVE_SORT_ORDERS
    | ( REF | REFERENCE ) ORDERED ALTERNATIVE_SORT_ORDERS | ( REF |
    REFERENCE ) LIST_OF_VALUES | ( REF | REFERENCE ) DRILL_TO_DETAIL | (
    REF | REFERENCE ) ALTERNATIVE_SORT_ORDER | ( REF | REFERENCE )
    DRILL_LEVELS | ( REF | REFERENCE ) LOCAL JOINS | ( REF | REFERENCE )
    REMOTE JOINS )
retrieveItemFolderJoinClause = JOIN "QUOTED_STRING" (
    "retrieveItemFolderJoinComponentClause" | GET ( "getPropertiesClause"
    | JOIN_COMPONENTS | ( REF | REFERENCE ) FOREIGN_KEY | ( REF |
    REFERENCE ) ITEM_FOLDERS ) )
retrieveItemFolderConditionClause = CONDITION "QUOTED_STRING" GET
    "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getItemFolderSCOClaue = ITEMS | JOINS | CONDITIONS | SOURCE_OBJECTS | (
    REF | REFERENCE ) USING JOINS | ( REF | REFERENCE ) BUSINESS_AREAS | (
    REF | REFERENCE ) ROLE | ( REF | REFERENCE ) LEVEL | DEPENDENTS
retrieveItemFolderJoinComponentClause = JOIN_COMPONENT "QUOTED_STRING" GET
    ( "getPropertiesClause" | ( REF | REFERENCE ) LOCAL ITEM | ( REF |
    REFERENCE ) REMOTE ITEM )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveItemFolderCommand

To retrieve an item folder.

QUOTED_STRING

name of the item folder.

retrieveItemFolderClause

This clause retrieves the contents of an item folder.

GET

For an item folder, this clause retrieves the following

ITEMS retrieves the items in the item folder.

JOINS retrieves the joins in the item folder.

CONDITIONS retrieves the conditions in the item folder.

SOURCE_OBJECTS retrieves the objects the item folder is based on.

REF USING JOINS retrieves the join usages for the item folder.

REF BUSINESS_AREAS retrieves the business areas the item folder belongs to.

REF ROLE retrieves the dimension use reference for the item folder.

REF LEVEL retrieves the level reference for the item folder.

DEPENDENTS retrieves the dependents for the item folder.

retrieveItemFolderItemClause

Retrieves a specific item.

QUOTED_STRING

name of the item.

GET

For an item, this clause retrieves the following

SOURCE_OBJECT retrieves the object that the item is based on.

In the case of a complex item (that is, an expression) the list of items referenced by the current item

REF LIST_OF_VALUES retrieves the List of Values associated with the item.

REF DRILL_TO_DETAIL retrieves the Drill to Detail associated with the item.

REF ALTERNATIVE_SORT_ORDER retrieves the Alternative Sort Order associated with the item.

REF DEFINING LISTS_OF_VALUES retrieves a list of Lists of Values that use this Item to define their values.

REF DEFINING ALTERNATIVE_SORT_ORDERS retrieves a list of Alternative Sort Orders that use this Item to define their values.

REF ORDERED ALTERNATIVE_SORT_ORDERS retrieves a list of Alternative Sort Orders that use this Item to define their order.

REF DRILL LEVELS retrieves the list of Drill Levels that reference this Item.

REF LOCAL JOINS retrieves the list of Join Components that reference this Item as their local Item.

REF REMOTE JOINS retrieves the list of Join Components that reference this Item as their remote Item.

retrieveItemFolderJoinClause
Retrieves a specific join.

QUOTED_STRING
name of the join.

GET
For join this clause retrieves the following
JOIN_COMPONENTS retrieves the list of Join Components of this Join.
REF FOREIGN KEY retrieves the foreign key associated with this join.
REF ITEM_FOLDERS retrieves the list of complex Item Folders that use this Join to join their component Item Folders.

retrieveItemFolderConditionClause
Retrieves a specific condition.

QUOTED_STRING
name of the condition.

getPropertiesClause
Retrieves the properties of the object.

Basic properties for ITEM_FOLDER:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the item folder

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "

Description of the item folder

Name: EXTERNAL_TABLE_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The physical name for the corresponding table or view. This is automatically set if the Folder is associated with a Table

Name: VISIBLE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the item folder should be visible to the user

Name: FOLDER_TYPE

Type: STRING(40)

Valid Values: SIMPLE, COMPLEX

Default: "

The type of item folder

Basic properties for ITEM:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the item

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the item

Name: ALIGNMENT

Type: STRING(40)

Valid Values: GENERAL, LEFT, CENTER, RIGHT

Default: 'GENERAL'

The default alignment for displaying the item

Name: DISPLAY_CASE

Type: STRING(40)

Valid Values: GENERAL, LOWER, UPPER, INITCAPPED

Default: 'GENERAL'

How alphabetic characters should be displayed

Name: CASE_STORAGE

Type: STRING(40)

Valid Values: GENERAL, LOWER, UPPER, MIXED

Default: 'GENERAL'

How alphabetic characters are stored

Name: CONTENT_TYPE

Type: STRING(40)

Valid Values: No Value or FILE. For datatypes such as BLOB, it may contain a file extension such as DOC, AVI, WAV, JPG

Default: "

Details on whether the Item contains a file name or should be processed by an external application

Name: DEFAULT_AGGREGATE

Type: STRING(255)

Valid Values: Detail, AVG, COUNT, MAX, MIN, SUM

Default: 'SUM' when the datatype is Numeric, 'Detail' otherwise

Name of the default rollup function for the item

Name: DEFAULT_POSITION

Type: STRING(40)

Valid Values: MEASURE, TOP OR SIDE, TOP, SIDE, PAGE

Default: 'MEASURE' when the datatype is NUMBER or FLOAT, 'TOP OR SIDE' otherwise

Default position for the item

Name: REPLACE_NULL_WITH

Type: STRING(255)

Valid Values: N/A

Default: "

The value to be displayed for null values

Name: FORMULA

Type: STRING

Valid Values: N/A

Default: "

The text of the derivation expression for a derived item

Name: EXTERNAL_COLUMN_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The external name of the corresponding column. This is automatically set if the Item is associated with a Column

Name: FORMAT_MASK

Type: STRING(255)

Valid Values: N/A

Default: "

The display format mask for the item

Name: HEADING

Type: STRING(255)

Valid Values: N/A

Default: "

The displayed heading text for the item

Name: DATATYPE

Type: STRING(40)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.ROW_LCR, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE

TIMESTAMP WITH TIME ZONE, UNSPECIFIED, VARCHAR, VARCHAR2,
XMLTYPE,

SYS.XMLFORMAT, BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
SYS.XMLSEQUENCETYPE,

SYS_REFCURSOR, BLAST_MATCH_PLSQLRECORDTYPE

Default: 'VARCHAR2'

The datatype for the item

Name: VISIBLE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the item should be visible to the user

Name: MAX_CHAR_FETCHED

Type: Number

Valid Values: N/A

Default: "

The maximum number of characters fetched for an item

Name: DEFAULT_WIDTH

Type: Number

Valid Values: N/A

Default: "

The default number of characters to display

Name: WORD_WRAP

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether wordwrap is allowed in the display

Basic properties for JOIN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the join

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the join

Name: OUTER_JOIN_ON_MASTER

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether detail rows with no related master row should be included in the join

Name: OUTER_JOIN_ON_DETAIL

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether master rows with no related detail rows should be included in the join

Name: EXTERNAL_KEY_NAME

Type: STRING(255)

Valid Values: N/A

Default: "

The external name of the corresponding foreign key. This is automatically set if the Join is associated with a Foreign Key

Name: DETAIL_ALWAYS_HAS_MASTER

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether every detail row must reference a unique master row

Name: ONE_TO_ONE

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether a master row only ever has a single detail row

Basic properties for JOIN_COMPONENT:

Name: JOIN_OPERATOR

Type: STRING(200)

Valid Values: =, <>, <, <=, > or >=

Default: "

Business name of the join

Basic properties for CONDITION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the condition

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the condition

Name: MATCH_CASE

Type: BOOLEAN

Valid Values: Y,N

Default: 'Y'

Whether the case of alphabetic characters must match exactly

Name: FORMULA

Type: STRING

Valid Values: N/A

Default: "

The expression for the condition

Name: MANDATORY

Type: BOOLEAN

Valid Values: Y,N

Default: 'N'

Whether the Condition is optional or mandatory

Properties for ITEM_FOLDER:

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for the referenced database object

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Name: OPTIMIZER_HINT

Type: STRING

Valid Values: N/A

Default: "

Optimizer Hint to be added when this Item Folder is used in a query

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

getItemFolderSCOClause

Retrieves the contents of the item folder.

retrieveItemFolderJoinComponentClause

Retrieves the join components.

QUOTED_STRING

name of the condition.

GET

For join component this clause retrieves the following

REF LOCAL ITEM retrieves the local item used in the join.

REF REMOTE ITEM retrieves the remote item used in the join.

propertyNameList

This is the list of property names.

Examples

```
OMBRETRIEVE ITEM_FOLDER 'COST' GET PROPERTIES(DESCRIPTION)
```

See Also

OMBALTER ITEM_FOLDER, OMBCREATE ITEM_FOLDER

OMBRETRIEVE LIST_OF_VALUES

Purpose

Retrieve details of a List Of Values.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveListOfValuesCommand = OMBRETRIEVE LIST_OF_VALUES "QUOTED_STRING"
    "retrieveListOfValuesClauseDetails"
retrieveListOfValuesClauseDetails = GET (
    "getPropertiesClauseforLOVandD2D" | "getReferenceIconSetClause" | (
        REF | REFERENCE ) DEFINING ITEM | ( REF | REFERENCE ) ITEMS |
        DEPENDENTS )
getPropertiesClauseforLOVandD2D = PROPERTIES "(
    "propertyNameListforLOVandD2D" )"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyNameListforLOVandD2D = ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) {
    ", " ( "UNQUOTED_STRING" | DRILL_TO_DETAIL ) }
```

Keywords And Parameters

retrieveListOfValuesCommand

Retrieves the list of values.

QUOTED_STRING

name of the list of values.

retrieveListOfValuesClauseDetails

This clause retrieves the contents of a list of values.

GET

This clause retrieves the following

REF DEFINING ITEM retrieves the Item that holds the individual values for this list of values.

REF ITEMS retrieves the list of Items that use this list of values.

DEPENDENTS retrieves a list of Item Folders that the list of values depends on.

(This will return the Item Folder containing the Values Item).

getPropertiesClauseforLOVandD2D

This clause gets the properties of the object.

Basic properties for LIST_OF_VALUES:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the list of values

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the list of values

Name: DRILL_TO_DETAIL

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the list of values enables drilling between the item folders containing the items that use the list of values

Name: RETRIEVE_VALUES_GROUP_SIZE

Type: Number

Valid Values: N/A

Default: '100'

The number of rows to be fetched from the database at a time

Name: CACHE_VALUES

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the list of values should be cached in memory

Name: REQUIRE_SEARCH

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether search criteria should be requested

Name: SHOW_IN_NAVIGATOR

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be shown in the item navigator

Name: SORTED_DISTINCT

Type: Boolean

Valid Values: Y,N

Default: 'Y'

Whether the values should be displayed sorted with duplicates hidden

Properties for LIST_OF_VALUES:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts

to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

propertyNameListforLOVandD2D

This is the list of property names.

Examples

OMBRETRIEVE LIST_OF_VALUES 'COLORS' GET PROPERTIES(DESCRIPTION)

See Also

OMBALTER LIST_OF_VALUES, OMBCREATE LIST_OF_VALUES

OMBRETRIEVE LOCATION

Purpose

Retrieve details of the location.

Prerequisites

Can be in any context.

Syntax

```
retrieveLocationCommand = OMBRETRIEVE LOCATION "QUOTED_STRING" (
    "getPropertiesClause" | "getReferenceIconSetClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveLocationCommand

Retrieve details of the named location.

getPropertiesClause

Get specified properties of the location.

getReferenceIconSetClause

Get specified Icon Set.

propertyNameList

The names of the properties whose values you want to retrieve.

Properties for LOCATION:

Basic properties:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the location.

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the location.

Name: TYPE

Type: STRING

Valid Values:

'CONCURRENT_MANAGER'

'AUTOSYS_AGENT'

'AUTOSYS_INSTANCE'

'BIBEANS'

'DISCOVERER'

'FILE_SYSTEM'

'OEM_AGENT'

'ORACLE_DATABASE'

'ORACLE_GATEWAY'

'ORACLE_WORKFLOW'

'SAP'

'TRANSPORTABLE_MODULE_SOURCE'

'TRANSPORTABLE_MODULE_TARGET'

Default: N/A

The type of system the location represents.

Name: VERSION

Type: STRING

Valid Values:

for 'CONCURRENT_MANAGER' : '11i'

for 'AUTOSYS_AGENT' : '0'

for 'AUTOSYS_INSTANCE' : '0'

for 'BIBEANS' : '10.1'

for 'DISCOVERER' : '10.1'

for 'FILE_SYSTEM' : do not set version

for 'OEM_AGENT' : '9.0','9.2'

for 'ORACLE_DATABASE' : '8.1','9.0','9.2','10.1','10.2'

for 'ORACLE_GATEWAY' : do not set version

for 'ORACLE_WORKFLOW' : '2.6.2','2.6.3','2.6.4','11i'

for 'SAP' : '4.x','3.x'

for 'TRANSPORTABLE_MODULE_SOURCE' : '8.1','9.0','9.2','10.1','10.2'

for 'TRANSPORTABLE_MODULE_TARGET' : '8.1','9.0','9.2','10.1','10.2'

Default: N/A

The version of the system(s) the location represents.

Lists of available properties for different types of LOCATION:

for 'CONCURRENT_MANAGER' :

TYPE,PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA,VERSION,APPLICATION,APPLICATION_USER,RESPONSIBILITY

for 'AUTOSYS_AGENT':

TYPE,VERSION,PASSWORD,HOST

for 'AUTOSYS_INSTANCE':

TYPE,VERSION,USER (or USER_NAME),PASSWORD,INSTANCE

for 'BIBEANS':

TYPE,VERSION,USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME

for 'DISCOVERER':

TYPE,VERSION,USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME

for 'FILE_SYSTEM':

TYPE,USER (or USER_NAME),PASSWORD,HOST,ROOTPATH

for 'OEM_AGENT':

TYPE,USER (or USER_NAME),PASSWORD,VERSION,DOMAIN,AGENT

for 'ORACLE_DATABASE':

TYPE,VERSION,CONNECT_AS_USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,DATABASE_NAME,SCHEMA

for 'ORACLE_GATEWAY':

TYPE,CONNECT_AS_USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA

for 'ORACLE_WORKFLOW':

TYPE,VERSION,PASSWORD,HOST,PORT,SERVICE_NAME,NET_SERVICE_NAME,SCHEMA

for 'SAP':

TYPE, VERSION, USER (or USER_NAME), PASSWORD, APPLICATION_SERVER,

SYSTEM_NUMBER, CLIENT, LANGUAGE, HOST_LOGIN_USER, HOST_LOGIN_PASSWORD,

FTP_DIRECTORY, EXECUTION_FM

for 'TRANSPORTABLE_MODULE_SOURCE':

TYPE, VERSION,CONNECT_AS_USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME,FTP_USER,FTP_PASSWORD

for 'TRANSPORTABLE_MODULE_TARGET':

TYPE, VERSION,CONNECT_AS_USER (or

USER_NAME),PASSWORD,HOST,PORT,SERVICE_NAME

Some other properties for LOCATIONS:

Name: CONNECTION_TYPE

Type: STRING

Valid Values: 'HOST_PORT_SERVICE', 'SQL_NET_CONNECTION', 'DATABASE_LINK'

Default: 'HOST_PORT_SERVICE'

The location connection details format.

Name: HOST

Type: STRING

Valid Values: N/A

Default: N/A

The machine name.

Name: PORT

Type: NUMBER

Valid Values: 1 - 65535

Default: 1521

The port number of a database listener.

Name: SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The database service name.

Name: NET_SERVICE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The database netservice name.

Name: SCHEMA

Type: STRING

Valid Values: N/A

Default: N/A

The database schema name.

Name: PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: CONNECT_AS_USER

Synonym: USER_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: DOMAIN

Type: STRING

Valid Values: N/A

Default: N/A

The address of a machine running the Oracle Management Service.

Name: AGENT

Type: STRING

Valid Values: N/A

Default: N/A

The name of an Oracle Enterprise Manager (OEM) node running an OEM Agent.

This name must be entered exactly as shown under the nodes in the Oracle Management Service.

Name: ROOTPATH

Type: STRING

Valid Values: N/A

Default: N/A

The file system directory.

Name: APPLICATION

Type: STRING

Valid Values: N/A

Default: N/A

The Application name.

Name: APPLICATION_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name.

Name: DATABASE_NAME

Type: STRING

Valid Values: N/A

Default: N/A

The Data Base name.

Name: RESPONSIBILITY

Type: STRING

Valid Values: N/A

Default: N/A

The responsibility role.

Name: APPLICATION_SERVER

Type: STRING

Valid Values: N/A

Default: N/A

The application server.

Name: SYSTEM_NUMBER

Type: STRING

Valid Values: N/A

Default: N/A

The number of SAP system.

Name: CLIENT

Type: STRING

Valid Values: N/A

Default: N/A

The client.

Name: LANGUAGE

Type: STRING

Valid Values: N/A

Default: N/A

The language of SAP.

Name: HOST_LOGIN_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user.

Name: HOST_LOGIN_PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The password.

Name: EXECUTION_FM

Type: STRING

Valid Values: N/A

Default: N/A

RFC Function Module for remote ABAP report execution

Name: FTP_USER

Type: STRING

Valid Values: N/A

Default: N/A

The user name used for creating ftp connection.

Name: FTP_PASSWORD

Type: STRING

Valid Values: N/A

Default: N/A

The ftp password.

Name: FTP_DIRECTORY

Type: STRING

Valid Values: N/A

Default: N/A

The directory used in a ftp session

Note:

1. N/A means any valid character in supported character set.

2. " " represents an empty string

Examples

```
OMBRETRIEVE LOCATION 'A_LOCATION' GET PROPERTIES (TYPE, VERSION,  
HOST,
```

```
PORT, SERVICE_NAME, DESCRIPTION, BUSINESS_NAME)
```

This will retrieve the location "A_LOCATION"'s type, version, host, port,

service name, description, and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE MAPPING to OMBRETRIEVE VIEW

This chapter lists commands associated with OMBRETRIEVE in alphabetical order starting with OMBRETRIEVE MAPPING.

OMBRETRIEVE MAPPING

Purpose

Retrieve mapping details such as the number of operators and their connections.

Prerequisites

The current context must be in an Oracle Module.

Syntax

```

retrieveMappingCommand = OMBRETRIEVE MAPPING "mappingName" (
    "retrieveOperatorOwnerDetailClause" | "testConnectionClause" )
mappingName = "QUOTED_STRING"
retrieveOperatorOwnerDetailClause = GET ( PROPERTIES "propertyKeyList" |
    "getReferenceIconSetClause" | [ "operatorType" ] OPERATORS [
    "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "operatorLocator" "getOperatorDetailClause"
testConnectionClause = HAS CONNECTION FROM "mappableBottomUpLocator" TO
    "mappableBottomUpLocator"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
connectionConditionClause = CONNECTED ( FROM "mappableBottomUpLocator" |
    TO "mappableBottomUpLocator" )
childType = "UNQUOTED_STRING"
getChildDetailClause = ( "childType" "childName" )+ GET ( PROPERTIES
    "propertyKeyList" | "childType" )
operatorLocator = [ "pluggableMapLocator" ] OPERATOR "operatorName"
getOperatorDetailClause = GET ( PROPERTIES "propertyKeyList" | [
    "groupDirection" ] GROUPS [ "connectionConditionClause" ] |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause" | "groupLocator"
    "getGroupDetailClause"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
pluggableMapLocator = ( PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapLocator" ] )
operatorName = "QUOTED_STRING"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupLocator = GROUP "groupName"
getGroupDetailClause = GET ( PROPERTIES "propertyKeyList" | ATTRIBUTES [
    "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "attributeLocator" "getAttributeDetailClause"
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
pluggableMapName = "QUOTED_STRING"
groupName = "QUOTED_STRING"
attributeLocator = ATTRIBUTE "attributeName"
getAttributeDetailClause = GET ( PROPERTIES "propertyKeyList" |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
attributeName = "QUOTED_STRING"

```

Keywords And Parameters

retrieveMappingCommand

Retrieve the detail of a mapping such as how many mapping operators are there or which mapping operators are connected to each other.

mappingName

Name of the mapping.

retrieveOperatorOwnerDetailClause

Retrieve the desired detail of a mapping or a pluggable mapping.

testConnectionClause

Verify if there is a connection between mapping operators, mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDUPLICATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCAST, LCRSPLITTER, MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION, TRANSFORMATION, UNPIVOT, VIEW.

connectionConditionClause

List objects only if they are connected from or to objects specified in the connection condition.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

getChildDetailClause

Get the desired detail of a child object that belongs to the mapping, map variable, mapping operator, mapping group or mapping attribute.

operatorLocator

Location of a mapping operator.

getOperatorDetailClause

Get the desired detail of a mapping operator.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the operator

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the group

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the attribute

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Length of the attribute.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Precision of the attribute.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for MAPPING:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_LANGUAGE

Type: STRING

Valid Values: ABAP, PLSQL, SQLLOADER, UNDEFINED

Default: UNDEFINED

The language used when generating code for the mapping.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be

used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,

NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN
 Valid Values: true, false
 Default: false
 Description not available.

Name: IS_OPTIONAL
 Type: BOOLEAN
 Valid Values: true, false
 Default: false
 If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT
 Type: BOOLEAN
 Valid Values: true, false
 Default: false
 Description not available.

Name: LENGTH
 Type: NUMBER
 Valid Values: N/A
 Default: 0
 The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME
 Type: STRING
 Valid Values: N/A
 Default: "
 Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME
 Type: STRING
 Valid Values: N/A
 Default: "
 Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier.

Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,
NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,
NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,
NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,
NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,
NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,
NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1,
NA_EXTRA_10,
NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,
NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,
NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,
NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,
NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,
NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,
NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,

NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMP CORRECTED, NA_STREETCOMP MATCH, NA_STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE of SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | |=

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before

loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading

the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key

"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of

reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAS_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORD_CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT,

TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using

the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE
Type: STRING
Valid Values: N/A
Default: "
Description not available.

Name: ROW-BASED_ONLY
Type: BOOLEAN
Valid Values: true, false
Default: false
Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT
Type: STRING
Valid Values: N/A
Default: "
Row count

Name: ROW_COUNT_ENABLED
Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this

operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: ≥ 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

pluggableMapLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

operatorName

Name of a mapping operator.

groupDirection

Direction of a mapping group.

groupLocator

Location of a mapping group.

getGroupDetailClause

Get the desired detail of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

pluggableMapName

Name of the pluggable map.

groupName

Name of a mapping group.

attributeLocator

Location of a mapping attribute.

getAttributeDetailClause

Get the desired detail of a mapping attribute.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

attributeName

Name of a mapping attribute.

Examples

```
OMBRETRIEVE MAPPING 'MAP1' GET OPERATORS
```

```
OMBRETRIEVE MAPPING 'MAP1' GET VARIABLES
```

```
OMBRETRIEVE MAPPING 'MAP1' OPERATOR 'SRC1' GROUP 'INOUTGRP1'  
GET ATTRIBUTES CONNECTED TO OPERATOR 'target1'
```

```
OMBRETRIEVE MAPPING 'MAP1' OPERATOR 'SRC1'  
GET PROPERTIES (BUSINESS_NAME, DESCRIPTION)
```

```
OMBRETRIEVE MAPPING 'MAP1' VARIABLE 'LAST_CUST'  
GET PROPERTIES (BUSINESS_NAME, DATATYPE)
```

See Also

OMBRETRIEVE, OMBCREATE MAPPING, OMBALTER MAPPING, OMBDROP MAPPING

OMBRETRIEVE MATERIALIZED_VIEW

Purpose

To retrieve properties of a materialized view.

Prerequisites

In the context of an Oracle Module

Syntax

```

retrieveMaterializedViewCommand = OMBRETRIEVE MATERIALIZED_VIEW
  "QUOTED_STRING" ( "retrieveMaterializedViewClause" |
    "retrieveColumnClause" | "retrieveUkPkClause" | "retrieveFkClause" |
    "retrieveCheckConstraintClause" | "retrieveIndexConfigurationClause" |
    "retrievePartitionConfigurationClause" |
    "retrievePartitionKeyConfigurationClause" |
    "retrieveTemplateSubpartitionConfigurationClause" |
    "retrieveSubPartitionConfigurationClause" |
    "retrieveSubPartitionKeyConfigurationClause" |
    "retrieveDataRuleUsageClause" )
retrieveMaterializedViewClause = GET ( "getPropertiesClause" |
  "getReferenceIconSetClause" |
  "getMaterializedViewSCOrDependentsClause" )
retrieveColumnClause = COLUMN "QUOTED_STRING" GET "getPropertiesClause"
retrieveUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" GET (
  "getPropertiesClause" | COLUMNS )
retrieveFkClause = FOREIGN_KEY "QUOTED_STRING" GET ( "getPropertiesClause"
  | COLUMNS | UNIQUE_KEY | PRIMARY_KEY | REFERENCED_KEY )
retrieveCheckConstraintClause = CHECK_CONSTRAINT "QUOTED_STRING" GET
  "getPropertiesClause"
retrieveIndexConfigurationClause = INDEX "QUOTED_STRING" ( GET (
  "getConfigurationPropertiesClause" | INDEX_COLUMNS | INDEX_PARTITIONS
  | INDEX_PARTITION_KEYS ) | ( INDEX_COLUMN "QUOTED_STRING" GET
  "getConfigurationPropertiesClause" ) | ( INDEX_PARTITION_KEY
  "QUOTED_STRING" GET "getConfigurationPropertiesClause" ) | (
  INDEX_PARTITION "QUOTED_STRING" GET "getConfigurationPropertiesClause"
  ) )
retrievePartitionConfigurationClause = PARTITION "QUOTED_STRING" GET
  "getConfigurationPropertiesClause"
retrievePartitionKeyConfigurationClause = PARTITION_KEY "QUOTED_STRING"
  GET "getConfigurationPropertiesClause"
retrieveTemplateSubpartitionConfigurationClause = TEMPLATE_SUBPARTITION
  "QUOTED_STRING" GET "getConfigurationPropertiesClause"
retrieveSubPartitionConfigurationClause = SUBPARTITION "QUOTED_STRING" OF
  PARTITION "QUOTED_STRING" GET "getConfigurationPropertiesClause"
retrieveSubPartitionKeyConfigurationClause = SUBPARTITION_KEY
  "QUOTED_STRING" GET "getConfigurationPropertiesClause"
retrieveDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" ( GET (
  "getPropertiesClause" | GROUPS ) | GROUP "QUOTED_STRING" ( GET (
  "getPropertiesClause" | ATTRIBUTES | REF ( TABLE | VIEW |
  MATERIALIZED_VIEW | EXTERNAL_TABLE ) ) | ATTRIBUTE "QUOTED_STRING" (
  GET ( "getPropertiesClause" | REF COLUMN ) ) ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getMaterializedViewSCOrDependentsClause = COLUMNS | UNIQUE_KEYS |
  PRIMARY_KEY | FOREIGN_KEYS | CHECK_CONSTRAINTS | COLUMN AT POSITION
  "INTEGER_LITERAL" | INDEXES | PARTITIONS | PARTITION_KEYS |

```

```
SUBPARTITION_KEYS | DATA_RULE_USAGES | ( REF | REFERENCE ) ( TABLES |  
VIEWS | MATERIALIZED_VIEWS )  
getConfigPropertiesClause = PROPERTIES "(" "propertyNameList" ")"  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveMaterializedViewCommand

This clause retrieves a materialized view.

QUOTED_STRING

name of the materialized view.

retrieveMaterializedViewClause

This clause retrieves a properties of a materialized view.

retrieveColumnClause

This clause will retrieve columns.

QUOTED_STRING

Name of the column.

retrieveUkPkClause

This clause will retrieve a unique key or primary key.

QUOTED_STRING

Name of the unique key or the primary key.

retrieveFkClause

This clause will retrieve a key referenced by a foreign key, either a unique key or primary key. Use REFERENCED_KEY to retrieve the referenced key for a foreign key regardless of the type of referenced key (unique or primary).

QUOTED_STRING

Name of the foreign key.

retrieveCheckConstraintClause

This clause gets the check constraint.

QUOTED_STRING

Name of the check constraint.

retrieveIndexConfigurationClause

Gets the index in this clause.

QUOTED_STRING

Name of the index.

retrievePartitionConfigurationClause

Gets the partition.

QUOTED_STRING

The partition name.

retrievePartitionKeyConfigurationClause

This clause gets the partition key.

QUOTED_STRING

The name of the partition key.

retrieveDataRuleUsageClause

This clause retrieves the data rule usages.

QUOTED_STRING

Name of data rule usage, group or attribute.

GROUPS

Retrieve the names of all relation groups in the data rule usage.

ATTRIBUTES

Retrieve the names of all attributes in a data rule usage group.

TABLE

Table name associated with the data rule usage group.

VIEW

View name associated with the data rule usage group.

MATERIALIZED_VIEW

Materialized view name associated with the data rule usage group.

EXTERNAL_TABLE

External table name associated with the data rule usage group.

COLUMN

Column name associated with the data rule usage group attribute.

getPropertiesClause

This clause retrieves all the properties.

Note:

Constraints can be specified but will not be generated for either View or Materialized View in this release.

Basic properties for VIEW, MATERIALIZED_VIEW:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the View, MaterializedView

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the View, MaterializedView

Name: VIEW_QUERY

Type: STRING(4000)

Valid Values: N/A

Default: "

Sets the query definition in View and MaterializedView.

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Default value of the column

Name: NOT_NULL

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify "true" to enforce Not Null restriction on a column.

Basic properties for PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the table rows according to a Hash Algorithm, lists of values, or specified ranges.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH partitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of partitions that is a power of 2. If you have multiple Partition Keys, you only have to specify once.

Basic properties for PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Commas can be escaped using "" (for example, '1,2,3'). Always specify DEFAULT as the value of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH customized subpartitions the database should create for a particular main RANGE partition (RANGE-HASH BY QUANTITY partitioning). For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once. Set it to 0 to reverse to the use of generic template HASH_QUANTITY specified in SUBPARTITION_KEY.

Basic properties for SUBPARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

For partition-level partitioning according to a Hash Algorithm or lists of values. Each partition is further sorted into subpartitions.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH subpartitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once.

Basic properties for SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST subpartition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any SUBPARTITION. Each LIST subpartition must have at least one value. No value, including NULL, can appear in more than one subpartition.

Basic properties for TEMPLATE_SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

In composite partitioning, template subpartitions are automatically applied to those partitions without their subpartitions specified. Here for LIST subpartitions only, specify a comma-delimited, ordered list of literal values corresponding to the LIST subpartitioning column. Always specify DEFAULT as the value of the last template LIST subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any TEMPLATE_SUBPARTITION. Each LIST template subpartition must have at least one value. No value, including NULL, can appear in more than one template subpartition.

Basic properties for INDEX:

Name: INDEX_TYPE

Type: STRING

Valid Values: UNIQUE, NON-UNIQUE, BITMAP, FUNCTION-BASED

Default: (No default, must be one of the preceding choices)

Specify the type of an index. NORMAL can be used in place of NON-UNIQUE.

Name: LOCAL_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify if an index is Global or Local. The default is Global.

Specify Local so that the index is partitioned on the same columns, with the same number of partitions and the same partition bounds as table.

Oracle Database automatically maintains local index partitioning as the underlying table is repartitioned.

Name: COLUMN_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

Specify an expression built from columns of table, constants, SQL functions, and user-defined functions to create a FUNCTION-BASED index.

Basic properties for INDEX_PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the index rows according to a Hash Algorithm, lists of values, or specified ranges. Hash index partitioning is supported starting with Oracle Database 10g version.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH index partitions the database should create based on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of index partitions that is a power of 2. If you have multiple index Partition Keys, you only have to specify once.

Basic properties for INDEX_PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition. No need to specify VALUES_LESS_THAN for Local index.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last partition. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition. No need to specify VALUES_EQUAL_TO for Local index.

Basic properties for CHECK_CONSTRAINTS:

Name: CHECK_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Specify a condition that each row in the table must satisfy.

Properties for MATERIALIZED_VIEW:

Name: BASE_TABLES

Type: STRING

Valid Values: N/A

Default: "

Specify a comma separated list of base tables for generating materialized view log.

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for the database object. The default is DEFAULT.

Name: BUILD

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE, PREBUILT

Default: "

Specify IMMEDIATE to populate the view when it is created. Specify DEFERRED to delays population until the next refresh operation. IMMEDIATE is the default.

Name: CONSTRAINTS

Type: STRING

Valid Values: , ENFORCED, TRUSTED

Default: "

Specify TRUSTED to let Oracle Database use dimension and constraint information that has been declared trustworthy by the database administrator but that has not been validated by the database. If the dimension and constraint information is valid, then performance may improve. However, if this information is invalid, then the refresh procedure may corrupt the materialized view even though it returns a success status. ENFORCED is the default.

Name: DEFAULTINDEXBUFFERPOOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for the database object. The default is DEFAULT.

Name: DEFAULTINDEXFREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: DEFAULTINDEXFREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the database object. The default is 1.

Name: DEFAULTINDEXINITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: DEFAULTINDEXINITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2.

Name: DEFAULTINDEXMAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the database object.

Name: DEFAULTINDEXMAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: DEFAULTINDEXMINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: DEFAULTINDEXNEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: DEFAULTINDEXPCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: DEFAULT_INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Specify tablespace for default index storage.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FOR_UPDATE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES (FOR UPDATE) to allow a subquery, primary key, object, or rowid materialized view to be updated. When used in conjunction with Advanced Replication, these updates will be propagated to the master. The default is NO.

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the database object. The default is 1.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: HASH_PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

A comma separated list of tablespaces to use for [sub]partition storage.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the database object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can

update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: NEXTDATE

Type: STRING

Valid Values: N/A

Default: "

Specify a datetime expression for calculating the interval between automatic refreshes.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Specify NOPARALLEL for serial execution. This is the default. Specify PARALLEL if you want Oracle to select a degree of parallelism equal to the number of CPUs available on all participating instances times the value of the PARALLEL_THREADS_PER_CPU initialization parameter.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Specify the number of parallel threads used in the parallel operation.

Normally Oracle calculates the optimum degree of parallelism, so it is not

necessary for you to specify it.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: QUERY_REWRITE

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE to mark the materialized view eligible for query rewrite or DISABLE to mark the materialized view ineligible for query rewrite. DISABLE is the default.

Name: REFRESH

Type: STRING

Valid Values: , COMPLETE, FAST, FORCE, NEVER

Default: "

Specify FAST to indicate the incremental refresh method. Specify COMPLETE to indicate the complete refresh method, which is implemented by executing the defining query of the materialized view. Specify FORCE to indicate that when a refresh occurs, Oracle Database will perform a fast refresh if one is possible or a complete refresh otherwise. FORCE is the default. Specify NEVER to prevent the materialized view from being refreshed with any Oracle Database refresh mechanism or packaged procedure.

Name: REFRESH_ON

Type: STRING

Valid Values: , COMMIT, DEMAND

Default: "

Specify COMMIT to indicate that a fast refresh is to occur whenever the database commits a transaction that operates on a master table of the materialized view. Specify DEMAND to indicate that the materialized view will be refreshed on demand by calling one of the three DBMS_MVIEW refresh procedures. DEMAND is the default.

Name: ROLLBACK

Type: STRING

Valid Values: , DEFAULT, DEFAULT LOCAL, DEFAULT MASTER, NONE

Default: DEFAULT LOCAL

Specify DEFAULT for Oracle Database to choose automatically which rollback segment to use. Specify DEFAULT MASTER for the remote rollback segment to be used at the remote master site for the individual materialized view.

Specify DEFAULT LOCAL for the remote rollback segment to be used for the local refresh group that contains the materialized view. DEFAULT LOCAL is the default. Specify NONE to name both master and local rollback segments.

Name: ROLLBACKSEGMENTLOCAL

Type: STRING

Valid Values: N/A

Default: "

Specify a named remote rollback segment to be used for the local refresh group that contains the materialized view. Default is null. Ignore if DEFAULT or DEFAULT LOCAL is specified for default rollback segment.

Name: ROLLBACKSEGMENTMASTER

Type: STRING

Valid Values: N/A

Default: "

Specify a named remote rollback segment to be used at the remote master site for the individual materialized view. Default is null. Ignore if DEFAULT or DEFAULT MASTER is specified for default rollback segment.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: STARTWITH

Type: STRING

Valid Values: N/A

Default: "

Specify a datetime expression for the first automatic refresh time.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: USING_INDEX_MODE

Type: STRING

Valid Values: , USING_INDEX, USING_NO_INDEX

Default: "

Specify USING_NO_INDEX to suppress the creation of the default index for

Materialized View. You can create an alternative index for a Materialized View explicitly. The default is USING_INDEX.

Name: WITH

Type: STRING

Valid Values: , PRIMARY_KEY, ROWID

Default: "

Specify PRIMARY KEY to create a primary key materialized view. Specify ROWID to create a rowid materialized view. Rowid materialized views are useful if the materialized view does not include all primary key columns of the master tables. Rowid materialized views must be based on a single table and meet other restrictions. PRIMARY KEY is the default.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for INDEX:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: COMPUTESTATISTICS

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to collect statistics at relatively little cost during the creation of an index.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: INDEXORDER

Type: STRING

Valid Values: , ASC, DESC

Default: "

Use ASC or DESC to indicate whether the index should be created in ascending or descending order. The Oracle default is ASC.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2 for Index.

Name: KEYCOMPRESS

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Specify COMPRESS to enable key compression.

Name: KEYCOMPRESSPREFIXLENGTH

Type: STRING

Valid Values: N/A

Default: "

Specify the prefix length (number of prefix columns to compress). For unique indexes, the valid range of prefix length values is from 1 to the number of key columns minus 1. The default prefix length is the number of key columns minus 1. For nonunique indexes, the valid range of prefix length values is from 1 to the number of key columns. The default prefix length is the number of key columns. Oracle compresses only nonpartitioned indexes that are nonunique or unique indexes of at least two columns. You cannot specify COMPRESS for a bitmap index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (2-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: ONLINE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to indicate that DML operations on the table will be allowed during creation of the index.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of Index tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition

overflow Index tablespaces. The number of Index tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of Index tablespaces, then Oracle cycles through the names of the Index tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for a locally partitioned index. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. If specified, then individual local Hash index partitions specified will be ignored for Local Hash or Range-Hash Index partitioning.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: SORT

Type: STRING

Valid Values: , NOSORT, REVERSE, SORT

Default: "

Specify NOSORT to indicate to Oracle that the rows are already stored in the database in ascending order. Specify REVERSE to store the bytes of the index block in reverse order, excluding the rowid.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PARTITION:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The

default is DEFAULT.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1 for Table and 2 for Index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getMaterializedViewSCOorDependentsClause

This clause will retrieves materialized view components like columns, keys, and so on, or relational objects that this materialized view have referential dependency on.

getConfigurationPropertiesClause

This clauses gets the configuration properties of the object.

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE MATERIALIZED_VIEW 'NEW_MATERIALIZED_VIEW' GET  
PROPERTIES
```

```
(DESCRIPTION, BUSINESS_NAME)
```

This will retrieve its description and business name.

See Also

```
OMBRETRIEVE, OMBCREATE MATERIALIZED_VIEW, OMBALTER  
MATERIALIZED_VIEW, OMBDROP MATERIALIZED_VIEW
```

OMBRETRIEVE MDL_ACTION_PLAN

Purpose

Retrieve the details of an existing metadata loader action plan.

Prerequisites

Connection must be established to the repository.

Syntax

```
retrieveMDLActionPlanCommand = OMBRETRIEVE ( MDL_ACTION_PLAN )  
    "QUOTED_STRING" ( "getActionsClause" | "retrieveActionClause" )  
getActionsClause = GET ACTIONS  
retrieveActionClause = ACTION "QUOTED_STRING" GET "getReferenceClause"  
getReferenceClause = ( REF | REFERENCE )
```

Keywords And Parameters

retrieveMDLActionPlanCommand

Retrieve the details of an existing metadata loader action plan.

getActionsClause

Get a list of actions from an action plan.

retrieveActionClause

Retrieve a set of references of an action.

getReferenceClause

Retrieve the object type and its references for an action.

Examples

```
OMBRETRIEVE MDL_ACTION_PLAN 'MY_ACTION_PLAN' GET ACTIONS
```

```
OMBRETRIEVE MDL_ACTION_PLAN 'MULTI_PROJECT_ACTION_PLAN'  
ACTION 'MULTI_PROJ' GET REFERENCE
```

See Also

OMBCREATE MDL_ACTION_PLAN, OMBALTER MDL_ACTION_PLAN,
OMBDROP MDL_ACTION_PLAN, OMBEXPORT MDL_FILE

OMBRETRIEVE MINING_MODEL

Purpose

Retrieve the mining model details the case id, target attribute, operators in the build map and their connections

Prerequisites

The current context must be in an Oracle Module.

Syntax

```

retrieveMiningModelCommand = OMBRETRIEVE MINING_MODEL "miningModelName"
    "retrieveMiningDetailClause"
miningModelName = "QUOTED_STRING"
retrieveMiningDetailClause = GET ( PROPERTIES "propertyKeyList" |
    MINING_FUNCTION | MINING_ALGORITHM | CASE_ID_COLUMN | TARGET_COLUMN |
    [ "operatorType" ] OPERATORS [ "connectionConditionClause" ] |
    "childType" ) | "getChildDetailClause" | "operatorLocator"
    "getOperatorDetailClause"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
connectionConditionClause = CONNECTED ( FROM "mappableBottomUpLocator" |
    TO "mappableBottomUpLocator" )
childType = "UNQUOTED_STRING"
getChildDetailClause = ( "childType" "childName" )+ GET ( PROPERTIES
    "propertyKeyList" | "childType" )
operatorLocator = [ "pluggableMapLocator" ] OPERATOR "operatorName"
getOperatorDetailClause = GET ( PROPERTIES "propertyKeyList" | [
    "groupDirection" ] GROUPS [ "connectionConditionClause" ] |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause" | "groupLocator"
    "getGroupDetailClause"
propertyKey = "UNQUOTED_STRING"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
childName = "QUOTED_STRING"
pluggableMapLocator = ( PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapLocator" ] )
operatorName = "QUOTED_STRING"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupLocator = GROUP "groupName"
getGroupDetailClause = GET ( PROPERTIES "propertyKeyList" | ATTRIBUTES [
    "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "attributeLocator" "getAttributeDetailClause"
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
pluggableMapName = "QUOTED_STRING"
groupName = "QUOTED_STRING"
attributeLocator = ATTRIBUTE "attributeName"
getAttributeDetailClause = GET ( PROPERTIES "propertyKeyList" |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
attributeName = "QUOTED_STRING"

```

Keywords And Parameters

propertyKeyList

The list of property keys.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CUBE, DATA_GENERATOR,

DEDUPLICATOR, DIMENSION, EXPRESSION, EXTERNAL_PROCESS, EXTERNAL_TABLE,

FILTER, FLAT_FILE, INPUT_PARAMETER, JOINER, KEY_LOOKUP, LCRCAST,

LCRSPLITTER, MATCHMERGE, MATERIALIZED_VIEW, NAME_AND_ADDRESS,

OUTPUT_PARAMETER, PIVOT, POSTMAPPING_PROCESS, PREMAPPING_PROCESS, SEQUENCE,

SET_OPERATION, SORTER, SPLITTER, TABLE, TRANSFORMATION, UNPIVOT, VIEW.

connectionConditionClause

List objects only if they are connected from or to objects specified in the connection condition.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

getChildDetailClause

Get the desired detail of a child object that belongs to the mapping, map variable, mapping operator, mapping group or mapping attribute.

operatorLocator

Location of a mapping operator.

getOperatorDetailClause

Get the desired detail of a mapping operator.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the attribute

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the attribute

Name: DATATYPE
Type: STRING(20)
Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARHCAR, VARCHAR2, XMLTYPE
Default: "
Datatype of the Attribute

Name: LENGTH
Type: NUMBER
Valid Values: N/A
Default: 0
Length of the attribute.

Name: PRECISION

Type: NUMBER
Valid Values: N/A
Default: 0
Precision of the attribute.

Name: SCALE
Type: NUMBER
Valid Values: N/A
Default: 0
Scale of the attribute.

Properties for MINING_MODEL:

Name: GENERATION_COMMENTS
Type: STRING
Valid Values: N/A
Default: "
Enter additional comments for the generated code.

Name: SETTINGS_TABLE_NAME
Type: STRING
Valid Values: N/A
Default: "
Name of table which stores the settings for model build.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

pluggableMapLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

operatorName

Name of a mapping operator.

groupDirection

Direction of a mapping group.

groupLocator

Location of a mapping group.

getGroupDetailClause

Get the desired detail of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

pluggableMapName

Name of the pluggable map.

groupName

Name of a mapping group.

attributeLocator

Location of a mapping attribute.

getAttributeDetailClause

Get the desired detail of a mapping attribute.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

attributeName

Name of a mapping attribute.

Examples

```
OMBRETRIEVE MINING_MODEL 'MODEL1' GET MINING_ALGORITHM
```

See Also

OMBRETRIEVE, OMBCREATE MINING_MODEL, OMBALTER MINING_MODEL, OMBDROP MINING_MODEL

OMBRETRIEVE NESTED_TABLE

Purpose

Retrieve details of the Nested Table.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
retrieveNestedTableCommand = OMBRETRIEVE NESTED_TABLE "QUOTED_STRING" (  
    GET ( "getPropertiesClause" | "getReferenceIconSetClause" ) )  
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"  
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET  
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveNestedTableCommand

Retrieves the details of the Nested Table with the given name.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for NESTED_TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Nested Table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Nested Table

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Base Element of the Nested Table

Properties for NESTED_TABLE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE NESTED_TABLE 'SOME_NESTED_TABLE' GET PROPERTIES
(DATATYPE)
```

This will retrieve the Nested Table "SOME_NESTED_TABLE"'s base element datatype.

See Also

OMBRETRIEVE

OMBRETRIEVE OBJECT_TYPE

Purpose

Retrieve details of the Object Type.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
retrieveObjectTypeCommand = OMBRETRIEVE OBJECT_TYPE "QUOTED_STRING" (
    "retrieveObjectTypeClause" | "retrieveAttributeClause" )
retrieveObjectTypeClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getAttributesClause" )
retrieveAttributeClause = OBJECT_TYPE_ATTRIBUTE "QUOTED_STRING" GET
    "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getAttributesClause = OBJECT_TYPE_ATTRIBUTES
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveObjectTypeCommand

Retrieves the details of the Object Type with the given name.

retrieveObjectTypeClause

Gets the properties or the Attribute names.

retrieveAttributeClause

Gets the properties of the Attribute with the given name.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for OBJECT_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Object Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Object Type

Basic properties for OBJECT_TYPE_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Properties for OBJECT_TYPE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getAttributesClause

Gets the names of all the Attributes of the Object Type with the given name.

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE OBJECT_TYPE 'SOME_OBJECT_TYPE' OBJECT_TYPE_ATTRIBUTE  
'ATTR1'
```

```
GET PROPERTIES (DATATYPE)
```

This will retrieve the Object Type "SOME_OBJECT_TYPE"'s attribute "ATTR1"'s datatype.

See Also

OMBRETRIEVE, OMBALTER OBJECT_TYPE, OMBCREATE OBJECT_TYPE,
OMBDROP OBJECT_TYPE

OMBRETRIEVE ORACLE_MODULE

Purpose

Retrieve details of the Oracle module.

Prerequisites

Should be in the context of project.

Syntax

```
retrieveOracleModuleCommand = OMBRETRIEVE ORACLE_MODULE "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferenceLocationClause" |  
    "getReferenceDefaultLocationClause" |  
    "getReferenceMetadataLocationClause" | "getReferenceIconSetClause" |  
    "getReferenceLocationsClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION  
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT  
    LOCATION  
getReferenceMetadataLocationClause = GET ( REF | REFERENCE )  
    METADATA_LOCATION  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveOracleModuleCommand

This command retrieves the details of an Oracle Module

QUOTED_STRING

Name of the existing Oracle module or path to the Oracle module.

getPropertiesClause

Retrieve a set of properties that is associated with an Oracle Module.

Basic properties for ORACLE_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of an Oracle Module

Name: DESCRIPTION

Type: STRING(4000)
Valid Values: N/A
Default: "
Description of an Oracle Module

Name: MODULE_TYPE
Type: STRING
Valid Values: N/A
Default: N/A
Type of oracle module. Supported values are: 'WAREHOUSE_TARGET',
'DATA_SOURCE'.By default, it is 'WAREHOUSE_TARGET'.

Properties for ORACLE_MODULE:

Name: ABAP_DIRECTORY
Type: STRING
Valid Values: N/A
Default: abap\
Location where SAP data is dumped as flat files

Name: ABAP_EXTENSION
Type: STRING
Valid Values: N/A
Default: .abap
File name extension for ABAP scripts

Name: ABAP_RUN_PARAMETER_FILE
Type: STRING
Valid Values: N/A
Default: _run.ini
Run Parameter File Suffix for the parameter script in a ABAP job.

Name: ABAP_SPOOL_DIRECTORY
Type: STRING
Valid Values: N/A
Default: abap\log\
Location where ABAP scripts are buffered during script generation

processing.

Name: APPLICATION_SHORT_NAME

Type: STRING

Valid Values: N/A

Default: WB

Application Short Name

Name: ARCHIVE_DIRECTORY

Type: STRING

Valid Values: N/A

Default: archive\

Archive Directory

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

If this is a source module, this value indicates the location from which data will be read. If this is a target warehouse module, this value indicates the location where generated code will be deployed to and/or where data will be written to.

Name: DDL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ddl\

Location where scripts for database objects for the target schema are stored.

Name: DDL_EXTENSION

Type: STRING

Valid Values: N/A

Default: .ddl

File name extension for DDL scripts.

Name: DDL_SPOOL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ddl\log\

Location where DDL scripts are buffered during script generation processing.

Name: DEFAULT_INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Default name of tablespace to install indexes into.

Name: DEFAULT_OBJECT_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Default name of tablespace to install objects into.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: END_OF_LINE

Type: STRING

Valid Values: N/A

Default: \r\n

End of Line

Name: INPUT_DIRECTORY

Type: STRING

Valid Values: N/A

Default: input\

Input Directory

Name: INVALID_DIRECTORY

Type: STRING

Valid Values: N/A

Default: invalid\

Directory for SQL*Loader errors and rejected records

Name: LIB_DIRECTORY

Type: STRING

Valid Values: N/A

Default: lib\

LIB Directory

Name: LIB_EXTENSION

Type: STRING

Valid Values: N/A

Default: .lib

LIB Extension

Name: LIB_SPOOL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: lib\log\

LIB Spool Directory

Name: LOADER_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ctl\

Location where control files are stored.

Name: LOADER_EXTENSION

Type: STRING

Valid Values: N/A

Default: .ctl

Suffix for the loader scripts

Name: LOADER_RUN_PARAMETER_FILE

Type: STRING
Valid Values: N/A
Default: _run.ini
Suffix for the parameter initialization file.

Name: LOG_DIRECTORY
Type: STRING
Valid Values: N/A
Default: log\
Log Directory for the SQL*Loader

Name: MAIN_APPLICATION_SHORT_NAME
Type: STRING
Valid Values: N/A
Default: ora
Main Application Short Name

Name: PLSQL_DIRECTORY
Type: STRING
Valid Values: N/A
Default: pls\
Location where PL/SQL scripts are stored.

Name: PLSQL_EXTENSION
Type: STRING
Valid Values: N/A
Default: .pls
File name extension for PL/SQL scripts.

Name: PLSQL_GENERATION_MODE
Type: STRING
Valid Values: Default, Oracle Database 10g, Oracle Database 10gR2, Oracle Database 8i, Oracle Database 9i
Default: Default
Generation mode controls validation and generation for version specific features.

Name: PLSQL_RUN_PARAMETER_FILE

Type: STRING

Valid Values: N/A

Default: _run.ini

Suffix for the parameter script in a PL/SQL job.

Name: PLSQL_SPOOL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: pls\log\

Location where PL/SQL scripts are buffered during script generation processing.

Name: RECEIVE_DIRECTORY

Type: STRING

Valid Values: N/A

Default: receive\

Receive Directory

Name: SORT_DIRECTORY

Type: STRING

Valid Values: N/A

Default: sort\

Sort Directory

Name: STREAMS_ADMINISTRATOR_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location corresponding to the Streams Administrator

Name: TCL_DIRECTORY

Type: STRING

Valid Values: N/A

Default: tcl\

Location for TCL scripts that are generated after registration with Oracle Enterprise Manager

Name: TOP_DIRECTORY

Type: STRING

Valid Values: N/A

Default: ..\..\codegen\

Top Directory where generated code will get stored

Name: WORK_DIRECTORY

Type: STRING

Valid Values: N/A

Default: work\

Work Directory

Note:

1. N/A means any valid character in supported character set.

2. " " represents an empty string

getReferenceLocationClause

Retrieve the name of the runtime location referenced by this Oracle module.

getReferenceDefaultLocationClause

Retrieve the default runtime location referenced by this Oracle module.

getReferenceIconSetClause

Retrieve the icon set referenced by this Oracle module.

getReferenceLocationsClause

Retrieve the runtime locations referenced by this Oracle module.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE ORACLE_MODULE 'src_module' GET PROPERTIES
(DESCRIPTION,
BUSINESS_NAME)
```

This will retrieve the Oracle module "src_module"'s description and

business name.

See Also

OMBRETRIEVE

OMBRETRIEVE PACKAGE

Purpose

Retrieve details of the Package.

Prerequisites

Should be in the context of a Oracle Module or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations or WB_PREDEFINED_TRANS for Predefined Transformations.

WB_CUSTOM_TRANS may be modified by an administrator. WB_PREDEFINED_TRANS

may not be modified.

WB_CUSTOM_TRANS and WB_PREDEFINED_TRANS are not dependent on any project.

Syntax

```
retrievePackageCommand = OMBRETRIEVE PACKAGE "QUOTED_STRING" (
    "getPropertiesClause" | "getRelationalDependentsClause" |
    "getReferenceIconSetClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getRelationalDependentsClause = GET ( REF | REFERENCE ) ( TABLES | VIEWS |
    MATERIALIZED_VIEWS | SEQUENCES | FUNCTIONS | PROCEDURES | PACKAGES )
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrievePackageCommand

This command retrieves the details of a Package

QUOTED_STRING

Name of the existing Package or path to the Package.

getPropertiesClause

Used to get properties (core, user-defined) for packages. Valid properties are as shown:

Basic properties for PACKAGE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the package

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the package

Name: PACKAGE_SPEC

Type: STRING

Valid Values: N/A

Default: "

Retrieves the Package Spec of Imported Package

Name: PACKAGE_BODY

Type: STRING

Valid Values: N/A

Default: "

Retrieves the Package Body of Imported Package

Name: IS_IMPORTED

Type: BOOLEAN

Valid Values: N/A

Default: "

Retrieves 'true' if the Package is Imported otherwise 'false'

Properties for PACKAGE:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the package with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getRelationalDependentsClause

This clause retrieves referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE PACKAGE 'package_1' GET PROPERTIES (DESCRIPTION, UOID,  
BUSINESS_NAME,)
```

This will retrieve the Package "package_1's description, uoid, and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE PLSQL_RECORD_TYPE

Purpose

Retrieve details of the PLSQL Record Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
retrievePlSqlRecordTypeCommand = OMBRETRIEVE PLSQL_RECORD_TYPE
    "QUOTED_STRING" ( "retrievePlSqlRecordTypeClause" |
    "retrievePlSqlRecordAttributeClause" )
retrievePlSqlRecordTypeClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getPlSqlRecordAttributesClause" )
retrievePlSqlRecordAttributeClause = ATTRIBUTE "QUOTED_STRING" GET
    "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getPlSqlRecordAttributesClause = ATTRIBUTES
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrievePlSqlRecordTypeCommand

Retrieves the details of the PLSQL Record Type with the given name.

retrievePlSqlRecordTypeClause

Gets the properties or the Attribute names.

retrievePlSqlRecordAttributeClause

Gets the properties of the Attribute with the given name.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for PLSQL_RECORD_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the PLSQL Record Type

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the PLSQL Record Type

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME
Type: STRING(200)
Valid Values: N/A
Default: "
Business name of the Attribute

Name: DESCRIPTION
Type: STRING(4000)
Valid Values: N/A
Default: "
Description of the Attribute

Name: DATATYPE
Type: STRING(20)
Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARHCAR, VARCHAR2, XMLTYPE
Default: "
Datatype of the Attribute

Properties for PLSQL_RECORD_TYPE:

Name: GENERATION_COMMENTS
Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE,
NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME,
NA_LASTLINE,
NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_
LINE10, NA_LINE2,
NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,
NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_
NEIGHBORHOOD, NA_NONE,
NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE,
NA_POSTNAME,
NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE
Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_
FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN
Valid Values: true, false
Default: false
Description not available.

Name: LENGTH
Type: NUMBER
Valid Values: N/A
Default: 0
The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME
Type: STRING
Valid Values: N/A
Default: "
Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME
Type: STRING
Valid Values: N/A
Default: "
Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME
Type: STRING
Valid Values: N/A
Default: "
The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME
Type: STRING
Valid Values: N/A
Default: "
The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY, NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,

NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
 NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_
 EXTRA_7,
 NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_
 CODE,
 NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_
 FIRSTNAMESTD,
 NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_
 INSTALLATIONTYPE,
 NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_
 ISGOODGROUP,
 NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
 NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME,
 NA_LOCALITY_2,
 NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_
 ORDER, NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_
 MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
 NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_
 NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_
 DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_
 PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
 NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
 POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
 PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
 ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
 STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
 STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output

attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | |=

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

getPLSqlRecordAttributesClause

Gets the names of all the attributes of the PLSQL Record Type

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE PLSQL_RECORD_TYPE 'SOME_PLSQL_RECORD_TYPE'
ATTRIBUTE 'ATTR1'
```

```
GET PROPERTIES (DATATYPE)
```

This will retrieve the PLSQL Record Type "SOME_PLSQL_RECORD_TYPE"'s attribute "ATTR1"'s datatype.

See Also

OMBRETRIEVE, OMBALTER PLSQL_RECORD_TYPE, OMBCREATE PLSQL_RECORD_TYPE, OMBDROP PLSQL_RECORD_TYPE

OMBRETRIEVE PLSQL_REF_CURSOR_TYPE

Purpose

Retrieve details of the Ref-cursor Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
retrieveRefCursorTypeCommand = OMBRETRIEVE PLSQL_REF_CURSOR_TYPE
    "QUOTED_STRING" "retrieveRefCursorTypeClause"
retrieveRefCursorTypeClause = GET "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveRefCursorTypeCommand

Retrieves the details of the Ref-cursor Type with the given name.

retrieveRefCursorTypeClause

Retrieves the details of the Ref-cursor Type with the given name.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for PLSQL_REF_CURSOR_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Ref-Cursor Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Ref-Cursor Type

Name: RETURN_TYPE

Type: STRING(20)

Valid Values: N/A

Default: "

Return type of the Ref-Cursor Type. This should be a PLSQL Record Type.

Properties for PLSQL_REF_CURSOR_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier.

Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this

attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,
NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_
ABBREV,
NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_
COUNTRYCODE,
NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_
DELIVERYBEATCODE,
NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1,
NA_EXTRA_10,
NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,
NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_
EXTRA_7,
NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_
CODE,
NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_
FIRSTNAMESTD,
NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_
INSTALLATIONTYPE,
NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_
ISGOODGROUP,
NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME,
NA_LOCALITY_2,
NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_
ORDER, NA_MCD,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_
MIDDLENAME3,
NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_
NEIGHBORHOOD,
NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_
DESIG,
NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_
PARSESTATUSDESC,
NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
NA_PHONE,
NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
POSTALCODEFORMATTED,
NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
PRENAME,
NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
ROUTENAME,

NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
 STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
 STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE of SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, -=, =| |, || =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE PLSQL_REF_CURSOR_TYPE 'SOME_PLSQL_REF_CURSOR_TYPE'  
GET  
PROPERTIES (RETURN_TYPE)
```

This will retrieve the return type of the Ref-cursor Type
"SOME_PLSQL_REF_CURSOR_TYPE".

See Also

OMBRETRIEVE, OMBALTER PLSQL_REF_CURSOR_TYPE, OMBCREATE PLSQL_REF_CURSOR_TYPE, OMBDROP PLSQL_REF_CURSOR_TYPE

OMBRETRIEVE PLSQL_TABLE_TYPE

Purpose

Retrieve details of the Table Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
retrievePlSqlTableTypeCommand = OMBRETRIEVE PLSQL_TABLE_TYPE
    "QUOTED_STRING" "retrievePlSqlTableTypeClause"
retrievePlSqlTableTypeClause = GET "getPropertiesClause"
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrievePlSqlTableTypeCommand

Retrieves the details of the Table Type with the given name.

retrievePlSqlTableTypeClause

Retrieves the properties of the table type with the given name.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for PLSQL_TABLE_TYPE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Table Type

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Table Type

Name: RETURN_TYPE

Type: STRING(20)

Valid Values: NUMBER, VARCHAR2, VARCHAR, DATE, FLOAT

Default: "

Return type of the Table Type. This can be a scalar type or a PLSQL Record Type.

Properties for PLSQL_TABLE_TYPE:

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier

indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero,

based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be

used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,
NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,
NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,
NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,
NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,
NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1,
NA_EXTRA_10,
NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,
NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,
NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,
NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,
NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,
NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,
NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,

NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
NA_STREETCOMP CORRECTED, NA_STREETCOMP MATCH, NA_STREETCORRECTED,
NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE of SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, || =

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE PLSQL_TABLE_TYPE 'SOME_PLSQL_TABLE_TYPE' GET  
PROPERTIES
```

```
(RETURN_TYPE)
```

This will retrieve the return type of the Table Type

"SOME_PLSQL_TABLE_TYPE".

See Also

OMBRETRIEVE, OMBALTER PLSQL_TABLE_TYPE, OMBCREATE PLSQL_TABLE_
TYPE, OMBDROP PLSQL_TABLE_TYPE

OMBRETRIEVE PLUGGABLE_MAPPING

Purpose

Retrieve details of pluggable mapping such as the number of operators and their connections.

Prerequisites

The current context of scripting must be a project or pluggable map folder.

Syntax

```
retrievePluggableMappingCommand = OMBRETRIEVE PLUGGABLE_MAPPING
    "pluggableMapName" ( "retrieveOperatorOwnerDetailClause" |
        "testConnectionClause" )
pluggableMapName = "QUOTED_STRING"
retrieveOperatorOwnerDetailClause = GET ( PROPERTIES "propertyKeyList" |
    "getReferenceIconSetClause" | [ "operatorType" ] OPERATORS [
        "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "operatorLocator" "getOperatorDetailClause"
testConnectionClause = HAS CONNECTION FROM "mappableBottomUpLocator" TO
    "mappableBottomUpLocator"
propertyKeyList = "( " "propertyKey" { "," "propertyKey" } )"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
connectionConditionClause = CONNECTED ( FROM "mappableBottomUpLocator" |
    TO "mappableBottomUpLocator" )
childType = "UNQUOTED_STRING"
getChildDetailClause = ( "childType" "childName" )+ GET ( PROPERTIES
    "propertyKeyList" | "childType" )
operatorLocator = [ "pluggableMapLocator" ] OPERATOR "operatorName"
getOperatorDetailClause = GET ( PROPERTIES "propertyKeyList" | [
    "groupDirection" ] GROUPS [ "connectionConditionClause" ] |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause" | "groupLocator"
    "getGroupDetailClause"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
pluggableMapLocator = ( PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapLocator" ] )
operatorName = "QUOTED_STRING"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupLocator = GROUP "groupName"
getGroupDetailClause = GET ( PROPERTIES "propertyKeyList" | ATTRIBUTES [
    "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "attributeLocator" "getAttributeDetailClause"
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
groupName = "QUOTED_STRING"
attributeLocator = ATTRIBUTE "attributeName"
getAttributeDetailClause = GET ( PROPERTIES "propertyKeyList" |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapBottomUpLocator" ] )
```

```
attributeName = "QUOTED_STRING"
```

Keywords And Parameters

retrievePluggableMappingCommand

Retrieve the detail of a pluggable mapping such as how many mapping operators are there are which mapping operators are connected to each other.

pluggableMapName

Name of the pluggable map.

retrieveOperatorOwnerDetailClause

Retrieve the desired detail of a mapping or a pluggable mapping.

testConnectionClause

Verify if there is a connection between mapping operators, mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT, CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDUPLICATOR, DIMENSION, EXPAND_OBJECT, EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCAST, LCRSPLITTER, MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER, OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION, TRANSFORMATION, UNPIVOT, VIEW.

connectionConditionClause

List objects only if they are connected from or to objects specified in the

connection condition.

childType

Type of a child that belongs to map, mapping operator, mapping group or mapping attribute.

getChildDetailClause

Get the desired detail of a child object that belongs to the mapping, map variable, mapping operator, mapping group or mapping attribute.

operatorLocator

Location of a mapping operator.

getOperatorDetailClause

Get the desired detail of a mapping operator.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Length of the attribute.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Precision of the attribute.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for PLUGGABLE_MAPPING:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING

Valid Values: N/A

Default: "

Record Type Values.

Name: RETURN_TABLE_OF_SCALAR

Type: BOOLEAN

Valid Values: true, false

Default: false

This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR

Type: STRING

Valid Values: N/A

Default: "

An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES

Type: STRING

Valid Values: N/A

Default: NULL, NULL

A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SPLIT_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Condition that defines when to perform the attribute maps for the attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero, based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9,

NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,

NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,

NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,

NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,

NA_COUNTRYCODE3, NA_COUNTRYNAM, NA_COUNTYNAM, NA_DELIVERYBEATCODE,

NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1, NA_EXTRA_10,

NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,

NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,

NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,

NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,

NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,

NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,

NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,

NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
 NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME,
 NA_LOCALITY_2,
 NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_
 ORDER, NA_MCD,
 NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_
 MIDDLENAME3,
 NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
 NA_NAMEDESIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_
 NEIGHBORHOOD,
 NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_
 DESIG,
 NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_
 PARSESTATUSDESC,
 NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT,
 NA_PHONE,
 NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_
 POSTALCODEFORMATTED,
 NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_
 PRENAME,
 NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_
 ROUTENAME,
 NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
 NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_
 STREETCORRECTED,
 NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_
 STREETNUMBERMATCH,
 NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
 NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
 NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level

Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, || =

Default: =

The computation to be performed on this attribute between the incoming data

and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default

it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using

the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to

compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a
staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this

operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: ≥ 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to lookup a value in the bound table. If the condition is not met, the default value expression will be returned. If a default expression is not defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE

Type: STRING

Valid Values: N/A

Default: "

A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"MATCHMERGE.GENERAL.MATCH_NEW_RECORD_
CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT
Type: STRING
Valid Values: N/A
Default: "
Hint used on inline view when extracting using SQL

Name: SET_OPERATION
Type: STRING
Valid Values: INTERSECT, MINUS, UNION, UNIONALL
Default: UNION
Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a

staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by

constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: ≥ 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row

will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING
Valid Values: N/A
Default: "
Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING
Valid Values: N/A
Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING
Valid Values: N/A
Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)
Valid Values: N/A
Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING
Valid Values: N/A
Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING
Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

pluggableMapLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

operatorName

Name of a mapping operator.

groupDirection

Direction of a mapping group.

groupLocator

Location of a mapping group.

getGroupDetailClause

Get the desired detail of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

groupName

Name of a mapping group.

attributeLocator

Location of a mapping attribute.

getAttributeDetailClause

Get the desired detail of a mapping attribute.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

attributeName

Name of a mapping attribute.

Examples

```
OMBRETRIEVE PLUGGABLE_MAP 'PLUGGABLE_MAP1' GET OPERATORS
```

```
OMBRETRIEVE PLUGGABLE_MAP 'PLUGGABLE_MAP1' OPERATOR 'SRC1'  
GROUP
```

```
'INOUTGRP1'
```

```
GET ATTRIBUTE CONNECTED TO OPERATOR 'target1'
```

```
OMBRETRIEVE PLUGGABLE_MAP 'PLUGGABLE_MAP1' OPERATOR 'SRC1'
```

GET PROPERTIES (BUSINESS_NAME, DESCRIPTION)

See Also

OMBRETRIEVE, OMBCREATE MAPPING, OMBALTER MAPPING, OMBDROP MAPPING

OMBRETRIEVE PLUGGABLE_MAPPING_FOLDER

Purpose

Retrieve details of pluggable map folder such as its business name and description.

Prerequisites

The current context of scripting must be a project.

Syntax

```
retrievePluggableMappingFolderCommand = OMBRETRIEVE
    PLOGGABLE_MAPPING_FOLDER "pluggableMapFolderName" GET ( PROPERTIES
        "propertyKeyList" | "getReferenceIconSetClause" )
pluggableMapFolderName = "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyKey = "UNQUOTED_STRING"
```

Keywords And Parameters

retrievePluggableMappingFolderCommand

Retrieve the detail of a pluggable mapping such as its business name and description.

pluggableMapFolderName

Name of the pluggable map folder.

propertyKeyList

The list of property keys.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

Examples

```
OMBRETRIEVE PLUGGABLE_MAP_FOLDER 'PLUGGABLE_MAP_FOLDER1' GET  
PROPERTIES  
(BUSINESS_NAME)
```

See Also

OMBRETRIEVE

OMBRETRIEVE PRESENTATION_TEMPLATE

Purpose

Retrieve details of a presentation template.

Prerequisites

Should be in the context of a business presentation or use the full path.

Syntax

```
retrieveReportCommand = OMBRETRIEVE PRESENTATION_TEMPLATE "QUOTED_STRING"
  ( "retrieveReportClause" | "retrieveReportDataItemClause" |
    "retrieveReportEdgeItemClause" )
retrieveReportClause = GET ( "getPropertiesClause" |
  "getReferenceIconSetClause" | "getReportSCOClaue" )
retrieveReportDataItemClause = DATA_ITEM "QUOTED_STRING" GET ( ( REF |
  REFERENCE ) MEASURE | "getPropertiesClause" )
retrieveReportEdgeItemClause = EDGE_ITEM "QUOTED_STRING" GET ( REFERENCES
  | "getPropertiesClause" | ( REF | REFERENCE ) ( ROLE | DIMENSION ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getReportSCOClaue = EDGE_ITEMS | DATA_ITEMS | ( REF | REFERENCE ) CUBE
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveReportCommand

To retrieve a presentation template.

QUOTED_STRING

name of the presentation template.

retrieveReportClause

Retrieves the contents of the presentation template.

GET

For presentation template this clause retrieves the following

DATA_ITEMS retrieves the list of data items for the presentation template.

EDGE_ITEMS retrieves the list of edge items for the presentation template.

REF CUBE retrieves the Cube that the presentation template relates to.

retrieveReportDataItemClause

Retrieves the properties of the data item.

QUOTED_STRING

name of the data item.

GET

For data item this clause retrieves the following

REF MEASURE retrieves the measure the report item uses.

retrieveReportEdgeItemClause

Retrieves the properties of the edge item.

QUOTED_STRING

name of the edge item.

GET

For edge item this clause retrieves the following

REF ROLE retrieves the dimension role the edge item is associated with,
this returns an empty value if not present.

getPropertiesClause

Retrieves the properties of the object.

Basic properties for presentation template:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the presentation template

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the presentation template

Name: PRESENTATION_TYPE

Type: STRING(40)

Valid Values: CROSSTAB, TABLE or a subtype of graph

Default: "

The type of the presentation template

Basic properties for EDGE_ITEM:

Name: PLACEMENT

Type: STRING(40)

Valid Values: TOP OR SIDE, TOP, SIDE, PAGE

Default: "

The placement of the edge item in the presentation template

Properties for PRESENTATION_TEMPLATE:

Name: CATALOG_FOLDER

Type: STRING

Valid Values: N/A

Default: "

Catalog Folder for deployed BI Beans presentation

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for referenced database objects

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Note:

1. N/A means any valid character in supported character set.

2. " represents an empty string

getReferenceIconSetClause

Get specified Icon Set.

getReportSCOClause

This clause gets the details of the presentation template.

propertyNameList

This is the list of property names.

Examples

```
OMBRETRIEVE PRESENTATION_TEMPLATE 'SALES' GET  
PROPERTIES(DESCRIPTION)
```

See Also

```
OMBALTER PRESENTATION_TEMPLATE, OMBCREATE PRESENTATION_  
TEMPLATE
```

OMBRETRIEVE PROCEDURE

Purpose

Retrieve details of the Procedure.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

Syntax

```
retrieveProcedureCommand = OMBRETRIEVE PROCEDURE "QUOTED_STRING" (
    "getPropertiesClause" | "getFuncProcParameterClause" |
    "getFuncProcParameterPositionClause" | "getFuncProcSignatureClause" |
    "retrieveFuncProcParameterClause" | "getRelationalDependentsClause" |
    "getReferenceIconSetClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getFuncProcParameterClause = GET PARAMETERS
getFuncProcParameterPositionClause = GET PARAMETER AT POSITION
    "INTEGER_LITERAL"
getFuncProcSignatureClause = GET SIGNATURE
retrieveFuncProcParameterClause = PARAMETER "QUOTED_STRING"
    "getPropertiesClause"
getRelationalDependentsClause = GET ( REF | REFERENCE ) ( TABLES | VIEWS |
    MATERIALIZED_VIEWS | SEQUENCES | FUNCTIONS | PROCEDURES | PACKAGES )
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveProcedureCommand

This command retrieves the details of a Procedure

QUOTED_STRING

Name of the existing Procedure or path to the Procedure.

getPropertiesClause

Used to get properties (core, user-defined) for procedure. Valid properties are as shown:

Basic properties for PROCEDURE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Procedure

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Procedure

Name: IMPLEMENTATION

Type: STRING

Valid Values: N/A

Default: "

Set the code for Procedure which is included global variable declaration and code between BEGIN and END.

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB, BOOLEAN,

CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO

MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,
VARHCAR,

VARCHAR2, XMLTYPE

Default: NUMBER

Set the data type for Parameter

Name: IN_OUT

Type: STRING

Valid Values: IN, OUT, INOUT

Default: 'IN'

Set the parameter mode for Parameter

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Set the default value for Parameter

Properties for PROCEDURE:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getFuncProcParameterClause

Get all the parameter names of the Procedure

getFuncProcParameterPositionClause

Get the parameter position of Function

getFuncProcSignatureClause

Get the complete signature of the Function which includes parameter names, datatype, in/out type and default values

retrieveFuncProcParameterClause

Get the parameter information such as datatype, default value, in/out type and position

QUOTED_STRING

Name of the existing Parameter

getRelationalDependentsClause

This clause retrieves referential dependencies to other relational objects.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE PROCEDURE 'proc' GET PROPERTIES (DESCRIPTION, UOID,  
BUSINESS_NAME, IMPLEMENTATION, IS_IMPORTED)
```

See Also

OMBRETRIEVE

OMBRETRIEVE PROCESS_FLOW

Purpose

Retrieves the details of the Process Flow.

Prerequisites

Should be in the context of a Process Flow Package.

Syntax

```
retrieveProcessFlowCommand = OMBRETRIEVE PROCESS_FLOW "QUOTED_STRING" (
    "retrieveProcessFlowClause" | "retrieveActivityClause" |
    "retrieveTransitionClause" | "retrieveProcessParameterClause" |
    "retrieveProcessVariableClause" )
retrieveProcessFlowClause = "getPropertiesClause" |
    "getReferenceIconSetClause" | GET "getProcessFlowSCOClauses"
retrieveActivityClause = ACTIVITY "QUOTED_STRING" ( "getPropertiesClause"
    | "getReferenceIconSetClause" | "getActivityBoundObject" |
    "getActivityTransitionClause" | "getActivityParameterClause" |
    "getActivityParameterPropertiesClause" )
retrieveTransitionClause = TRANSITION "QUOTED_STRING" (
    "getPropertiesClause" | GET "getTransitionActivityClause" )
retrieveProcessParameterClause = PARAMETER "QUOTED_STRING"
    "getPropertiesClause"
retrieveProcessVariableClause = VARIABLE "QUOTED_STRING"
    "getPropertiesClause"
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
getProcessFlowSCOClauses = ACTIVITIES | ( ( "UNQUOTED_STRING" |
    DATA_AUDITOR | USER_DEFINED | SUBPROCESS | MAPPING | TRANSFORMATION )
    ACTIVITIES ) | TRANSITIONS | PARAMETERS | VARIABLES
getActivityBoundObject = GET ( REFERENCE | REF )
getActivityTransitionClause = GET ( INCOMING_TRANSITIONS |
    OUTGOING_TRANSITIONS )
getActivityParameterClause = ( GET PARAMETERS )
getActivityParameterPropertiesClause = PARAMETER "QUOTED_STRING"
    "getPropertiesClause"
getTransitionActivityClause = ( SOURCE_ACTIVITY | DESTINATION_ACTIVITY )
propertyNameList = ( "UNQUOTED_STRING" | BINDING ) { "," (
    "UNQUOTED_STRING" | BINDING ) }
```

Keywords And Parameters

retrieveProcessFlowCommand

Retrieve the details of an existing process flow.

retrieveProcessFlowClause

This clause retrieve the Process Flow.

retrieveActivityClause

This clause retrieves the Activity of a Process Flow.

retrieveTransitionClause

This clause retrieves the Transition of a Process Flow.

retrieveProcessParameterClause

This clause retrieves the Parameter of a Process Flow.

getPropertiesClause

Used to get properties (core, user-defined) for process flow. Valid properties are as shown:

Base properties for Process Flow, Activity, Transition and Parameter:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Process Flow

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Process Flow

Basic properties for Transition:

Name: TRANSITION_CONDITION

Type: STRING

Valid Values: ", SUCCESS, ERROR, WARNING

Default: ", that is, Unconditional

Gets the Transition Condition of a Transition

Basic properties for Activity Parameter :

Name: DATATYPE

Type: STRING

Valid Values: INTEGER, FLOAT, DATE, STRING, BOOLEAN

Default: STRING

Gets the datatype of a Activity Parameter

Name: DIRECTION

Type: STRING

Valid Values: IN

Default: IN

Gets the direction of a Activity Parameter

Name: ISLITERALVALUE

Type: STRING

Valid Values: TRUE FALSE

Default: TRUE

Whether the value stored is a literal value or an interpreted value, valid for PARAMETERS and VARIABLES.

Name: VALUE

Type: STRING

Valid Values: Examples '123', '123.456', 'Jan-08-2003', 'I am String', 'true'

Default: "

Gets the value of a Activity Parameter

Name: BINDING

Type: STRING

Valid Values: Examples 'PROCESS_PARAM1', 'PROCESS_PARAM2'

Default: "

Get the name of the process parameter that this parameter is bound to, empty if not bound.

getProcessFlowSCOClause

For the current process flow, retrieve all activities or only activities of a specific type.

getActivityBoundObject

Retrieve the name of the object that the current activity refers to, pertinent to only MAPPING, SUBPROCESS and TRANSFORMATION activities.

getActivityTransitionClause

This clause retrieves all Activities of a Process Flow.

getActivityParameterClause

Retrieve the activity parameters for the current activity.

getActivityParameterPropertiesClause

Retrieve the details of an activity parameter.

getTransitionActivityClause

This clause retrieves all Transitions of a Process Flow.

propertyNameList

A comma delimited set of property names to set.

Examples

```
OMBRETRIEVE PROCESS_FLOW 'process_flow' GET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the Process Flow "process_flow"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE PROCESS_FLOW_MODULE

Purpose

Retrieve details of the Process Flow Module.

Prerequisites

Should be in the context of a project.

Syntax

```
retrieveProcessFlowModuleCommand = OMBRETRIEVE PROCESS_FLOW_MODULE
    "QUOTED_STRING" ( "getPropertiesClause" | "getReferenceLocationClause"
    | "getReferenceIconSetClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveProcessFlowModuleCommand

Retrieve the details of an existing process flow module.

getPropertiesClause

Used to get properties (core, user-defined) for process flow module.

Base properties for PROCESS_FLOW_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Process Flow Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Process Flow Module

getReferenceLocationClause

Retrieve the name of the Workflow engine location referenced by this process flow module.

propertyNameList

Comma separated list of property names. Property names are not in quotation marks.

Examples

```
OMBRETRIEVE PROCESS_FLOW_MODULE 'process_module' GET PROPERTIES  
(DESCRIPTION, BUSINESS_NAME)
```

This will retrieve the Process Flow Module "process_module"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE PROCESS_FLOW_PACKAGE

Purpose

Retrieve details of the Process Flow Package.

Prerequisites

Should be in the context of a Process Flow Module.

Syntax

```
retrieveProcessFlowPackageCommand = OMBRETRIEVE PROCESS_FLOW_PACKAGE
    "QUOTED_STRING" ( "getPropertiesClause" | "getReferenceIconSetClause"
    )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = ( "UNQUOTED_STRING" | BINDING ) { "," (
    "UNQUOTED_STRING" | BINDING ) }
```

Keywords And Parameters

retrieveProcessFlowPackageCommand

Retrieve the details of an existing process flow package.

getPropertiesClause

Used to get properties (core, user-defined) for process flow packages.

Valid properties are as shown:

Basic properties for PROCESS_FLOW_PACKAGE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Process Flow Package

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Process Flow Package

propertyNameList

A comma delimited set of property names to set.

Examples

```
OMBRETRIEVE PROCESS_FLOW_PACKAGE 'process_package' GET PROPERTIES  
(DESCRIPTION, BUSINESS_NAME)
```

This will retrieve the Process Flow Package "process_package"'s
description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE PROFILE_REFERENCE

Purpose

To retrieve properties of a profile reference.

Prerequisites

In the context of a Data Profile.

Syntax

```

retrieveProfileReferenceCommand = OMBRETRIEVE PROFILE_REFERENCE
    "QUOTED_STRING" ( "retrieveProfileReferenceClause" |
        "retrieveDataRuleUsageClause" | "retrieveProfileAttributeClause" |
        "retrieveProfileForeignKeyClause" | "retrieveProfileUniqueKeyClause" |
        "retrieveRowRelationshipClause" |
        "retrieveFunctionalDependencyClause" )
retrieveProfileReferenceClause = GET ( "getPropertiesClause" |
    "getProfileReferenceSCOClause" )
retrieveDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" ( GET (
    "getPropertiesClause" | GROUPS ) | GROUP "QUOTED_STRING" ( GET (
    "getPropertiesClause" | ATTRIBUTES | REF TABLE ) | ATTRIBUTE
    "QUOTED_STRING" ( GET ( "getPropertiesClause" | REF COLUMN ) ) ) )
retrieveProfileAttributeClause = PROFILE_ATTRIBUTE "QUOTED_STRING" ( GET (
    "getPropertiesClause" | "getDomainValuesClause" |
    "getCharPatternValuesClause" | "getWordPatternValuesClause" ) |
    DOMAIN_VALUE "QUOTED_STRING" ( GET "getPropertiesClause" ) |
    CHARPATTERN_VALUE "QUOTED_STRING" ( GET "getPropertiesClause" ) |
    WORDPATTERN_VALUE "QUOTED_STRING" ( GET "getPropertiesClause" ) )
retrieveProfileForeignKeyClause = PROFILE_FOREIGN_KEY "QUOTED_STRING" (
    GET ( "getPropertiesClause" | "getRedundantAttributesForFKClause" |
    "getReferencingAttributesClause" | "getReferencingUKClauseForFK" ) |
    REDUNDANT_ATTRIBUTE "QUOTED_STRING" ( GET "getPropertiesClause" ) )
retrieveProfileUniqueKeyClause = PROFILE_UNIQUE_KEY "QUOTED_STRING" ( GET
    ( "getPropertiesClause" | "getReferencingAttributesClause" ) )
retrieveRowRelationshipClause = ROW_RELATIONSHIP "QUOTED_STRING" ( GET (
    "getPropertiesClause" | "getReferencingAttributesClause" |
    "getReferencingUKClauseForRR" ) )
retrieveFunctionalDependencyClause = FUNCTIONAL_DEPENDENCY "QUOTED_STRING"
    ( GET ( "getPropertiesClause" | "getDeterminantAttributesClause" |
    "getDependentAttributeClause" ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getProfileReferenceSCOClause = PROFILE_ATTRIBUTES | PROFILE_ATTRIBUTE AT
    POSITION "INTEGER_LITERAL" | PROFILE_FOREIGN_KEYS |
    PROFILE_UNIQUE_KEYS | DATA_RULE_USAGES | FUNCTIONAL_DEPENDENCIES |
    ROW_RELATIONSHIPS
getDomainValuesClause = DOMAIN_VALUES
getCharPatternValuesClause = CHARPATTERN_VALUES
getWordPatternValuesClause = WORDPATTERN_VALUES
getRedundantAttributesForFKClause = REDUNDANT_ATTRIBUTES
getReferencingAttributesClause = PROFILE_ATTRIBUTES
getReferencingUKClauseForFK = UNIQUE_KEY
getReferencingUKClauseForRR = ROW_RELATIONSHIP
getDeterminantAttributesClause = DETERMINANT_ATTRIBUTES
getDependentAttributeClause = DEPENDENT_ATTRIBUTE
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }

```


Keywords And Parameters

retrieveProfileReferenceCommand

This clause retrieves the results of a profile of a source.

retrieveDataRuleUsageClause

This clause retrieves the data rule usages defined or derived for this particular profile reference.

retrieveProfileForeignKeyClause

This clause retrieves the discovered foreign keys for this particular profile reference.

retrieveProfileUniqueKeyClause

This clause retrieves the discovered unique keys for this particular profile reference.

retrieveRowRelationshipClause

This clause retrieves the discovered row relationships for this particular profile reference.

retrieveFunctionalDependencyClause

This clause retrieves the discovered functional dependencies for this particular profile reference. The format for discovered functional dependencies are A->B, where A is the determinant column/s and B is the dependent column.

getPropertiesClause

This clause retrieves the properties for a particular object.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' GET PROFILE_ATTRIBUTES
```

This will retrieve its columns.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' PROFILE_ATTRIBUTE 'JOB'  
GET
```

```
DOMAIN_VALUES
```

This will retrieve the discovered domain values for the column JOB.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' GET FUNCTIONAL_
DEPENDENCIES
```

This will retrieve the discovered functional dependencies for table EMPLOYEES.

The format of the discovered functional dependencies is A->B, where A is composed of the determinant attributes while B is the dependent attribute.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' GET PROFILE_FOREIGN_
KEYS
```

This will retrieve the discovered foreign keys for table EMPLOYEES.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' GET PROFILE_UNIQUE_
KEYS
```

This will retrieve the discovered unique keys for table EMPLOYEES.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' GET ROW_RELATIONSHIPS
```

This will retrieve the discovered row relationships for table EMPLOYEES.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' PROFILE_FOREIGN_KEY
'FK_1' GET
```

```
PROPERTIES (COMPLIANT_QUERY)
```

This will retrieve the sql query for drilling down into the rows that satisfy the discovered foreign key FK_1.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' PROFILE_FOREIGN_KEY
'FK_1' GET
```

```
PROPERTIES (NONCOMPLIANT_QUERY)
```

This will retrieve the sql query for drilling down into the rows that do not satisfy the discovered foreign key FK_1.

```
OMBRETRIEVE PROFILE_REFERENCE 'EMPLOYEES' PROFILE_FOREIGN_KEY
'FK_1' GET
```

```
PROPERTIES (DRILLDOWN_QUERY)
```

This will retrieve the sql query for drilling down into all the rows that both satisfy and do not satisfy the discovered foreign key FK_1. The returned rowset will begin with either a 0 (NONCOMPLIANT) or 1 (COMPLIANT). You can use this same format to get the corresponding queries for each of the discovered metadata.

See Also

OMBRETRIEVE

OMBRETRIEVE PROJECT

Purpose

Retrieve details of the project.

Prerequisites

Should be in the top level context.

Syntax

```
retrieveProjectCommand = OMBRETRIEVE PROJECT "QUOTED_STRING" (
    "getPropertiesClause" | "getReferenceIconSetClause" )
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveProjectCommand

Retrieve the details of a Project

QUOTED_STRING

Name of the existing project or path to the project.

getPropertiesClause

Retrieve a set of properties that is associated with a Project.

Basic properties for PROJECT:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a Project

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of a Project

getReferenceIconSetClause

Retrieve the icon set referenced by this project.

propertyNameList

Comma separated list of property names. Property names are unquoted.

Examples

OMBRETRIEVE PROJECT 'New Project' GET PROPERTIES (DESCRIPTION,
BUSINESS_NAME)

This will retrieve the project "New Project"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE QUEUE_PROPAGATION

Purpose

Retrieve details of the Queue Propagation.

Prerequisites

Should be in the context of an Advanced Queue.

Syntax

```
retrieveQPCommand = OMBRETRIEVE QUEUE_PROPAGATION "QUOTED_STRING" ( GET
    "getQPPropertiesClause" )
getQPPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveQPCommand

Retrieves the details of the Queue Propagation with the given name.

getQPPropertiesClause

Retrieves the values of the given Properties for the Queue Propagation with the given name.

Basic properties for QUEUE_PROPAGATION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Queue Propagation

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Queue Propagation

Name: TARGET_QUEUE

Type: STRING(4000)

Valid Values: N/A

Default: "

Target Queue for the Queue Propagation. This has to be the name of a Queue existing in any Oracle Module.

Properties for QUEUE_PROPAGATION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DURATION

Type: STRING

Valid Values: N/A

Default: "

The duration of proagation to be done. The default value is null.Applicable only for non-streams queue propagation.

Name: GENERATE_DBLINK

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Database Link which is used for propagation

Name: GENERATE_QUEUE_PROPAGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Queue Propagation

Name: GENERATE_REPLICATION_RULE

Type: BOOLEAN

Valid Values: true, false

Default: false

Generate Ruleset and Rule for Replication purpose in Streams queue propagation

Name: GENERATE_SCHEDULE_PROPAGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate script for Scheduling propagation. Applicable only for non-streams queue propagation.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: LATENCY

Type: STRING

Valid Values: N/A

Default: 60

The latency for the queue propagation. By default the value is 60. Applicable only for non-streams queue propagation.

Name: NEXT_TIME

Type: STRING

Valid Values: N/A

Default: "

Next time when the propagation to be done. The default value is null. Applicable only for non-streams queue propagation.

Name: NOT_PERMITTED_TAG_VALUES

Type: STRING

Valid Values: N/A

Default: "

List of comma separated Tag values (in Hex numbers) which are not allowed for propagation

Name: PERMITTED_TAG_VALUES

Type: STRING

Valid Values: N/A

Default: "

List of comma separated Tag values (in Hex numbers) which are allowed for propagation

Name: START_TIME

Type: STRING

Valid Values: N/A

Default: SYSDATE

The start time for the propagation to happen. The default value is SYSDATE. Applicable only for non-streams queue propagation.

Name: SUBSCRIBER_TRANSFORMATION

Type: STRING

Valid Values: N/A

Default: "

A Transformation that will be applied before propagation to the target queue. Applicable only for non-streams queue propagation.

Name: SUBSCRIBER_RULE_CONDITION

Type: STRING

Valid Values: N/A

Default: "

A Rule condition to check whether the message can be propagated to the subscriber. Applicable only for non-streams queue propagation.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE QUEUE_PROPAGATION 'SOME_QUEUE_PROPAGATION' GET
PROPERTIES
```


(SUBSCRIBER_RULE_CONDITION, SUBSCRIBER_TRANSFORMATION, START_TIME, DURATION,

NEXT_TIME, LATENCY, PERMITTED_TAG_VALUES, NOT_PERMITTED_TAG_VALUES,

GENERATE_DBLINK, GENERATE_SCHEDULE_PROPAGATION)

This will retrieve the Queue Propagation "SOME_QUEUE_PROPAGATION"'s properties.

See Also

OMBRETRIEVE, OMBALTER QUEUE_PROPAGATION, OMBCREATE QUEUE_PROPAGATION, OMBDROP QUEUE_PROPAGATION

OMBRETRIEVE QUEUE_TABLE

Purpose

Retrieve details of the Queue Table.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
retrieveQTCommand = OMBRETRIEVE QUEUE_TABLE "QUOTED_STRING" ( GET
    "getQTPropertiesClause" )
getQTPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveQTCommand

Retrieves the details of the Queue Table with the given name.

getQTPropertiesClause

Retrieves the values of the given Properties for the Queue Table with the given name.

Basic properties for QUEUE_TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Queue Table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Queue Table

Name: PAYLOAD_TYPE

Type: STRING(4000)

Valid Values: N/A

Default: "

Object Type for the Queue Table. This has to be the name of an Object Type (OBJECT_TYPE) existing in any Oracle Module.

Properties for QUEUE_TABLE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: GENERATE_QUEUE_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate code to create the queue table that will persist the messages of this Advanced Queue.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE QUEUE_TABLE 'SOME_QUEUE_TABLE' GET PROPERTIES  
(TABLESPACE,  
GENERATE_QUEUE_TABLE)
```

This will retrieve the Queue Table "SOME_QUEUE_TABLE"'s properties.

See Also

OMBRETRIEVE, OMBALTER QUEUE_TABLE, OMBCREATE QUEUE_TABLE,
OMBDROP QUEUE_TABLE

OMBRETRIEVE REAL_TIME_MAPPING

Purpose

Retrieve Real Time mapping details such as the number of operators and their connections.

Prerequisites

The current context must be in an Oracle Module.

Syntax

```

retrieveRealTimeMappingCommand = OMBRETRIEVE REAL_TIME_MAPPING
    "mappingName" ( "retrieveOperatorOwnerDetailClause" |
        "testConnectionClause" )
mappingName = "QUOTED_STRING"
retrieveOperatorOwnerDetailClause = GET ( PROPERTIES "propertyKeyList" |
    "getReferenceIconSetClause" | [ "operatorType" ] OPERATORS [
    "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "operatorLocator" "getOperatorDetailClause"
testConnectionClause = HAS CONNECTION FROM "mappableBottomUpLocator" TO
    "mappableBottomUpLocator"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
operatorType = PLUGGABLE_MAPPING | "UNQUOTED_STRING"
connectionConditionClause = CONNECTED ( FROM "mappableBottomUpLocator" |
    TO "mappableBottomUpLocator" )
childType = "UNQUOTED_STRING"
getChildDetailClause = ( "childType" "childName" )+ GET ( PROPERTIES
    "propertyKeyList" | "childType" )
operatorLocator = [ "pluggableMapLocator" ] OPERATOR "operatorName"
getOperatorDetailClause = GET ( PROPERTIES "propertyKeyList" | [
    "groupDirection" ] GROUPS [ "connectionConditionClause" ] |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause" | "groupLocator"
    "getGroupDetailClause"
mappableBottomUpLocator = "operatorBottomUpLocator" |
    "groupBottomUpLocator" | "attributeBottomUpLocator"
propertyKey = "UNQUOTED_STRING"
childName = "QUOTED_STRING"
pluggableMapLocator = ( PLUGGABLE_MAPPING "pluggableMapName" [
    "pluggableMapLocator" ] )
operatorName = "QUOTED_STRING"
groupDirection = INPUT | OUTPUT | INPUT_OUTPUT
groupLocator = GROUP "groupName"
getGroupDetailClause = GET ( PROPERTIES "propertyKeyList" | ATTRIBUTES [
    "connectionConditionClause" ] | "childType" ) | "getChildDetailClause"
    | "attributeLocator" "getAttributeDetailClause"
operatorBottomUpLocator = OPERATOR "operatorName" [
    "pluggableMapBottomUpLocator" ]
groupBottomUpLocator = GROUP "groupName" OF "operatorBottomUpLocator"
attributeBottomUpLocator = ATTRIBUTE "attributeName" OF
    "groupBottomUpLocator"
pluggableMapName = "QUOTED_STRING"
groupName = "QUOTED_STRING"
attributeLocator = ATTRIBUTE "attributeName"
getAttributeDetailClause = GET ( PROPERTIES "propertyKeyList" |
    BOUND_OBJECT | "childType" ) | "getChildDetailClause"
pluggableMapBottomUpLocator = ( OF PLUGGABLE_MAPPING "pluggableMapName" [

```

```
    "pluggableMapBottomUpLocator" ] )  
attributeName = "QUOTED_STRING"
```

Keywords And Parameters

mappingName

Name of the mapping.

retrieveOperatorOwnerDetailClause

Retrieve the desired detail of a mapping or a pluggable mapping.

testConnectionClause

Verify if there is a connection between mapping operators, mapping groups or mapping attributes.

propertyKeyList

The list of property keys.

operatorType

Type of a mapping operator. The following operator types are available:

ADVANCED_QUEUE, AGGREGATOR, ANYDATA_CAST, CONSTANT,
CONSTRUCT_OBJECT, CUBE,

DATA_GENERATOR, DEDUPLICATOR, DIMENSION, EXPAND_OBJECT,
EXPRESSION,

EXTERNAL_TABLE, FILTER, FLAT_FILE, INPUT_PARAMETER, INPUT_
SIGNATURE,

ITERATOROPERATOR, JOINER, KEY_LOOKUP, LCRCAST, LCRSPLITTER,
MATCHMERGE,

MATERIALIZED_VIEW, NAME_AND_ADDRESS, OUTPUT_PARAMETER,
OUTPUT_SIGNATURE,

PIVOT, PLUGGABLE_MAPPING, POSTMAPPING_PROCESS, PREMAPPING_
PROCESS,

SEQUENCE, SET_OPERATION, SORTER, SPLITTER, TABLE, TABLE_FUNCTION,
TRANSFORMATION, UNPIVOT, VIEW.

connectionConditionClause

List objects only if they are connected from or to objects specified in the connection condition.

childType

Type of a child that belongs to map, mapping operator, mapping group or

mapping attribute.

getChildDetailClause

Get the desired detail of a child object that belongs to the mapping, map variable, mapping operator, mapping group or mapping attribute.

operatorLocator

Location of a mapping operator.

getOperatorDetailClause

Get the desired detail of a mapping operator.

mappableBottomUpLocator

Location of the object to be bound to a mapping mapping operator or mapping attribute.

propertyKey

A property key for an object.

Basic properties for MAPPING:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the mapping

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the mapping

Basic properties for OPERATOR:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the operator

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the operator

Basic properties for GROUP:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the group

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the group

Basic properties for ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the attribute

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE,

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Attribute

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Length of the attribute.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Precision of the attribute.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Scale of the attribute.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Properties for MAPPING:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

True if the map is deployable to a physical implementation

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: GENERATION_LANGUAGE

Type: STRING

Valid Values: ABAP, PLSQL, SQLLOADER, UNDEFINED

Default: UNDEFINED

The language used when generating code for the mapping.

Name: REFERRED_CALENDAR

Type: STRING

Valid Values: N/A

Default: "

Enter the Schedule to associate with this object.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for GROUP:

Name: COMMANDTYPE

Type: STRING

Valid Values: ALL_COMMAND_TYPES, DELETE, INSERT, UPDATE

Default: ALL_COMMAND_TYPES

The operation causing the change described by the LCR

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open record.

Name: DIMENSION_KEY

Type: STRING(32)

Valid Values: N/A

Default: "

The column in which dimension key value is to be stored when this is a target of star schema.

Name: EXPRESSION_INOUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXPRESSION_OUT

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: EXTRACTING_TYPE

Type: STRING

Valid Values: FROM_ALL, FROM_CURRENT

Default: FROM_CURRENT

The extracting operation to be performed when this is a source. If Extract Current Only (Type 2 Only) is specified, only current records will be extracted. If Extract All is specified, all records will be extracted.

Name: GROUP_TYPE

Type: STRING

Valid Values: REF_CURSOR, SCALAR

Default: SCALAR

This property specifies whether the input parameter is a scalar or a ref cursor type

Name: INTERNAL_TABLE

Type: STRING

Valid Values: N/A

Default: "

Internal staging table for this operator group

Name: LEVEL_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of this level.

Name: MODULENAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter group

Name: RECORD_TYPE_VALUES

Type: STRING
Valid Values: N/A
Default: "
Record Type Values.

Name: RETURN_TABLE_OF_SCALAR
Type: BOOLEAN
Valid Values: true, false
Default: false
This property specifies whether the return of the table function is a TABLE of SCALAR or not.

Name: ROW_LOCATOR
Type: STRING
Valid Values: N/A
Default: "
An expression indicating which attribute within the input group is the row locator.

Name: ROW_LOCATOR_VALUES
Type: STRING
Valid Values: N/A
Default: NULL, NULL
A comma-delimited expression that gives the possible values of the row locator within a unpivot group.

Name: SLOWLY_CHANGING_TYPE
Type: STRING
Valid Values: TYPE1, TYPE2, TYPE3
Default: TYPE2
The slowly changing type of this target.

Name: SPLIT_CONDITION
Type: STRING
Valid Values: N/A
Default: "
Condition that defines when to perform the attribute maps for the

attributes in this group.

Name: TABLENAME

Type: STRING

Valid Values: N/A

Default: "

The source table corresponding to the LCR

Name: TARGET_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The table name bound to this target.

Properties for ATTRIBUTE:

Name: ADDRESS_TYPE

Type: STRING

Valid Values: NA_ADDRTYPE_DUAL, NA_ADDRTYPE_NORMAL

Default: NA_ADDRTYPE_NORMAL

You can designate an address type as Normal or Dual. For example, a dual address occurs when a record contains both a street address and a P.O. Box; this is common with business data. A normal address contains only one type of address.

Name: ATTRIBUTE_ROLE

Type: STRING

Valid Values: END_DATE, LOOKUP_ACTIVE_DATE, MEASURE, NATURAL_KEY, NONE,

PARENT_NATURAL_KEY, PARENT_REF_KEY, PARENT_SURROGATE_KEY, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier

indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: BINDING_COLUMN_NAME

Type: STRING

Valid Values: N/A

Default: "

The binding column name for this attribute

Name: DATA_TYPE

Type: STRING

Valid Values: N/A

Default: "

The data type of the attribute

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

The Default Value for the function input parameter

Name: DIMENSION_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The dimension attribute referenced to by this level attribute.

Name: EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

The output expression for the attribute

Name: FIELD_DATA_TYPE

Type: STRING

Valid Values: BYTEINT, CHAR, DATE, DECIMAL, DECIMAL EXTERNAL, DOUBLE, FLOAT, FLOAT EXTERNAL, GRAPHIC, GRAPHIC EXTERNAL, INTEGER, INTEGER

EXTERNAL, INTEGER UNSIGNED, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH,

LONG VARRAW, RAW, SMALLINT, SMALLINT UNSIGNED, TIMESTAMP, TIMESTAMP WITH

LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHARC, VARGRAPHIC,

VARRAW, VARRAWC, ZONED, ZONED EXTERNAL

Default: CHAR

SQL Data Type of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

Field Length of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

Field Precision of the field in the file to which this operator is bound.

Name: FIELD_DATA_TYPE_SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

Field Scale of the field in the file to which this operator is bound.

Name: FIELD_DEFAULTIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is blank or zero,

based on the datatype.

Name: FIELD_END_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The ending position of the field in the file

Name: FIELD_MASK

Type: STRING

Valid Values: N/A

Default: "

The mask for the field

Name: FIELD_NULLIF_VALUE

Type: STRING

Valid Values: N/A

Default: "

The expression that indicates the value of the field is null

Name: FIELD_START_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The starting position of the field in the file

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data fractional seconds precision value of the attribute

Name: FUNCTION_RETURN

Type: BOOLEAN

Valid Values: true, false

Default: false

Specifies whether this output is the return value of this function

Name: GROUP_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

A boolean value to indicate whether this input attribute is a part of the unpivot group key.

Name: INPUT_ROLE

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_CITY, NA_COUNTRYCODE, NA_COUNTRYNAME, NA_FIRMNAME, NA_FIRSTNAME, NA_FIRSTPARTNAME, NA_LASTLINE,

NA_LASTLINE_2, NA_LASTNAME, NA_LASTPARTNAME, NA_LINE1, NA_LINE10, NA_LINE2,

NA_LINE3, NA_LINE4, NA_LINE5, NA_LINE6, NA_LINE7, NA_LINE8, NA_LINE9, NA_LOCALITYNAME, NA_LOCALITY_2, NA_LOCALITY_3, NA_LOCALITY_4,

NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME3, NA_NEIGHBORHOOD, NA_NONE,

NA_PASSTHRU, NA_PERSON, NA_PERSON2, NA_PERSON3, NA_POSTALCODE, NA_POSTNAME,

NA_PRENAME, NA_PRIMARYADDRESS, NA_SECONDARYADDRESS, NA_STATE

Default: NA_NONE

Assigns a name-address input role to the selected input attribute

Name: INSTANCE

Type: STRING

Valid Values: NA_INSTANCE_FIFTH, NA_INSTANCE_FIRST, NA_INSTANCE_FOURTH,

NA_INSTANCE_SECOND, NA_INSTANCE_SIXTH, NA_INSTANCE_THIRD

Default: NA_INSTANCE_FIRST

The instance option is used when an address contains multiple names, you can specify which name in the group should be used. In addition, you can use this option to assign an address type to a miscellaneous address component.

Name: IS_CAST_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: IS_OPTIONAL

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the input is not required to be connected

Name: IS_PREDEFINED_CONSTANT

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

The data length value of the attribute

Name: LEVEL_ATTRIBUTE_COLNAME_NAME

Type: STRING

Valid Values: N/A

Default: "

Column name in the AW staging table and source view for this attribute.

Name: LEVEL_ATTRIBUTE_LEVEL_INDICATOR_COLNAME

Type: STRING

Valid Values: N/A

Default: "

Level indicating columns name. This level will contain the name of the name of the level this parent reference belongs to. This is relevant only for skip level hierarchies.

Name: LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Level Attribute.

Name: LEVEL_RELATIONSHIP_NAME

Type: STRING

Valid Values: N/A

Default: "

The level relationship name associated to this attribute.

Name: LOAD_COLUMN_WHEN_INSERTING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the insert load operation.

Name: LOAD_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will participate in the update load operation.

Name: LOOKUP_ATTRIBUTE_ROLE

Type: STRING

Valid Values: DIMENSION_KEY, END_DATE, NATURAL_KEY, NONE, PREV_VALUE,

START_DATE, SURROGATE_KEY, TRIGGER

Default: NONE

The type of role to indicate how this attribute will be used in history logging. History As Previous Value indicates that this attribute will be used to keep previous value. Surrogate Identifier indicates that this attribute will be used to keep the surrogate identifier. Natural Identifier indicates that this attribute will be used to keep the natural identifier. Effective Time indicates that this attribute will be used as the effective time of the version. Expiration Time indicates that this attribute will be

used as the expiration time of the version. Trigger indicates that this attribute will be used to trigger history logging. If none is specified, this attribute will be used to keep current value.

Name: MATCHING_ROW

Type: NUMBER

Valid Values: 1 - 1000

Default: 1

An positive integer to indicate from which row within the unpivot group this output attribute obtains its data.

Name: MATCH_COLUMN_WHEN_DELETING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the delete load operation.

Name: MATCH_COLUMN_WHEN_UPDATING_ROW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether this attribute will be used to construct the matching criteria between the incoming data and the existing data on the target during the update load operation.

Name: MERGE_ATTR

Type: STRING

Valid Values: N/A

Default: "

Related merge attribute

Name: OUTPUT_COMPONENT

Type: STRING

Valid Values: NA_ADDRESS, NA_ADDRESS2, NA_ADDRESSCORRECTED, NA_ADDRESSTYPE,

NA_AUTO_ZONE_IND, NA_BOXNAME, NA_BOXNUMBER, NA_BUILDINGNAME, NA_CART,
NA_CBSA_CODE, NA_CBSA_DESC, NA_CENSUSID, NA_CHECKDIGIT, NA_CITY,
NA_CITYCORRECTED, NA_CITYMATCH, NA_CITYWARNING, NA_CITY_ABBREV,
NA_CITY_ABBREV_2, NA_CITY_ALTERNATE, NA_COMPLEX, NA_COUNTRYCODE,
NA_COUNTRYCODE3, NA_COUNTRYNAME, NA_COUNTYNAME, NA_DELIVERYBEATCODE,
NA_DELIVERYOFFICECODE, NA_DELIVERYPOINT, NA_EMAIL, NA_EXTRA_1,
NA_EXTRA_10,
NA_EXTRA_11, NA_EXTRA_12, NA_EXTRA_13, NA_EXTRA_14, NA_EXTRA_15,
NA_EXTRA_16, NA_EXTRA_17, NA_EXTRA_18, NA_EXTRA_19, NA_EXTRA_2,
NA_EXTRA_20, NA_EXTRA_3, NA_EXTRA_4, NA_EXTRA_5, NA_EXTRA_6, NA_EXTRA_7,
NA_EXTRA_8, NA_EXTRA_9, NA_FIPS, NA_FIPSCOUNTY, NA_FIPS_PLACE_CODE,
NA_FIRMCOUNT, NA_FIRMNAME, NA_FIRM_LOC, NA_FIRSTNAME, NA_FIRSTNAMESTD,
NA_GENDER, NA_GEO_MATCH_PREC, NA_INSTALLATIONNAME, NA_INSTALLATIONTYPE,
NA_ISADDRESSVERIFIABLE, NA_ISFOUND, NA_ISGOODADDRESS, NA_ISGOODGROUP,
NA_ISGOODNAME, NA_ISPARSED, NA_LACS, NA_LASTLINE, NA_LASTLINE_2,
NA_LASTNAME, NA_LATITUDE, NA_LOCALITYCODE, NA_LOCALITYNAME, NA_LOCALITY_2,
NA_LOCALITY_3, NA_LOCALITY_4, NA_LONGITUDE, NA_LOT, NA_LOT_ORDER, NA_MCD,
NA_MIDDLENAME, NA_MIDDLENAME2, NA_MIDDLENAME2STD, NA_MIDDLENAME3,
NA_MIDDLENAME3STD, NA_MIDDLENAMESTD, NA_MISCADDRESS, NA_MSA,
NA_NAMEDSIGNATOR, NA_NAMEWARNING, NA_NAME_FIRM_EXTRA, NA_NEIGHBORHOOD,
NA_NONAMBIGUOUSMATCH, NA_NONE, NA_NP_SEC_ADDR, NA_NP_UNIT_DESIG,
NA_NP_UNIT_NBR, NA_OTHERPOSTNAME, NA_PARSESTATUS, NA_PARSESTATUSDESC,
NA_PARSINGCOUNTRY, NA_PASSTHRU, NA_PERSON, NA_PERSONCOUNT, NA_PHONE,
NA_POSTALCODE, NA_POSTALCODECORRECTED, NA_POSTALCODEFORMATTED,
NA_POSTDIRECTIONAL, NA_POSTNAME, NA_PREDIRECTIONAL, NA_PRENAME,

NA_PRIMARYADDRESS, NA_PRIM_NAME_2, NA_RELATIONSHIP, NA_ROUTENAME,
NA_ROUTENUMBER, NA_SECONDARYADDRESS, NA_SSN, NA_STATE,
NA_STREETCOMPCORRECTED, NA_STREETCOMPMATCH, NA_STREETCORRECTED,
NA_STREETNAME, NA_STREETNAMEMATCH, NA_STREETNUMBER, NA_STREETNUMBERMATCH,
NA_STREETTYPE, NA_STREETWARNING, NA_TITLE, NA_UNITDESIGNATOR,
NA_UNITNUMBER, NA_URBANIZATIONNAME, NA_URBAN_IND, NA_ZIP4,
NA_ZIP5

Default: NA_NONE

Assigns a Name and Address output component to the selected output attribute.

Name: PARAMETER_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

The position of the argument in the table function signature corresponding to this parameter

Name: PIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

A comma-delimited expression that gives the input attribute to be used for each output row in the pivot group.

Name: PRECISION

Type: NUMBER

Valid Values: N/A

Default: 0

The data precision value of the attribute

Name: REFERENCED_LEVEL_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level attribute associated to this attribute.

Name: REFERENCED_LEVEL_NAME

Type: STRING

Valid Values: N/A

Default: "

The bound name of the parent level group associated to this attribute.

Name: REFERENCING_TYPE

Type: STRING

Valid Values: LOOKUP_KEY, NONE, REFERENCE_KEY_ALL, REFERENCE_KEY_ONLY

Default: NONE

The type of reference to indicate how this attribute participates in resolving existing level relationships and level implementations. If Lookup Reference Attribute is specified, this attribute will be used as lookup attribute upon parent level to resolve level implementations during loading. If Level Relationship Attribute (Snowflake) is specified, this attribute will be directly used as level relationship attribute and no lookup upon parent level would be performed during loading. If Level Relationship Attribute (Star) is specified, this attribute will be directly used as level relationship attribute, as well as lookup attribute upon parent level to resolve level implementations during loading. If none is specified, this attribute does not participate in any level relationship.

Name: SCALE

Type: NUMBER

Valid Values: N/A

Default: 0

The data scale value of the attribute

Name: SKIP_LEVEL_DIMENSION

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether this level has a skip level parent.

Name: TYPE_ATTRIBUTE_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the field of the PLS Record or attribute of the Object Type or column of the ROWTYPE that corresponds to this attribute. This property is not applicable if the return type is TABLE or SCALAR.

Name: UNPIVOT_EXPRESSION

Type: STRING

Valid Values: N/A

Default: NULL

An expression that gives the input attribute to be used as the output of this attribute.

Name: UPDATE_OPERATION

Type: STRING(3)

Valid Values: +=, -=, =, =-, =| |, | |=

Default: =

The computation to be performed on this attribute between the incoming data and the existing data on the target during the update load operation.

Name: VALUETYPE

Type: STRING

Valid Values: NEW, NEW_OLD, OLD

Default: NEW_OLD

Specifies the value type of this attribute

Properties for AGGREGATOR_OPERATOR:

Name: GROUP_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Group By clause for the aggregation

Name: HAVING_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Having clause for the aggregation

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for CONSTRUCT_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for CUBE_OPERATOR:

Name: ALLOW_PARALLEL_SOLVE

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, allow parallel solve when solving the cube.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CUBE_STORAGE_ISAW

Type: STRING

Valid Values: NO, YES

Default: NO

Indicates whether the storage for this cube in AW.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: INCREMENTAL_AGGREGATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether incremental aggregation should be done or full aggregation.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to insert all input data into target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: MAX_JOB_QUEUES_ALLOCATED

Type: NUMBER

Valid Values: 1 - 1000

Default: 0

The maximum number of job queues allocated when solving the cube.

Name: RUN_AGGREGATION

Type: STRING

Valid Values: NO, YES

Default: YES

A boolean value to indicate whether to solve the cube or not . Specify YES and cube data will be precomputed for the levels specified in the definition of cube

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Properties for DEDUPLICATOR_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for DIMENSION_OPERATOR:

Name: AW_LOAD_MODEL

Type: STRING

Valid Values: OTHER, SNOWFLAKE, STAR

Default: SNOWFLAKE

The loading model for the AW dimension - star, snowflake, or other.

Name: AW_STAGED_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the set-based AW load data is staged into a temporary table before loading into the AW.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DEFAULT_EFFECTIVE_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: SYSDATE

A date value to be served as the effective time of a newly created open record. The same date value is also served as the expiration time to close an existing open record.

Name: DEFAULT_EXPIRATION_TIME_OF_OPEN_RECORD

Type: STRING

Valid Values: N/A

Default: NULL

A date value to be served as the expiration time of a newly created open

record.

Name: DIMENSION_AWNAME

Type: STRING

Valid Values: N/A

Default: "

AW Name which contains this Dimension.

Name: DIMENSION_ISAW

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Flag to indicate whether storage is AW.

Name: LOADING_TYPE

Type: STRING

Valid Values: LOAD, REMOVE

Default: LOAD

The loading operation to be performed when this is a target. If LOAD is specified, OWB will try to match between the input data and target data to compute new data and existing data; it will then create new data, as well as modify existing data onto target. If REMOVE is specified, OWB will try to match between the input data and target data to compute existing data; it will then remove existing data from target.

Name: SLOWLY_CHANGING_TYPE

Type: STRING

Valid Values: TYPE1, TYPE2, TYPE3

Default: TYPE2

The slowly changing type of this target.

Name: SURROGATE_IDENTIFIER_LOADING_POLICY

Type: STRING

Valid Values: N/A

Default: "

The name of the sequence used to generate surrogate key values when loading the dimension.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRUNCATE_LOAD

Type: BOOLEAN

Valid Values: true, false

Default: false

The truncate flag to indicate whether all existing dimension values should be truncated before load begins (AW only).

Name: TYPE2_MATCH_CURRENT_ONLY

Type: STRING

Valid Values: NO, YES

Default: YES

If set to YES, only the current record will be used when performing the selected operation (remove or extract) for type 2 dimension.

Properties for EXPAND_OBJECT_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for EXTERNAL_TABLE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for FILTER_OPERATOR:

Name: FILTER_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The boolean filtering condition that identifies what data is to be processed. Any row with a false condition will be ignored.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for FLAT_FILE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by reconciliation for this item. Unlike other operators, it is not needed for generation. By default it is the same name as the item.

Name: CONCATENATE_RECORDS

Type: NUMBER

Valid Values: N/A

Default: 0

Number of Physical Records in a Logical Record.

Name: CONTINUATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the record is continued on the next line.

Name: CONTINUATION_CHARACTER_ON_NEXT_LINE

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

If there is a continuation character, is it at the start of the line.

Name: FIELD_ENCLOSURE_CHARACTERS

Type: STRING

Valid Values: N/A

Default: "

Characters that wrap fields. Example ' or ".

Name: FIELD_NAMES_IN_THE_FIRST_ROW

Type: BOOLEAN

Valid Values: true, false

Default: FALSE

Indicates whether file contains a header row.

Name: FIELD_TERMINATION_CHARACTER

Type: STRING

Valid Values: N/A

Default: ,

Character that separates the fields of a delimited file.

Name: FILE_FORMAT

Type: STRING

Valid Values: DELIMITED, FIXED

Default: DELIMITED

File Format (Fixed or Delimited).

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: INSERT, NONE, UPDATE

Default: INSERT

The loading operation to be performed

Name: OUTPUT_AS_XML

Type: BOOLEAN

Valid Values: true, false

Default: false

Output data to file in XML format.

Name: RECORD_DELIMITER

Type: STRING

Valid Values: N/A

Default: "

Character that indicates the end of the record.

Name: RECORD_SIZE

Type: NUMBER

Valid Values: N/A

Default: 0

Size of a fixed length record.

Name: RECORD_TYPE_LENGTH

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the length of the data that identifies the type of record. It is used with the Record Type Position.

Name: RECORD_TYPE_POSITION

Type: NUMBER

Valid Values: N/A

Default: 0

If this is a multi record file, this will indicate the position of the field that identifies the type of record.

Name: SAMPLED_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

The default name of the physical file to be used by sqlloader. If the file was sampled, the default was set from sampled file name.

Name: SOURCE_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The Location of the File Module of this Flat File at the time of reconciliation. Stored as UOID.

Name: TARGET_DATA_FILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access referenced entity.

Name: TARGET_DATA_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the target data file, including extension (file type). This name should not include the file path. To specify where the target data file will be created/appended, set the Target Data File Location.

Properties for ITERATOR_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for JOINER_OPERATOR:

Name: JOIN_CONDITION

Type: STRING

Valid Values: N/A

Default: "

The Join Condition for the join operator

Properties for KEY_LOOKUP_OPERATOR:

Name: BOUND_LEVEL

Type: STRING

Valid Values: N/A

Default: "

The level name of the bound level, if this is a dimension lookup.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: CREATE_NO_MATCH_ROW

Type: BOOLEAN

Valid Values: true, false

Default: true

If true, a row is created and the user-defined default values are used, in the case where no lookup match is found. If false, no row is produced.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOOKUP_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Key lookup condition based on the source inputs. This condition is used to
lookup a value in the bound table. If the condition is not met, the
default value expression will be returned. If a default expression is not
defined, null is used.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
WHERE clause for test data VIEW for this source or target

Name: TYPE2_HISTORY_LOOKUP_DATE
Type: STRING
Valid Values: N/A
Default: "
A date expression used when doing a lookup on a type 2 dimension level, to specify the historical date for which to retrieve data. If blank, the most current record is used.

Properties for LCRCAST_OPERATOR:

Name: BOUND_NAME
Type: STRING
Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Properties for MATCHMERGE_OPERATOR:

Name: MATCH_KEYS

Type: STRING

Valid Values: N/A

Default: "

Ordered list of attributes that control the set of records to be matched at any at any particular time.

Name: MATCH_NEW_RECORDS

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORDS:DESCRIPTION"

Name: MATCH_NEW_RECORD_CONDITION

Type: STRING

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "MATCHMERGE.GENERAL.MATCH_NEW_RECORD_CONDITION:DESCRIPTION"

Name: MERGED_PREFIX

Type: STRING

Valid Values: N/A

Default: "

Set the prefix used for the merged attributes in the cross-reference group.

Properties for MATERIALIZED_VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

'Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Properties for NAME_AND_ADDRESS_OPERATOR:

Name: ADDRESS_LINE_1

Type: STRING

Valid Values: N/A

Default: "

Address line 1

Name: ADDRESS_LINE_2

Type: STRING

Valid Values: N/A

Default: "

Address line 2

Name: ADDRESS_LINE_3

Type: STRING

Valid Values: N/A

Default: "

Address line 3

Name: ADDRESS_LINE_4

Type: STRING

Valid Values: N/A

Default: "

Address line 4

Name: DUAL_ADDRESS_ASSIGNMENT

Type: STRING

Valid Values: NA_DUALADDR_CLOSESTTOLASTLINE, NA_DUALADDR_POBOX,
NA_DUALADDR_STREET

Default: NA_DUALADDR_STREET

A dual address refers to two address lines for the same destination. For example, a record contains both a street address and a P.O. Box; this is common with business data. Select which of the two address lines should be assigned in these cases.

Name: GENERATE_CASS_REPORT

Type: STRING

Valid Values: NA_NO, NA_YES

Default: NA_NO

Select "Yes" to generate CASS (Coding Accuracy Support System) report. CASS report is a text file specified by the United States Postal Service. The report is written to the [nas/bin/admin/reports] folder under the home folder of the name/address server.

Name: LIST_NAME

Type: STRING

Valid Values: N/A

Default: "

The list name is optional and provides a reference for tracking multiple CASS reports.

Name: PARSING_TYPE

Type: STRING

Valid Values: NA_ADDRESSONLY, NA_NAMEANDADDRESS, NA_NAMEONLY

Default: NA_NAMEANDADDRESS

Select a name-address parsing type to be performed on the input data

Name: PRIMARY_COUNTRY

Type: STRING

Valid Values: NA_AND, NA_ARE, NA_ARG, NA_AUS, NA_AUT, NA_BEL, NA_BGD,

NA_BGR, NA_BHS, NA_BLZ, NA_BMU, NA_BRA, NA_BRB, NA_BRN, NA_CAN, NA_CHE,

NA_CHL, NA_CHN, NA_COL, NA_CZE, NA_DEU, NA_DNK, NA_EGY, NA_ESP, NA_EST,

NA_FIN, NA_FRA, NA_GBR, NA_GRC, NA_GUM, NA_HKG, NA_HUN, NA_ICL, NA_IND,

NA_IRL, NA_IRN, NA_IRQ, NA_ISR, NA_ITA, NA_JAM, NA_JOR, NA_JPN, NA_KHM,

NA_KOR, NA_LIE, NA_LTU, NA_LUX, NA_LVA, NA_MEX, NA_MYS, NA_NLD, NA_NOR,

NA_NZL, NA_PAK, NA_PER, NA_PHL, NA_POL, NA_PRT, NA_ROM, NA_RUS, NA_SGP,

NA_SVN, NA_SWE, NA_THA, NA_UKR, NA_USA, NA_VEN, NA_YUG, NA_ZAF

Default: NA_USA

Select the primary parsing country which best represents the input data.

Input addresses having the same country as the primary parsing country will only need to be parsed once. Input addresses having a different country than the primary parsing country may be reparsed by a different parser. For performance reasons, it is best to minimize the percentage of 2-pass parses by selecting the optimal parser.

Name: PROCESSOR_NAME

Type: STRING

Valid Values: N/A

Default: "

The processor name is the name of the organization submitting the CASS report.

Properties for PIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: PIVOT_GROUP_SIZE

Type: NUMBER

Valid Values: 1 - 1000

Default: 2

A number specifying the pivot group size for the pivot operation. Pivot group size determines the number of output rows that are produced from each input row.

Properties for PLUGGABLE_MAPPING_OPERATOR:

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Properties for POSTMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: POSTMAPPING_PROCESS_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS, ON_WARNING

Default: ON_SUCCESS

Indicates under what condition of the mapping the post-mapping process will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for PREMAPPING_PROCESS_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: MAPPING_RUN_CONDITION

Type: STRING(10)

Valid Values: ALWAYS, ON_ERROR, ON_SUCCESS

Default: ON_SUCCESS

Indicates under what condition of the pre-mapping process the mapping will be run.

Name: ROW-BASED_ONLY

Type: BOOLEAN

Valid Values: true, false

Default: false

Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for QUEUE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Properties for SEQUENCE_OPERATOR:

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for SET_OPERATION_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: SET_OPERATION

Type: STRING

Valid Values: INTERSECT, MINUS, UNION, UNIONALL

Default: UNION

Specifies the set operation that is to be performed by this operator.

Properties for SORTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Name: ORDER_BY_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

The Order By Clause

Properties for SPLITTER_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for TABLE_FUNCTION_OPERATOR:

Name: TABLE_FUNCTION_IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

Description not available.

Name: TABLE_FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

The name of the Table Function

Properties for TABLE_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Description not available.

Name: CONFLICT_RESOLUTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Detect and resolve any conflicts that may arise during DML using the LCR APIs

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DATA_COLLECTION_FREQUENCY

Type: STRING(16)

Valid Values: DAY, HOUR, MINUTE, MONTH, QUARTER, UNKNOWN, YEAR

Default: UNKNOWN

New Data Granularity

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: DIRECT

Type: BOOLEAN

Valid Values: true, false

Default: false

Directly swap source into target as a partition without first creating a staging table.

Name: ENABLE_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Enable Constraints

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EVALUATE_CHECK_CONSTRAINTS

Type: BOOLEAN

Valid Values: true, false

Default: false

Evaluate check constraints

Name: EXCEPTIONS_TABLE_NAME

Type: STRING

Valid Values: N/A

Default: "

Exceptions Table Name

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: JOINRANK

Type: FLOAT

Valid Values: N/A

Default: 0

Join Rank

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PEL_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

PEL Enabled

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: REPLACE_DATA

Type: BOOLEAN

Valid Values: true, false

Default: false

Replace existing data in target partition if there is any.

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA

Type: STRING(30)

Valid Values: N/A

Default: "

Schema

Name: SINGLEROW

Type: BOOLEAN

Valid Values: true, false

Default: false

Singlerow

Name: SORTED_INDEXES_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

Sorted Indexes Clause

Name: SUBPARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Subpartition Name

Name: TARGET_FILTER_FOR_DELETE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE

Type: STRING

Valid Values: N/A

Default: "

A condition on the rows in the target and if evaluated to true, that row will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Name: USE_LCR_API

Type: BOOLEAN

Valid Values: true, false

Default: true

Use LCR APIs if possible to perform the DML

Properties for TRANSFORMATION_OPERATOR:

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: FUNCTION_NAME

Type: STRING

Valid Values: N/A

Default: "

Name of the transformation to be called.

Name: IS_TARGET

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, then the function is a target

Name: RETURN_TYPE

Type: STRING
Valid Values: N/A
Default: "
Description not available.

Name: ROW-BASED_ONLY
Type: BOOLEAN
Valid Values: true, false
Default: false
Indicates if this transformation must be used only Row Based mode. Some transformations can be used in SQL mode as well as Row Based mode.

Name: ROW_COUNT
Type: STRING
Valid Values: N/A
Default: "
Row count

Name: ROW_COUNT_ENABLED
Type: BOOLEAN
Valid Values: true, false
Default: false
Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: TEST_DATA_COLUMN_LIST
Type: STRING
Valid Values: N/A
Default: "
Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Properties for UNPIVOT_OPERATOR:

Name: INLINEVIEW_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used on inline view when extracting using SQL

Properties for VIEW_OPERATOR:

Name: ADVANCED_MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: NO_CONSTRAINTS

This property is used by public API and scripting to influence how columns are used for UPDATE or DELETE DMLs. If this property is set with the name of a primary or unique key, all the columns in the key will be used for matching during UPDATE or DELETE; and all the columns not in the key are used for loading. The property can also be assigned the value "All constraints" or "No constraints". If the DML type is INSERT, TRUNCATE/INSERT, or CHECK/INSERT, setting this property causes no effect.

Name: BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

The name to be used by the code generator to identify this item. By default it is the same physical name as the item.

Name: DATABASE_FILE_NAME

Type: STRING

Valid Values: N/A

Default: "

Database file name to allocate extents from

Name: DATABASE_LINK

Type: STRING(128)

Valid Values: N/A

Default: "

The database link used to access this entity during mapping.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

The location used to access this entity during mapping.

Name: DEBUG_BOUND_NAME

Type: STRING

Valid Values: N/A

Default: "

Physical name used to bind to a physical entity during a debug session

Name: DEBUG_DB_LOCATION

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key
"8i.MAPPING.ENTITY.DEBUGGERPARAMS.LOCATION:DESCRIPTION"

Name: ERROR_SELECT_FILTER

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Rows selected from the error table will contain only errors created by this
operator in this map execution

Name: ERROR_SELECT_ROLL_UP

Type: STRING(3)

Valid Values: NO, YES

Default: YES

Records selected from the error table will be rolled up by the error name, so all errors generated by a particular input record will be rolled up into a single record with the error names concatenated in the error name attribute.

Name: ERROR_TABLE_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

The error table name of this target to log invalid records.

Name: EXTRACTION_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when extracting from this table using SQL

Name: LOADING_HINT

Type: STRING

Valid Values: N/A

Default: "

Hint used when loading into this table using SQL

Name: LOADING_TYPE

Type: STRING(16)

Valid Values: CHECK_INSERT, DELETE, DELETE_INSERT, INSERT, INSERT_UPDATE,

NONE, TRUNCATE_INSERT, UPDATE, UPDATE_INSERT

Default: INSERT

The loading operation to be performed when this is a target.

Name: MATCH_BY_CONSTRAINT

Type: STRING

Valid Values: N/A

Default: ALL_CONSTRAINTS

A property to indicate whether unique or primary key information on this target will override the matching criteria obtained from Match by constraint property on the attributes of this target.

Name: PARTITION_NAME

Type: STRING

Valid Values: N/A

Default: "

Partition Name

Name: PRIMARY_SOURCE

Type: STRING

Valid Values: NO, YES

Default: NO

A boolean value to indicate whether this is a primary source or not (only used in EDW).

Name: RECORDS_TO_SKIP

Type: NUMBER

Valid Values: >= 0

Default: 0

Number of records to skip

Name: ROW_COUNT

Type: STRING

Valid Values: N/A

Default: "

Row count

Name: ROW_COUNT_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: false

Row count enabled

Name: SCHEMA
Type: STRING(30)
Valid Values: N/A
Default: "
Schema

Name: SINGLEROW
Type: BOOLEAN
Valid Values: true, false
Default: false
Singerow

Name: SORTED_INDEXES_CLAUSE
Type: STRING
Valid Values: N/A
Default: "
Sorted Indexes Clause

Name: SUBPARTITION_NAME
Type: STRING
Valid Values: N/A
Default: "
Subpartition Name

Name: TARGET_FILTER_FOR_DELETE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row will participate in the delete loading operation.

Name: TARGET_FILTER_FOR_UPDATE
Type: STRING
Valid Values: N/A
Default: "
A condition on the rows in the target and if evaluated to true, that row

will participate in the update loading operation.

Name: TARGET_LOAD_ORDER

Type: STRING(65535)

Valid Values: N/A

Default: "

The Target Load Order property enables you to determine the order in which multiple targets within the same mapping get loaded. Warehouse Builder determines a default order based on the FK relationships. However, using the property you can overrule that default order.

Name: TEST_DATA_COLUMN_LIST

Type: STRING

Valid Values: N/A

Default: "

Column list for test data VIEW for this source or target

Name: TEST_DATA_WHERE_CLAUSE

Type: STRING

Valid Values: N/A

Default: "

WHERE clause for test data VIEW for this source or target

Name: TRAILING_NULLCOLS

Type: BOOLEAN

Valid Values: true, false

Default: false

Trailing Nullcols

Name: TRUNCATE_ERROR_TABLE

Type: STRING(3)

Valid Values: NO, YES

Default: NO

If YES, then the error table will be truncated prior to use. This is ignored if there are no active data rules applied to the object.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

childName

Name of a child that belongs to map, mapping operator, mapping group or mapping attribute.

pluggableMapLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

operatorName

Name of a mapping operator.

groupDirection

Direction of a mapping group.

groupLocator

Location of a mapping group.

getGroupDetailClause

Get the desired detail of a mapping group.

operatorBottomUpLocator

Location of a mapping operator.

groupBottomUpLocator

Location of a mapping group.

attributeBottomUpLocator

Location of a mapping attribute.

pluggableMapName

Name of the pluggable map.

groupName

Name of a mapping group.

attributeLocator

Location of a mapping attribute.

getAttributeDetailClause

Get the desired detail of a mapping attribute.

pluggableMapBottomUpLocator

Location of a child pluggable mapping within a mapping or another pluggable mapping.

attributeName

Name of a mapping attribute.

Examples

```
OMBRETRIEVE REAL_TIME_MAPPING 'MAP1' GET OPERATORS
```

```
OMBRETRIEVE REAL_TIME_MAPPING 'MAP1' GET VARIABLES
```

```
OMBRETRIEVE REAL_TIME_MAPPING 'MAP1' OPERATOR 'SRC1' GROUP  
'INOUTGRP1'
```

```
GET ATTRIBUTE CONNECTED TO OPERATOR 'target1'
```

```
OMBRETRIEVE REAL_TIME_MAPPING 'MAP1' OPERATOR 'SRC1'
```

```
GET PROPERTIES (BUSINESS_NAME, DESCRIPTION)
```

```
OMBRETRIEVE REAL_TIME_MAPPING 'MAP1' VARIABLE 'LAST_CUST'
```

```
GET PROPERTIES (BUSINESS_NAME, DATATYPE)
```

See Also

```
OMBRETRIEVE, OMBCREATE REAL_TIME_MAPPING, OMBALTER REAL_TIME_  
MAPPING, OMBDROP REAL_TIME_MAPPING
```

OMBRETRIEVE REGISTERED_FUNCTION

Purpose

Retrieve details of a function.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
retrieveRegisteredFunctionCommand = OMBRETRIEVE REGISTERED_FUNCTION
    "QUOTED_STRING" ( "retrieveFunctionClause" |
    "retrieveFunctionArgClause" )
retrieveFunctionClause = GET ( "getPropertiesClauseforRegFun" |
    "getReferenceIconSetClause" | "getFunctionSCOClaue" )
retrieveFunctionArgClause = PARAMETER "QUOTED_STRING" GET
    "getPropertiesClause"
getPropertiesClauseforRegFun = PROPERTIES "(" "propertyNameListforRegFun"
    ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getFunctionSCOClaue = PARAMETERS | DERIVATION_SOURCE
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameListforRegFun = ( "UNQUOTED_STRING" | ( PACKAGE | SIGNATURE )
    ) { ", " ( "UNQUOTED_STRING" | ( PACKAGE | SIGNATURE ) ) }
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveRegisteredFunctionCommand

To retrieve a function.

QUOTED_STRING

name of the function.

retrieveFunctionClause

Retrieves the contents of the function.

GET

For registered function this clause retrieves the following

PARAMETERS retrieves the list of parameters owned by this registered function.

DERIVATION_SOURCE retrieves the component that the registered function was derived from.

retrieveFunctionArgClause

Retrieves properties of the function parameter.

getPropertiesClauseforRegFun

Retrieves the properties of the object.

getReferenceIconSetClause

Get specified Icon Set.

getFunctionSCOClause

Retrieves the parameters of the function.

getPropertiesClause

Retrieves the properties of the object.

Basic properties for REGISTERED_FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the function

Name: AVAILABLE

Type: Boolean

Valid Values: Y,N

Default: 'N'

Whether the Function is available for the user to use in calculations

Name: RETURN_TYPE

Type: STRING()

Valid Values: BINARY_INTEGER, BLOB, BOOLEAN, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH
NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, SYS.ANYDATA,
SYS.ROW_LCR, SYS_REFCURSOR, TIMESTAMP, TIMESTAMP WITH LOCAL TIME
ZONE

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE,
SYS.XMLFORMAT,

BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
SYS.XMLSEQUENCETYPE,

BLAST_MATCH_PLSQLRECORDTYPE

Default: 'NUMBER'

Return type of the function

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the parameter

Name: DATATYPE

Type: STRING()

Valid Values: BINARY_INTEGER, BLOB, BOOLEAN, CHAR, CLOB, DATE, FLOAT,
INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH

NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, SYS.ANYDATA,
SYS.ROW_LCR, SYS_REFCURSOR, TIMESTAMP, TIMESTAMP WITH LOCAL TIME
ZONE

TIMESTAMP WITH TIME ZONE, VARCHAR, VARCHAR2, XMLTYPE,
SYS.XMLFORMAT,

BLAST_ALIGN_PLSQLRECORDTYPE

SYS.LCR\$_ROW_RECORD, BLAST_SQL_TABLE_OF_NUMBERS,
SYS.XMLSEQUENCETYPE,

BLAST_MATCH_PLSQLRECORDTYPE

Default: 'NUMBER'

Datatype of the parameter

Properties for REGISTERED_FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DB_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location for the referenced Function

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: PACKAGE

Type: STRING

Valid Values: N/A

Default: "

May be used to identify the name of a Package that contains the Function

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameListforRegFun

This is the list of property names.

propertyNameList

This is the list of property names.

Examples

```
OMBRETRIEVE REGISTERED_FUNCTION 'My_Sum' GET  
PROPERTIES(DESCRIPTION)
```

See Also

OMBALTER REGISTERED_FUNCTION, OMBCREATE REGISTERED_FUNCTION

OMBRETRIEVE ROLE

Purpose

To retrieve properties of a Warehouse Builder role.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
retrieveRoleCommand = OMBRETRIEVE ROLE "QUOTED_STRING" ( GET
    "getPropertiesClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveRoleCommand

This clause retrieves properties of a Warehouse Builder role.

getPropertiesClause

Basic properties for ROLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the role

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the role

Examples

```
OMBRETRIEVE ROLE 'DEVELOPMENT_ROLE' GET PROPERTIES(BUSINESS_
NAME,
DESCRIPTION)
```

See Also

OMBCREATE ROLE, OMBALTER ROLE, OMBDROP ROLE

OMBRETRIEVE SAP_MODULE

Purpose

Retrieve details of the SAP module.

Prerequisites

You must open a project to retrieve a SAP module.

Syntax

```
retrieveSAPModuleCommand = OMBRETRIEVE SAP_MODULE "QUOTED_STRING" (  
    "getPropertiesClause" | "getReferenceLocationClause" |  
    "getReferenceDefaultLocationClause" |  
    "getReferenceMetadataLocationClause" | "getReferenceIconSetClause" |  
    "getReferenceLocationsClause" )  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
getReferenceLocationClause = GET ( REF | REFERENCE ) LOCATION  
getReferenceDefaultLocationClause = GET ( REF | REFERENCE ) DEFAULT  
    LOCATION  
getReferenceMetadataLocationClause = GET ( REF | REFERENCE )  
    METADATA_LOCATION  
getReferenceIconSetClause = GET ( REF | REFERENCE ) ICONSET  
getReferenceLocationsClause = GET ( REF | REFERENCE ) LOCATIONS  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveSAPModuleCommand

Retrieve the details of an SAP Module

getPropertiesClause

Retrieve a set of properties that is associated with an SAP Module.

Base properties for SAP_MODULE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: NAME

Business name of a SAP Module

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of an SAP Module

`getReferenceLocationClause`

Retrieve the name of the runtime location referenced by this SAP module.

`getReferenceDefaultLocationClause`

Retrieve the default runtime location referenced by this SAP module.

`getReferenceMetadataLocationClause`

Retrieve the metadata location referenced by this SAP module.

`getReferenceIconSetClause`

Retrieve the icon set referenced by this SAP module.

`getReferenceLocationsClause`

Retrieve the runtime locations referenced by this SAP module.

`propertyNameList`

Comma separated list of property names. Property names are unquoted.

Examples

```
OMBRETRIEVE SAP_MODULE 'src_module' GET PROPERTIES (DESCRIPTION,  
BUSINESS_NAME)
```

This will retrieve the SAP module "src_module"'s description and business name.

See Also

OMBRETRIEVE

OMBRETRIEVE SEQUENCE

Purpose

To retrieve properties of a sequence.

Prerequisites

In the context of an Oracle Module

Syntax

```
retrieveSequenceCommand = OMBRETRIEVE SEQUENCE "QUOTED_STRING" ( GET (
    "getPropertiesClause" | "getReferenceIconSetClause" ) |
    "retrieveColumnClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
retrieveColumnClause = COLUMN "QUOTED_STRING" GET "getPropertiesClause"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveSequenceCommand

This clause retrieves properties of a sequence.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for SEQUENCE:

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the sequence.

Name: CURRVAL

Type: NUMBER

Valid Values: N/A

Default: 1

current increment value.

Name: NEXTVAL

Type: NUMBER

Valid Values: N/A

Default: 1

next increment value. next increment value.

Properties for SEQUENCE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: INCREMENT_BY

Type: NUMBER

Valid Values: -2147483648 - 2147483647

Default: 1

Sequence Incremented By

Name: START_WITH

Type: NUMBER

Valid Values: -2147483648 - 2147483647

Default: 1

Sequence Starts With

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

retrieveColumnClause

This clause will retrieve columns.

QUOTED_STRING

Name of the column.

propertyNameList

The list of properties.

Examples

OMBRETRIEVE SEQUENCE 'NEW_SEQUENCE' GET PROPERTIES (DESCRIPTION)

This will retrieve its description.

See Also

OMBRETRIEVE, OMBCREATE SEQUENCE, OMBALTER SEQUENCE, OMBDROP SEQUENCE

OMBRETRIEVE SNAPSHOT

Purpose

Since the snapshot may contain many components, this command lets the user view all the contents in a snapshot.

Prerequisites

Snapshot contents can be retrieved from any context.

Syntax

```
parseRetrieveCommand = OMBRETRIEVE "retrieveSnapshotCommand"
retrieveSnapshotCommand = ( SNAPSHOT "QUOTED_STRING" [ GET
    "getPropertiesClause" ] )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

parseRetrieveCommand

Root production of OMBRETRIEVE SNAPSHOT.

retrieveSnapshotCommand

To view contents of snapshot.

QUOTED_STRING

Name of snapshot whose contents are to be retrieved.

getPropertiesClause

Gets the property of snapshot which are DESCRIPTION, TYPE.

Basic properties for SNAPSHOT:

Name: TYPE

Type: STRING(200)

Valid Values: FULL,SIGNATURE

Default: FULL

This is the type of snapshot

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the snapshot

PROPERTIES

Valid set of properties are DESCRIPTION and TYPE.

propertyNameList

Property names for SNAPSHOT that can be retrieved.

Examples

OMBRETRIEVE SNAPSHOT 'S1'

This command gets all the contents of snapshot.

OMBRETRIEVE SNAPSHOT 'S1' GET
PROPERTIES(DESCRIPTION,TIMESTAMP,TYPE)

This command gets the properties DESCRIPTION,TIMESTAMP, and TYPE of snapshot S1.

See Also

OMBCREATE SNAPSHOT, OMBALTER SNAPSHOT, OMBDROP SNAPSHOT,
OMBRESTORE SNAPSHOT, OMBCOMPARE SNAPSHOT, OMBLIST SNAPSHOT

OMBRETRIEVE STREAMS_CAPTURE_PROCESS

Purpose

Retrieve details of the Streams Capture Process.

Prerequisites

Should be in the context of a Streams Queue.

Syntax

```
retrieveCaptureCommand = OMBRETRIEVE STREAMS_CAPTURE_PROCESS
    "QUOTED_STRING" ( GET ( "getCapturePropertiesClause" | TABLES ) )
getCapturePropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveCaptureCommand

Retrieves the details of the Streams Capture Process with the given name.

getCapturePropertiesClause

Get specified properties of the Streams Capture Process

propertyNameList

The list of properties.

Basic properties for STREAMS_CAPTURE_PROCESS:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Streams Capture Process

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Streams Capture

Properties for STREAMS_CAPTURE_PROCESS:

Name: CAPTURE_START_PARAMETER

Type: STRING

Valid Values: START_DATE, START_SCN

Default: START_SCN

This specifies whether the Streams Capture Process should start capturing changes based on the Start Date or the Start SCN.

Name: CAPTURE_TAGGED_LCR

Type: BOOLEAN

Valid Values: true, false

Default: false

If TRUE, then a redo entry is always considered for capture and an LCR is always considered for apply, regardless of whether redo entry or LCR has a non-NULL tag. If FALSE, then a redo entry is considered for capture and an LCR is considered for apply only when the redo entry or the LCR contains a NULL tag.

Name: CAPTURE_TIMEOUT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The maximum number of seconds to wait for another instance of the same capture process to finish.

Name: DBA_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Location of the DBA user who should create the supplemental logs.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Indicates whether the Object is deployable or not.

Name: DISABLE_ON_LIMIT

Type: BOOLEAN

Valid Values: true, false

Default: true

If set to true, the capture process will be disabled once the message/time limit is reached.

Name: MAXIMUM_SCN

Type: NUMBER

Valid Values: 0 - 1000000000

Default: 0

This is the Maximum SCN value whose corresponding changes will be captured by the Streams Capture Process.

Name: MESSAGE_COUNT_LIMIT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

If Disable On Limit is set to true, then the Streams Capture Process is disabled once the specified number of messages have been captured.

Name: PARALLELISM_DEGREE

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

The number of parallel server process that will mine the redo logs.

Name: START_DATE

Type: STRING

Valid Values: N/A

Default: 1970-01-01

The user specified date from which the Streams Capture Process should start capturing changes.

Name: START_SCN

Type: NUMBER

Valid Values: N/A

Default: 0

The user specified SCN from which the Streams Capture Process should start capturing changes.

Name: TIME_LIMIT

Type: NUMBER

Valid Values: 0 - 1000

Default: 0

If Disable On Limit is set to true, then the Streams Capture Process is disabled once the specified number of seconds elapse.

Name: WRITE_ALERT_LOG

Type: BOOLEAN

Valid Values: true, false

Default: true

If set to true, then the Streams Capture Process writes a message to the alert log on exit.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

Examples

```
OMBRETRIEVE STREAMS_CAPTURE_PROCESS 'SOME_CAPTURE_PROCESS' GET  
PROPERTIES
```

```
(DISABLE_ON_LIMIT,MAXIMUM_SCN, MESSAGE_COUNT_LIMIT,  
PARALLELISM_DEGREE,
```

```
CAPTURE_TIMEOUT, TIME_LIMIT, WRITE_ALERT_LOG, CAPTURE_START_  
PARAMETER,
```

```
START_SCN, CAPTURE_TAGGED_LCR)
```

This will retrieve the Streams Capture Process "SOME_CAPTURE_PROCESS"'s properties.

See Also

OMBRETRIEVE, OMBALTER STREAMS_CAPTURE_PROCESS, OMBCREATE STREAMS_CAPTURE_PROCESS, OMBDROP STREAMS_CAPTURE_PROCESS

OMBRETRIEVE STREAMS_QUEUE

Purpose

Retrieve details of the Streams Queue.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
retrieveANYQCommand = OMBRETRIEVE STREAMS_QUEUE "QUOTED_STRING" ( GET
    "getPropertiesClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveANYQCommand

Retrieves the details of the Streams Queue with the given name.

getPropertiesClause

Retrieves the values of the given Properties for the Streams Queue with the given name.

Basic properties for **ADVANCED_QUEUE**:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Advanced Queue

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Advanced Queue

Name: QTABLE

Type: STRING(4000)

Valid Values: N/A

Default: "

Queue Table for the Advanced Queue. This has to be the name of a Queue Table(Queue_Table) existing in the same Oracle Module.

Properties for STREAMS_QUEUE:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those objects marked as Deployable = true.

Name: DEQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Dequeue Enabled for AQ

Name: ENQUEUE_ENABLED

Type: BOOLEAN

Valid Values: true, false

Default: true

Enqueue enabled for AQ

Name: GENERATE_ADVANCED_QUEUE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the Advanced Queue.

Name: GENERATE_TEMPORARY_TABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Generate the code to create the temporary table.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: MAX_RETRIES

Type: NUMBER

Valid Values: N/A

Default: 5

Max. number of Retries

Name: RETENTION_TIME

Type: NUMBER

Valid Values: N/A

Default: 0

Message retention time

Name: RETRY_DELAY

Type: NUMBER

Valid Values: N/A

Default: 0

Delay period before retry

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE STREAMS_QUEUE 'SOME_STREAMS_QUEUE' GET PROPERTIES
(MAX_RETRIES,
RETRY_DELAY, RETENTION_TIME, ENQUEUE_ENABLED, DEQUEUE_ENABLED,
QTABLE)
```

This will retrieve the Streams Queue "SOME_STREAMS_QUEUE"'s properties.

See Also

OMBRETRIEVE, OMBALTER STREAMS_QUEUE, OMBCREATE STREAMS_QUEUE,
OMBDROP STREAMS_QUEUE

OMBRETRIEVE TABLE

Purpose

To retrieve properties of a table.

Prerequisites

In the context of an Oracle Module

Syntax

```

retrieveTableCommand = OMBRETRIEVE TABLE "QUOTED_STRING" (
    "retrieveTableClause" | "retrieveColumnClause" | "retrieveUkPkClause"
    | "retrieveFkClause" | "retrieveCheckConstraintClause" |
    "retrievePartitionConfigurationClause" |
    "retrievePartitionKeyConfigurationClause" |
    "retrieveTemplateSubpartitionConfigurationClause" |
    "retrieveSubPartitionConfigurationClause" |
    "retrieveSubPartitionKeyConfigurationClause" |
    "retrieveIndexConfigurationClause" | "retrieveDataRuleUsageClause" )
retrieveTableClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getTableSCOClaue" )
retrieveColumnClause = COLUMN "QUOTED_STRING" GET "getPropertiesClause"
retrieveUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" GET (
    "getPropertiesClause" | COLUMNS )
retrieveFkClause = FOREIGN_KEY "QUOTED_STRING" GET ( "getPropertiesClause"
    | COLUMNS | UNIQUE_KEY | PRIMARY_KEY | REFERENCED_KEY )
retrieveCheckConstraintClause = CHECK_CONSTRAINT "QUOTED_STRING" GET
    "getPropertiesClause"
retrievePartitionConfigurationClause = PARTITION "QUOTED_STRING" GET
    "getConfigurationPropertiesClause"
retrievePartitionKeyConfigurationClause = PARTITION_KEY "QUOTED_STRING"
    GET "getConfigurationPropertiesClause"
retrieveTemplateSubpartitionConfigurationClause = TEMPLATE_SUBPARTITION
    "QUOTED_STRING" GET "getConfigurationPropertiesClause"
retrieveSubPartitionConfigurationClause = SUBPARTITION "QUOTED_STRING" OF
    PARTITION "QUOTED_STRING" GET "getConfigurationPropertiesClause"
retrieveSubPartitionKeyConfigurationClause = SUBPARTITION_KEY
    "QUOTED_STRING" GET "getConfigurationPropertiesClause"
retrieveIndexConfigurationClause = INDEX "QUOTED_STRING" ( GET (
    "getConfigurationPropertiesClause" | INDEX_COLUMNS | INDEX_PARTITIONS
    | INDEX_PARTITION_KEYS ) | ( INDEX_COLUMN "QUOTED_STRING" GET
    "getConfigurationPropertiesClause" ) | ( INDEX_PARTITION_KEY
    "QUOTED_STRING" GET "getConfigurationPropertiesClause" ) | (
    INDEX_PARTITION "QUOTED_STRING" GET "getConfigurationPropertiesClause"
    ) )
retrieveDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" ( GET (
    "getPropertiesClause" | GROUPS ) | GROUP "QUOTED_STRING" ( GET (
    "getPropertiesClause" | ATTRIBUTES | REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) ) | ATTRIBUTE "QUOTED_STRING" (
    GET ( "getPropertiesClause" | REF COLUMN ) ) ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getTableSCOClaue = COLUMNS | UNIQUE_KEYS | PRIMARY_KEY | FOREIGN_KEYS |
    CHECK_CONSTRAINTS | COLUMN AT POSITION "INTEGER_LITERAL" | INDEXES |
    INDEX_PARTITION_KEYS OF INDEX "QUOTED_STRING" | INDEX_PARTITIONS OF
    INDEX "QUOTED_STRING" | PARTITIONS | SUBPARTITIONS OF PARTITION
    "QUOTED_STRING" | TEMPLATE_SUBPARTITIONS | PARTITION_KEYS |

```

```
SUBPARTITION_KEYS | DATA_RULE_USAGES
getConfigPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveTableCommand

This clause retrieves a table.

QUOTED_STRING

Name of the table.

retrieveTableClause

This clause retrieves a table.

retrieveColumnClause

This clause will retrieve columns.

QUOTED_STRING

Name of the column.

retrieveUkPkClause

This clause will retrieve a unique key or primary key.

QUOTED_STRING

Name of the unique key or the primary key.

retrieveFkClause

This clause will retrieve a key referenced by a foreign key, either a unique key or primary key. Use REFERENCED_KEY to retrieve the referenced key for a foreign key regardless of the type of referenced key (unique or primary).

QUOTED_STRING

Name of the foreign key.

retrieveCheckConstraintClause

This clause gets the check constraint.

QUOTED_STRING

Name of the check constraint.

retrievePartitionConfigurationClause

Gets the partition.

QUOTED_STRING

The partition name.

retrievePartitionKeyConfigurationClause

This clause gets the partition key.

QUOTED_STRING

The name of the partition key.

retrieveIndexConfigurationClause

Gets the index in this clause.

QUOTED_STRING

Name of the index.

retrieveDataRuleUsageClause

This clause retrieves the data rule usages.

QUOTED_STRING

Name of data rule usage, group or attribute.

GROUPS

Retrieve the names of all relation groups in the data rule usage.

ATTRIBUTES

Retrieve the names of all attributes in a data rule usage group.

TABLE

Table name associated with the data rule usage group.

VIEW

View name associated with the data rule usage group.

MATERIALIZED_VIEW

Materialized view name associated with the data rule usage group.

EXTERNAL_TABLE

External table name associated with the data rule usage group.

COLUMN

Column name associated with the data rule usage group attribute.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for TABLE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the table

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the table

Basic properties for COLUMN:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the column

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the column

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCHAR, VARCHAR2, XMLTYPE

Default: NUMBER

The datatype of a column

Name: LENGTH

Type: NUMBER

Valid Values:

Default: 1

The length of a number

Name: PRECISION

Type: NUMBER

Valid Values: 0 - 38

Default: 1

The precision of a number. Use 0 to specify floating-point numbers.

Name: SCALE

Type: NUMBER

Valid Values: -84 - 127

Default: 1

The scale of a number.

Name: FRACTIONAL_SECONDS_PRECISION

Type: NUMBER

Valid Values: 0 - 9

Default: 0

The precision of a timestamp or interval.

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Default value of the column

Name: NOT_NULL

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify "true" to enforce Not Null restriction on a column.

Basic properties for PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the table rows according to a Hash Algorithm, lists of values, or specified ranges.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH partitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of partitions that is a power of 2. If you have multiple Partition

Keys, you only have to specify once.

Basic properties for PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Commas can be escaped using "" (for example, '1,2,3'). Always specify DEFAULT as the value of the last partition, and make sure you have specified PARTITION_KEY(s) before you specify any PARTITION.. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH customized subpartitions the database should create for a particular main RANGE partition (RANGE-HASH BY QUANTITY partitioning). For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once. Set it to 0 to reverse to the use of generic template HASH_QUANTITY specified in SUBPARTITION_KEY.

Basic properties for SUBPARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: LIST, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

For partition-level partitioning according to a Hash Algorithm or lists of values. Each partition is further sorted into subpartitions.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH subpartitions the database should create on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of subpartitions that is a power of 2. If you have multiple Subpartition Keys, you only have to specify once.

Basic properties for SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST subpartition. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any SUBPARTITION. Each LIST subpartition must have at least one value. No value, including NULL, can appear in more than one subpartition.

Basic properties for TEMPLATE_SUBPARTITION:

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

In composite partitioning, template subpartitions are automatically applied to those partitions without their subpartitions specified. Here for LIST subpartitions only, specify a comma-delimited, ordered list of literal values corresponding to the LIST subpartitioning column. Always specify DEFAULT as the value of the last template LIST subpartition, and make sure you have specified SUBPARTITION_KEY(s) before you specify any TEMPLATE_SUBPARTITION. Each LIST template subpartition must have at least one value. No value, including NULL, can appear in more than one template subpartition.

Basic properties for INDEX:

Name: INDEX_TYPE

Type: STRING

Valid Values: UNIQUE, NON-UNIQUE, BITMAP, FUNCTION-BASED

Default: (No default, must be one of the preceding choices)

Specify the type of an index. NORMAL can be used in place of NON-UNIQUE.

Name: LOCAL_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

Specify if an index is Global or Local. The default is Global.

Specify Local so that the index is partitioned on the same columns, with the same number of partitions and the same partition bounds as table.

Oracle Database automatically maintains local index partitioning as the underlying table is repartitioned.

Name: COLUMN_EXPRESSION

Type: STRING

Valid Values: N/A

Default: "

Specify an expression built from columns of table, constants, SQL functions, and user-defined functions to create a FUNCTION-BASED index.

Basic properties for INDEX_PARTITION_KEY:

Name: TYPE

Type: STRING

Valid Values: RANGE, HASH, HASH BY QUANTITY

Default: (No default, must be one of the preceding choices)

Ask Oracle to partition the index rows according to a Hash Algorithm, lists of values, or specified ranges. Hash index partitioning is supported starting with Oracle Database 10g version.

Name: HASH_QUANTITY

Type: STRING

Valid Values: N/A

Default: '0'

Specify how many HASH index partitions the database should create based on HASH BY QUANTITY partitioning. For optimal load balancing you should specify a number of index partitions that is a power of 2. If you have multiple index Partition Keys, you only have to specify once.

Basic properties for INDEX_PARTITION:

Name: VALUES_LESS_THAN

Type: STRING

Valid Values: N/A

Default: "

Specify the noninclusive upper bound for the current RANGE partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column list. Always specify MAXVALUE(s) as the value(s) of the last partition. No need to specify VALUES_LESS_THAN for Local index.

Name: VALUES_EQUAL_TO

Type: STRING

Valid Values: N/A

Default: "

Specify a list of literal values for the current LIST partition in a global index. The value list is a comma-delimited, ordered list of literal values corresponding to the index partitioning column. Always specify DEFAULT as the value of the last partition. Each LIST partition must have at least one value. No value, including NULL, can appear in more than one partition. No need to specify VALUES_EQUAL_TO for Local index.

Basic properties for CHECK_CONSTRAINTS:

Name: CHECK_CONDITION

Type: STRING

Valid Values: N/A

Default: "

Specify a condition that each row in the table must satisfy.

Properties for TABLE:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: CACHE_MODE

Type: STRING

Valid Values: , CACHE, NOCACHE

Default: "

Indicate how Oracle should store blocks in the buffer cache.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size

in kilobytes or megabytes.

Name: INITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: MONITORING_MODE

Type: STRING

Valid Values: , MONITORING, NOMONITORING

Default: "

Specify MONITORING if you want modification statistics to be collected on this table.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow tablespaces. The number of tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of tablespaces, then Oracle cycles through the names of the tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. For composite-partitioned tables, it is used for subpartition template to store a list of tablespaces.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: ROWDEPENDENCIES_MODE

Type: STRING

Valid Values: , NOROWDEPENDENCIES, ROWDEPENDENCIES

Default: "

Specify ROWDEPENDENCIES to use row-level dependency tracking.

Name: ROW_MOVEMENT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify whether Oracle can move a table row.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PRIMARY_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use

the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for CHECK_CONSTRAINT:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Properties for INDEX:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: COMPUTESTATISTICS

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to collect statistics at relatively little cost during the creation of an index.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default is 1.

Name: INDEXORDER

Type: STRING

Valid Values: , ASC, DESC

Default: "

Use ASC or DESC to indicate whether the index should be created in ascending or descending order. The Oracle default is ASC.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (2-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 2 for Index.

Name: KEYCOMPRESS

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Specify COMPRESS to enable key compression.

Name: KEYCOMPRESSPREFIXLENGTH

Type: STRING

Valid Values: N/A

Default: "

Specify the prefix length (number of prefix columns to compress). For unique indexes, the valid range of prefix length values is from 1 to the number of key columns minus 1. The default prefix length is the number of key columns minus 1. For nonunique indexes, the valid range of prefix length values is from 1 to the number of key columns. The default prefix length is the number of key columns. Oracle compresses only nonpartitioned indexes that are nonunique or unique indexes of at least two columns. You cannot specify COMPRESS for a bitmap index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (2-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: ONLINE

Type: STRING

Valid Values: , NO, YES

Default: "

Specify YES to indicate that DML operations on the table will be allowed during creation of the index.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: OVERFLOW

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of Index tablespaces for overflow data. For simple-partitioned object, it is used for HASH BY QUANTITY partition overflow Index tablespaces. The number of Index tablespaces does not have to equal the number of partitions. If the number of partitions is greater than the number of Index tablespaces, then Oracle cycles through the names of the Index tablespaces.

Name: PARALLEL_ACCESS_MODE

Type: STRING

Valid Values: , NOPARALLEL, PARALLEL

Default: "

Enables or disables parallel processing when the table is created. Also enables or disables parallel processing or access. The default is PARALLEL.

Name: PARALLEL_DEGREE

Type: STRING

Valid Values: N/A

Default: "

Enter degree of parallelism, which is the number of parallel threads used in the parallel operation.

Name: PARTITION_TABLESPACE_LIST

Type: STRING

Valid Values: N/A

Default: "

Enter a comma separated list of tablespaces for a locally partitioned index. For simple-partitioned object, it is used for HASH BY QUANTITY partition tablespaces. If specified, then individual local Hash index partitions specified will be ignored for Local Hash or Range-Hash Index partitioning.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: SORT

Type: STRING

Valid Values: , NOSORT, REVERSE, SORT

Default: "

Specify NOSORT to indicate to Oracle that the rows are already stored in the database in ascending order. Specify REVERSE to store the bytes of the index block in reverse order, excluding the rowid.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Properties for PARTITION:

Name: BUFFER_POOL

Type: STRING

Valid Values: , DEFAULT, KEEP, RECYCLE

Default: "

Specify a default buffer pool (cache) for table or partition object. The default is DEFAULT.

Name: DATA_SEGMENT_COMPRESSION

Type: STRING

Valid Values: , COMPRESS, NOCOMPRESS

Default: "

Use this clause to instruct Oracle whether to compress data segments to reduce disk use. The default is NOCOMPRESS.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: FREELISTGROUPS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of groups of free lists for the database object you are creating. The default is 1.

Name: FREELISTS

Type: STRING

Valid Values: N/A

Default: "

Specify the number of free lists for each of the free list groups for the table, partition, cluster, or index.

Name: INITIAL

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the first extent. Use K or M to specify size in kilobytes or megabytes.

Name: INITRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the initial number (1-255) of concurrent transaction entries allocated within each data block allocated to the database object. The default is 1 for Table and 2 for Index.

Name: LOGGING_MODE

Type: STRING

Valid Values: , LOGGING, NOLOGGING

Default: "

Specify whether the creation of the table and of any indexes required because of constraints, partition, or LOB storage characteristics will be logged in the redo log file (LOGGING) or not (NOLOGGING). The default is LOGGING.

Name: MAXEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents, including the first, that Oracle can allocate for the object.

Name: MAXTRANS

Type: STRING

Valid Values: N/A

Default: "

Specify the maximum number (1-255) of concurrent transactions that can update a data block allocated to the database object.

Name: MINEXTENTS

Type: STRING

Valid Values: N/A

Default: "

Specify the total number of extents to allocate when the object is created.

Name: NEXT

Type: STRING

Valid Values: N/A

Default: "

Specify in bytes the size of the next extent to be allocated. Use K or M to specify size in kilobytes or megabytes.

Name: OPTIMAL

Type: STRING

Valid Values: N/A

Default: "

Specifies an optimal size in bytes for a rollback segment. Use K or M to specify this size in kilobytes or megabytes. Specify NULL for no optimal size for the rollback segment. The default is NULL.

Name: PCTFREE

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the percentage (0-99) of space in each data block of the database object reserved for future updates to the rows of the object. The default is 10.

Name: PCTINCREASE

Type: STRING

Valid Values: N/A

Default: "

Specify the percent by which the third and subsequent extents grow over the preceding extent. The default is 50.

Name: PCTUSED

Type: STRING

Valid Values: N/A

Default: "

Specify a whole number representing the minimum percentage (0-99) of used space that Oracle maintains for each data block of the database object. The default is 40.

Name: TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getTableSCOClauses

This clause retrieves components like columns, indexes and so on of a table.

getConfigPropertiesClause

This clause gets the configuration properties of the object.

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE TABLE 'NEW_TABLE' GET PROPERTIES (DESCRIPTION,
BUSINESS_NAME)
```

```
GET COLUMNS
```

This will retrieve its description and business name, and get columns.

See Also

OMBRETRIEVE, OMBCREATE TABLE, OMBALTER TABLE, OMBDROP TABLE

OMBRETRIEVE TABLE_FUNCTION

Purpose

Retrieve details of the Table Function.

Prerequisites

Should be in the context of Oracle Module or Package. The REFCursorType and PLSQLTableType which are set as Datatype for parameters should preexist in corresponding Package.

Syntax

```
retrieveTableFunctionCommand = OMBRETRIEVE TABLE_FUNCTION "QUOTED_STRING"  
    ( "retrieveTableFunctionClause" | "retrieveParameterClause" )  
retrieveTableFunctionClause = GET ( "propertiesClauseNoGet" |  
    "getParametersClause" | "getOrderedFieldsClause" |  
    "getPartitionedFieldsClause" | ( REF | REFERENCE ) ICONSET )  
retrieveParameterClause = PARAMETER "QUOTED_STRING" "getPropertiesClause"  
propertiesClauseNoGet = PROPERTIES "(" "propertyNameList" ")"  
getParametersClause = PARAMETERS  
getOrderedFieldsClause = ORDERED_FIELDS  
getPartitionedFieldsClause = PARTITIONED_FIELDS  
getPropertiesClause = GET PROPERTIES "(" "propertyNameList" ")"  
propertyNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveTableFunctionCommand

Retrieve details regarding a Table Function.

retrieveTableFunctionClause

Retrieve details regarding a Table Function.

retrieveParameterClause

Gets the properties of Parameter with the given name.

propertiesClauseNoGet

Gets the properties of the table function.

getParametersClause

Lists the Parameter names of this Table Function.

getOrderedFieldsClause

Lists the Field names of this Table Function on which the Ordering is to be

done.

getPartitionedFieldsClause

Lists the Field names of this Table Function on which the Partitioning is to be done.

getPropertiesClause

Gets the properties of the Table Function or any of its Parameter

Basic properties for FUNCTION:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Function

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Function

Name: RETURN_TYPE

Type: STRING

Valid Values: PLS_INTEGER, BINARY_INTEGER, BOOLEAN, NUMBER, FLOAT, CHAR,

VARCHAR, VARCHAR2, DATE

Default: NUMBER

Set the Return Type for Function

Name: IMPLEMENTATION

Type: STRING

Valid Values: N/A

Default: "

Set the code for Function which is included global variable declaration and code between BEGIN and END.

Name: IS_DETERMINISTIC

Type: BOOLEAN

Valid Values: true, false

Default: false

This setting helps the optimizer avoid redundant function calls.

Name: IS_PARALLEL_ENABLE

Type: BOOLEAN

Valid Values: true, false

Default: false

This option sets flag to a stored function can be used safely in the slave sessions of parallel DML evaluations.

Basic properties for PARAMETER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Parameter

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Parameter

Name: DATATYPE

Type: STRING

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BINARY_INTEGER, BLOB, BOOLEAN,

CHAR, CLOB, DATE, FLOAT, INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO

MONTH, NCHAR, NCLOB, NUMBER, NVARCHAR2, PLS_INTEGER, RAW, TIMESTAMP,

TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE, VARCHAR,

VARCHAR2, XMLTYPE

Default: NUMBER

Set the data type for Parameter

Name: IN_OUT

Type: STRING

Valid Values: IN, OUT, INOUT

Default: 'IN'

Set the parameter mode for Parameter

Name: DEFAULT_VALUE

Type: STRING

Valid Values: N/A

Default: "

Set the default value for Parameter

Properties for TABLE_FUNCTION:

Name: AUTHID

Type: STRING

Valid Values: Current_User, Definer, None

Default: None

Generate the transformation with selected AUTHID option. Function will be executed with the permissions defined by the AUTHID clause rather than the function owner's permissions.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

propertyNameList

Comma separated list of property names to retrieve values. Property names are unquoted.

Examples

```
OMBRETRIEVE TABLE_FUNCTION 'table_function' GET PARAMETERS
```

This will retrieve the Table Function "table_function" and list the names of its parameters.

See Also

OMBRETRIEVE

OMBRETRIEVE TIME_DIMENSION

Purpose

This command retrieve metadata from the time dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
retrieveTimeDimensionCommand = OMBRETRIEVE TIME_DIMENSION
    "TimeDimensionName" ( GET ( PROPERTIES "propertyKeyList" | FISCAL
        PROPERTIES "propertyKeyList" | DIMENSION_ATTRIBUTES | LEVELS |
        HIERARCHIES | DIMENSION_ROLES | YEARS | START_YEAR |
        IMPLEMENTATION_STRATEGY | MAPPING | IMPLEMENTED_OBJECTS ) |
        "dimensionAttributeDetailClause" | "levelDetailClause" |
        "hierarchyDetailClause" | "roleDetailClause" )
TimeDimensionName = "QUOTED_STRING"
propertyKeyList = "(" "propertyKey" { "," "propertyKey" } ")"
dimensionAttributeDetailClause = "dimensionAttributeLocator" GET
    PROPERTIES "propertyKeyList"
levelDetailClause = "levelLocator" ( GET ( PROPERTIES "propertyKeyList" |
    LEVEL_ATTRIBUTES | IMPLEMENTED_OBJECT ) | "levelAttributeDetailClause"
    )
hierarchyDetailClause = "hierarchyLocator" GET ( PROPERTIES
    "propertyKeyList" | ( REF | REFERENCE ) LEVELS )
roleDetailClause = "roleLocator" ( GET ( PROPERTIES "propertyKeyList" ) )
propertyKey = "UNQUOTED_STRING"
dimensionAttributeLocator = DIMENSION_ATTRIBUTE "dimensionAttributeName"
levelLocator = LEVEL "levelName"
levelAttributeDetailClause = "levelAttributeLocator" GET ( PROPERTIES
    "propertyKeyList" | IMPLEMENTED COLUMN )
hierarchyLocator = HIERARCHY "hierarchyName"
roleLocator = ROLE "roleName"
dimensionAttributeName = "QUOTED_STRING"
levelName = "QUOTED_STRING"
levelAttributeLocator = LEVEL_ATTRIBUTE "levelAttributeName"
hierarchyName = "QUOTED_STRING"
roleName = "QUOTED_STRING"
levelAttributeName = "QUOTED_STRING"
```

Keywords And Parameters

TimeDimensionName

The name of the time dimension.

propertyKeyList

A list of time dimension properties.

dimensionAttributeDetailClause

This clause gets property details of dimension attribute.

levelDetailClause

This clause gets property details of a level.

hierarchyDetailClause

This clause gets property details of a hierarchy.

roleDetailClause

This clause gets property details of a role.

propertyKey

Basic properties for TIME DIMENSION, TIME DIMENSION MAP, DIMENSION_ATTRIBUTE, LEVEL, LEVEL_ATTRIBUTE and HIERARCHY:

Basic properties for TIME DIMENSION :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Time Dimension

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Time Dimension

Name: STORAGE

Type: STRING

Valid Values: 'RELATIONAL', 'AW'

Default: 'RELATIONAL'

The storage of a dimension can be AW or relational

Name: AW_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the analytical workspace name where the dimension is implemented

Name: AW_DIMENSION_NAME

Type: STRING(32)

Valid Values: N/A

Default: "

Set the Analytical Workspace dimension physical object name

Basic properties for TIME MAP :

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Time Dimension Map

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Time Dimension Map

Basic properties for DIMENSION_ATTRIBUTE:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Dimension_Attribute

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Properties for DIMENSION:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: DEPLOYMENT_OPTIONS

Type: STRING

Valid Values: DEPLOY_ALL, DEPLOY_DATA_OBJECTS_ONLY, DEPLOY_TO_CATALOG_ONLY

Default: DEPLOY_DATA_OBJECTS_ONLY

Warehouse Builder generates a set of scripts for Dimension, they are DDL Scripts for Relational Dimensional or Scripts for ROLAP or or Scripts for AW.

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: VIEW_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Name of the view that is generated to hide the control rows on the dimension implementation table of a star schema. If this field is left blank, the view name will default to '<Name of Dimension>_v'

Name: VISIBLE

Type: BOOLEAN

Valid Values: true, false

Default: true

The Dimension is visible to OLAP end user if value is set = true.

Note:

1. N/A means any valid character in supported character set.
2. " " represents an empty string

levelLocator

This clause gets the level.

levelAttributeDetailClause

This clause gets property details of a level attribute.

hierarchyLocator

This clause gets the hierarchy.

levelAttributeLocator

This clause gets the level Attribute.

hierarchyName

The name of a hierarchy.

roleName

A role name.

Examples

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET LEVELS

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET HIERARCHIES

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET DIMENSION_ATTRIBUTES

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET ROLES

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET YEARS

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET START_YEAR

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET IMPLEMENTATION_
STRATEGY

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET MAPPING

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET FISCAL PROPERTIES (FISCAL_
TYPE)

OMBRETRIEVE TIME_DIMENSION 'FYR2005' GET PROPERTIES (DESCRIPTION,
BUSINESS_NAME)

OMBRETRIEVE TIME_DIMENSION 'FYR2005' DIMENSION_ATTRIBUTE 'ID' GET
PROPERTIES (DESCRIPTION, BUSINESS_NAME)

OMBRETRIEVE TIME_DIMENSION 'FYR2005' LEVEL 'MYCALMTH' GET
PROPERTIES (

DESCRIPTION, BUSINESS_NAME)

OMBRETRIEVE TIME_DIMENSION 'FYR2005' LEVEL 'MYCALMTH' GET LEVEL_ATTRIBUTES

OMBRETRIEVE TIME_DIMENSION 'FYR2005' LEVEL 'MYCALMTH' LEVEL_ATTRIBUTE 'ID'

GET PROPERTIES (DESCRIPTION, BUSINESS_NAME)

OMBRETRIEVE TIME_DIMENSION 'FYR2005' ROLE 'SALES' GET PROPERTIES (DESCRIPTION, BUSINESS_NAME)

and so on

See Also

OMBCREATE TIME_DIMENSION, OMBALTER TIME_DIMENSION, OMBDROP TIME_DIMENSION

OMBRETRIEVE TRANSPORTABLE_MODULE

Purpose

To retrieve properties of a transportable module, its contents and their properties, and source and target locations.

Prerequisites

In the context of a project.

Syntax

```
retrieveTMCommand = OMBRETRIEVE TRANSPORTABLE_MODULE "QUOTED_STRING" GET (
    "getPropertiesClause" | "getTablespacePropertiesClause" |
    "getDatafilePropertiesClause" | "getSchemaPropertiesClause" |
    "getTablespacesClause" | "getDatafilesInTablespaceClause" |
    "getSchemasInTablespaceClause" | "getAllSchemasClause" |
    "getSourceLocationClause" | "getTargetLocationClause" |
    "getReferenceIconSetClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getTablespacePropertiesClause = TRANSPORTABLE_MODULE_TABLESPACE
    "QUOTED_STRING" PROPERTIES "(" "propertyNameList" ")"
getDatafilePropertiesClause = DATAFILE "QUOTED_STRING" PROPERTIES "("
    "propertyNameList" ")"
getSchemaPropertiesClause = TRANSPORTABLE_MODULE_SCHEMA "QUOTED_STRING"
    PROPERTIES "(" "propertyNameList" ")"
getTablespacesClause = TRANSPORTABLE_MODULE_TABLESPACES
getDatafilesInTablespaceClause = DATAFILES IN
    TRANSPORTABLE_MODULE_TABLESPACE "QUOTED_STRING"
getSchemasInTablespaceClause = TRANSPORTABLE_MODULE_SCHEMAS IN
    TRANSPORTABLE_MODULE_TABLESPACE "QUOTED_STRING"
getAllSchemasClause = ALL TRANSPORTABLE_MODULE_SCHEMAS
getSourceLocationClause = SOURCE_LOCATION
getTargetLocationClause = TARGET_LOCATION
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveTMCommand

This command is for querying information about the transportable module and its contents.

getPropertiesClause

Retrieve properties for the transportable module.

getTablespacePropertiesClause

Retrieve properties for a tablespace within the transportable module.

QUOTED_STRING

The name of the tablespace whose properties are to be retrieved.

getDatafilePropertiesClause

Retrieve properties for a datafile within the transportable module.

QUOTED_STRING

The name of the datafile whose properties are to be retrieved.

getSchemaPropertiesClause

Retrieve properties for a schema within the transportable module.

QUOTED_STRING

The name of the schema whose properties are to be retrieved.

getTablespacesClause

Retrieve names of all tablespaces in the transportable module.

getDatafilesInTablespaceClause

Retrieve names of all datafiles of a tablespace within the transportable module.

QUOTED_STRING

The name of the tablespace whose datafile names are to be retrieved.

getSchemasInTablespaceClause

Retrieve names of schemas that are parts of the named tablespace.

QUOTED_STRING

The name of the tablespace whose component schemas names are to be retrieved.

getAllSchemasClause

Retrieve names of all schemas within the transportable module.

getSourceLocationClause

Retrieve name of the source location.

getTargetLocationClause

Retrieve name of the target location.

getReferenceIconSetClause

Retrieve name of the icon set.

propertyNameList

The list of unquoted property names.

Examples

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101' GET SOURCE_LOCATION
```

This command will return the source location name.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101' GET TARGET_LOCATION
```

This command will return the target location name.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET PROPERTIES (WORK_DIRECTORY, TARGET_OS_TYPE, WHAT_TO_DEPLOY,  
TRANSPORTA_TABLESPACE)
```

This command will retrieve the values of the listed properties.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET TRANSPORTABLE_MODULE_TABLESPACE 'src_tablespace_1'  
PROPERTIES (TARGET_TABLESPACE_NAME, DROP_EXISTING_TABLESPACE)
```

This command will retrieve properties associated with tablespace
src_tablespace_1.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET DATAFILE 'D:\TTSFILES\TTS1B.DBF'  
PROPERTIES (DIRECTORY, FILENAME, REUSE)
```

This command will retrieve properties associated with datafile
D:\TTSFILES\TTS1B.DBF.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET TRANSPORTABLE_MODULE_SCHEMA 'src_schema_1'  
PROPERTIES (SCHEMA_NAME, PASSWORD, DEFAULT_TABLESPACE,  
SCHEMA_EXISTS_ACTION, SCHEMA_DOESNT_EXIST_ACTION,  
TABLE_EXISTS_ACTION, COPY_SOURCE_SCHEMA, PARALLEL)
```

This command will retrieve properties associated with schema src_schema_1.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET TRANSPORTABLE_MODULE_TABLESPACES
```

This command will retrieve names of all tablespaces in the transportable module.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET DATAFILES IN TRANSPORTABLE_MODULE_TABLESPACES 'src_tablespace_1'
```

This command will retrieve datafile names of tablespaces src_tablespace_1.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET TRANSPORTABLE_MODULE_SCHEMAS IN TRANSPORTABLE_MODULE_
TABLESPACE
```

```
'src_tablespace_1'
```

This command will retrieve schemas that are parts of the tablespace src_tablespace_1.

```
OMBRETRIEVE TRANSPORTABLE_MODULE 'TM101'
```

```
GET ALL TRANSPORTABLE_MODULE_SCHEMAS
```

This command will retrieve all schemas in the transportable module.

See Also

OMBRETRIEVE, OMBCREATE TRANSPORTABLE_MODULE, OMBALTER TRANSPORTABLE_MODULE

OMBRETRIEVE USER

Purpose

To retrieve properties of a Warehouse Builder user.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
retrieveUserCommand = OMBRETRIEVE USER "QUOTED_STRING" ( GET
    "getPropertiesClause" )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveUserCommand

This clause retrieves properties of a Warehouse Builder user.

getPropertiesClause

Retrieve specified properties.

Basic properties for USER:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the User

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the User

Name: ISTARGETSCHEMA

Type: BOOLEAN

Valid Values: true, false

Default: false

If true, the user will be set up as target schema for deployment; and also

the property TARGETSCHEMAPWD must be provided when you are setting the ISTARGETSCHEMA as true.

Name: TARGETSCHEMAPWD

Type: STRING(30)

Valid Values: N/A

Default: N/A

This properties will be provided only when you are seting ISTARGETSCHEMA as true, so that the necessary target schema objects can be installed into the potential target schema. And this property cannot be retrieved due to security consideration.

User preferences:

Name: LOCALE

Type: STRING

Valid Values: Albanian, Arabic, Bulgarian, Byelorussian, Catalan, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, French, German, Greek, Hebrew, Hungarian, Icelandic, Italian, Japanese, Korean, Lithuanian, Macedonian, Norwegian, Polish, Portuguese, Romanian, Russian, Serbian, Serbo_Croatian, Slovak, Slovenian, Spanish, Swedish, Thai, Turkish, Ukranian

Default: "

Name: SHOW_PROJECT

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_MODULE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_LOCATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_ACTION

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_TYPE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: DEFAULT_PROFILE_LOCATION

Type: STRING

Valid Values: N/A

Default: "

Name: ALLOW_UNDO_REDO

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PAUSE_AFTER_COMPILE

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_COMMIT

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_JOB_NAME

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: PROMPT_FOR_EXECUTION_PARAMS

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_MONITOR

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DEPLOYMENT_COMPLETION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DEPLOYMENT_DEPENDENCIES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_MONITOR_RESULTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_MONITOR_LOGFILE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: PERSONALITY

Type: STRING

Valid Values: N/A

Default: Default

Name: SHOW_GUIDED_ASSISTANCE

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: HIDE_WIZARD_WELCOME_PAGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_DELETE_CONFIRMATION

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: RECYCLE_DELETED_OBJECTS

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: EMPTY_RECYCLE_BIN

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: CLEAR_CLIPBOARD

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_PROJECT

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_MODULE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: SHOW_GENERATION_LOCATION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_GENERATION_ACTION

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: SHOW_GENERATION_TYPE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_FILE_PATH

Type: STRING(1000)

Valid Values: N/A

Default: "

Name: LOG_FILE_NAME

Type: STRING(1000)

Valid Values: N/A

Default: log

Name: LOG_FILE_MAX_SIZE

Type: STRING

Valid Values: 1-10000000

Default: 100

Name: LOG_ERROR_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_WARNING_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: LOG_INFORMATION_MESSAGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: NAMING_MODE

Type: STRING

Valid Values: PHYSICAL_NAMING_MODE, BUSINESS_NAMING_MODE

Default: PHYSICAL_NAMING_MODE

Name: PROPAGATE_NAME_CHANGES

Type: BOOLEAN

Valid Values: true, false

Default: false

Name: DESIGNREPOS_PWD_PERSIST

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: RUNTIMEREPOS_PWD_SHARE

Type: BOOLEAN

Valid Values: true, false

Default: true

Name: DEFAULT_SEC_POLICY

Type: STRING

Valid Values: MINIMUM_SECURITY, MAXIMUM_SECURITY

Default: MINIMUM_SECURITY

Examples

```
OMBRETRIEVE USER 'USER1' GET PROPERTIES(BUSINESS_NAME,  
DESCRIPTION,  
ISTARGETSCHEMA)
```

See Also

OMBUNREGISTER USER, OMBALTER USER, OMBREGISTER USER

OMBRETRIEVE VARYING_ARRAY

Purpose

Retrieve details of the Varying Array.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
retrieveVaryingArrayCommand = OMBRETRIEVE VARYING_ARRAY "QUOTED_STRING" (
    GET ( "getPropertiesClause" | "getReferenceIconSetClause" ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

retrieveVaryingArrayCommand

Retrieves the details of the Varying Array with the given name.

getPropertiesClause

This clause retrieves all the properties.

Basic properties for VARYING_ARRAY:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the Varying Array

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the Varying Array

Name: DATATYPE

Type: STRING(20)

Valid Values: BINARY_DOUBLE, BINARY_FLOAT, BLOB, CHAR, CLOB, DATE, FLOAT,

INTEGER, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH, NCHAR, NCLOB,

NUMBER, NVARCHAR2, RAW, SYS.ANYDATA, SYS.LCR\$_ROW_RECORD, SYS.XMLFORMAT,

TIMESTAMP, TIMESTAMP WITH LOCAL TIME ZONE, TIMESTAMP WITH TIME ZONE,

VARHCAR, VARCHAR2, XMLTYPE

Default: "

Datatype of the Base Element of the Varying Array

Properties for VARYING_ARRAY:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

propertyNameList

The list of properties.

Examples

```
OMBRETRIEVE VARYING_ARRAY 'SOME_VARRAY' GET PROPERTIES  
(DATATYPE)
```

This will retrieve the Varying Array "SOME_VARRAY"'s base element datatype.

See Also

OMBRETRIEVE

OMBRETRIEVE VIEW

Purpose

To retrieve properties of a view.

Prerequisites

In the context of an Oracle Module

Syntax

```
retrieveViewCommand = OMBRETRIEVE VIEW "QUOTED_STRING" (
    "retrieveViewClause" | "retrieveColumnClause" | "retrieveUkPkClause" |
    "retrieveFkClause" | "retrieveDataRuleUsageClause" )
retrieveViewClause = GET ( "getPropertiesClause" |
    "getReferenceIconSetClause" | "getViewSCOrDependentsClause" )
retrieveColumnClause = COLUMN "QUOTED_STRING" GET "getPropertiesClause"
retrieveUkPkClause = ( UNIQUE_KEY | PRIMARY_KEY ) "QUOTED_STRING" GET (
    "getPropertiesClause" | COLUMNS )
retrieveFkClause = FOREIGN_KEY "QUOTED_STRING" GET ( "getPropertiesClause"
    | COLUMNS | UNIQUE_KEY | PRIMARY_KEY | REFERENCED_KEY )
retrieveDataRuleUsageClause = DATA_RULE_USAGE "QUOTED_STRING" ( GET (
    "getPropertiesClause" | GROUPS ) | GROUP "QUOTED_STRING" ( GET (
    "getPropertiesClause" | ATTRIBUTES | REF ( TABLE | VIEW |
    MATERIALIZED_VIEW | EXTERNAL_TABLE ) ) | ATTRIBUTE "QUOTED_STRING" (
    GET ( "getPropertiesClause" | REF COLUMN ) ) ) )
getPropertiesClause = PROPERTIES "(" "propertyNameList" ")"
getReferenceIconSetClause = ( REF | REFERENCE ) ICONSET
getViewSCOrDependentsClause = COLUMNS | UNIQUE_KEYS | PRIMARY_KEY |
    FOREIGN_KEYS | COLUMN AT POSITION "INTEGER_LITERAL" | DATA_RULE_USAGES
    | ( REF | REFERENCE ) ( TABLES | VIEWS | MATERIALIZED_VIEWS )
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
```

Keywords And Parameters

`retrieveViewCommand`

This clause retrieves a view.

`QUOTED_STRING`

name of the view.

`retrieveViewClause`

This clause will retrieve a view.

`retrieveColumnClause`

This clause will retrieve columns.

`QUOTED_STRING`

Name of the column.

`retrieveUkPkClause`

This clause will retrieve a unique key or primary key.

`QUOTED_STRING`

Name of the unique key or the primary key.

`retrieveFkClause`

This clause will retrieve a key referenced by a foreign key, either a unique key or primary key. Use `REFERENCED_KEY` to retrieve the referenced key for a foreign key regardless of the type of referenced key (unique or primary).

`QUOTED_STRING`

Name of the foreign key.

`retrieveDataRuleUsageClause`

This clause retrieves the data rule usages.

`QUOTED_STRING`

Name of data rule usage, group or attribute.

`GROUPS`

Retrieve the names of all relation groups in the data rule usage.

`ATTRIBUTES`

Retrieve the names of all attributes in a data rule usage group.

`TABLE`

Table name associated with the data rule usage group.

`VIEW`

View name associated with the data rule usage group.

`MATERIALIZED_VIEW`

Materialized view name associated with the data rule usage group.

EXTERNAL_TABLE

External table name associated with the data rule usage group.

COLUMN

Column name associated with the data rule usage group attribute.

getPropertiesClause

This clause retrieves all the properties.

Note:

Constraints can be specified but will not be generated for either View or Materialized View in this release.

Basic properties for VIEW, MATERIALIZED_VIEW:

Name: BUSINESS_NAME

Type: STRING(200)

Valid Values: N/A

Default: "

Business name of the View, MaterializedView

Name: DESCRIPTION

Type: STRING(4000)

Valid Values: N/A

Default: "

Description of the View, MaterializedView

Name: VIEW_QUERY

Type: STRING(4000)

Valid Values: N/A

Default: "

Sets the query definition in View and MaterializedView.

Properties for VIEW:

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: GENERATION_COMMENTS

Type: STRING

Valid Values: N/A

Default: "

Enter additional comments for the generated code.

Name: SHADOW_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

Use the Tablespace parameter to specify the name of tablespace.

Name: SHADOW_TABLE_NAME

Type: STRING(30)

Valid Values: N/A

Default: "

Use the shadow table name to specify the name of Shadow Table.

Properties for UNIQUE_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that

in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INDEX_TABLESPACE

Type: STRING(30)

Valid Values: N/A

Default: "

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.INDEXTABLESPACE:DESCRIPTION"

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: USING_INDEX

Type: BOOLEAN

Valid Values: true, false

Default: false

oracle.owb.scripting.help.OMBHelpGenerator(557): NLS Lookup error for key "9i.TABLE.CONSTRAINT.USINGINDEX:DESCRIPTION"

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated). The default is NOVALIDATE.

Properties for FOREIGN_KEY:

Name: DEFERRABLE

Type: STRING

Valid Values: , DEFERRABLE, NOT DEFERRABLE

Default: "

Specify DEFERRABLE to indicate that in subsequent transactions you can use the SET CONSTRAINT[S] clause to defer checking of this constraint until after the transaction is committed. Specify NOT DEFERRABLE to indicate that in subsequent transactions you cannot use the SET CONSTRAINT[S] clause to defer checking of this constraint until the transaction is committed. The default is NOT DEFERRABLE.

Name: DEPLOYABLE

Type: BOOLEAN

Valid Values: true, false

Default: true

Warehouse Builder generates a set of scripts to create an object only for those object marked as Deployable = true

Name: ENABLECONSTRAINT

Type: STRING

Valid Values: , DISABLE, ENABLE

Default: "

Specify ENABLE if you want the constraint to be applied to the data in the table. Specify DISABLE to disable the integrity constraint. The default is ENABLE.

Name: EXCEPTIONSINTO

Type: STRING

Valid Values: N/A

Default: "

Specify an exceptions table ([schema.]table). The EXCEPTIONS table or the table you specify must exist on your local database. If you create your own exceptions table, then it must follow the format prescribed by one of the two scripts supplied by Oracle. Do not use this property with NOVALIDATE option.

Name: INITIALLY

Type: STRING

Valid Values: , DEFERRED, IMMEDIATE

Default: "

Specify (INITIALLY) IMMEDIATE to indicate that Oracle should check a DEFERRABLE constraint at the end of each subsequent SQL statement. Specify (INITIALLY) DEFERRED to indicate that Oracle should check a DEFERRABLE constraint at the end of subsequent transactions. The default is (INITIALLY) IMMEDIATE.

Name: ONDELETE

Type: STRING

Valid Values: , CASCADE, SET NULL

Default: "

Specify CASCADE if you want Oracle to remove dependent foreign key values. Specify SET NULL if you want Oracle to convert dependent foreign key values to NULL.

Name: RELY

Type: STRING

Valid Values: , NORELY, RELY

Default: "

Specify RELY to activate an existing constraint in NOVALIDATE mode for

query rewrite in an unenforced query rewrite integrity mode. The default is NORELY.

Name: SUBSTITUTE_KEY

Type: BOOLEAN

Valid Values: true, false

Default: false

This is related to Streams Support. If this is true, deployment will result only in creation of the key metadata. The constraint itself will not be enforced. This will be done by creating a Streams substitute key.

Name: VALIDATECONSTRAINT

Type: STRING

Valid Values: , NOVALIDATE, VALIDATE

Default: "

The behavior of VALIDATE and NOVALIDATE always depends on whether the constraint is enabled or disabled, either explicitly or by default.

(ENABLE) VALIDATE specifies that all old and new data must comply with the constraint. (ENABLE) NOVALIDATE only ensures that all new DML operations on the constrained data comply with the constraint. (DISABLE) VALIDATE disables the constraint and drops the index on the constraint, but keeps the constraint valid. (DISABLE) NOVALIDATE signifies that Oracle makes no effort to maintain the constraint (because it is disabled) and cannot guarantee that the constraint is true (because it is not being validated).

The default is NOVALIDATE.

Note:

1. N/A means any valid character in supported character set.
2. " represents an empty string

getViewSCOorDependentsClause

This clause will retrieves view components like columns, keys, and so on or relational objects that this view has referential dependency on.

propertyNameList

The list of properties.

Examples

OMBRETRIEVE VIEW 'NEW_VIEW' GET PROPERTIES (DESCRIPTION, BUSINESS_NAME)

This will retrieve its description and business name.

See Also

OMBRETRIEVE, OMBCREATE VIEW, OMBALTER VIEW, OMBDROP VIEW

This chapter lists commands associated with OMBDROP in alphabetical order.

OMBDROP

Purpose

Drop a component.

Prerequisites

Should be in the parent context of the component to drop.

Syntax

```
dropCommand = OMBDROP "fco_type" "fco_name"
```

Keywords And Parameters

dropCommand

Specify the component to drop.

fco_type

The type of the component.

fco_name

The physical name of the component in single quotes.

Examples

This is an example of dropping a table:

```
OMBDROP TABLE 'T1'
```

See Also

OMBCREATE, OMBALTER

OMBDROP ACTIVITY_TEMPLATE

Purpose

To drop an activity template folder.

Prerequisites

Should be in the context of a Activity Template folder.

Syntax

```
dropActivityTemplate = OMBDROP ACTIVITY_TEMPLATE "QUOTED_STRING"
```

Examples

```
OMBDROP ACTIVITY_TEMPLATE 'TEMPLATE1'
```

See Also

OMBDROP, OMBCREATE ACTIVITY_TEMPLATE, OMBALTER ACTIVITY_TEMPLATE

OMBDROP ACTIVITY_TEMPLATE_FOLDER

Purpose

To drop an activity template folder.

Prerequisites

Should be in the context of a Project.

Syntax

```
dropActivityTemplateFolder = OMBDROP ACTIVITY_TEMPLATE_FOLDER  
    "QUOTED_STRING"
```

Examples

```
OMBDROP ACTIVITY_TEMPLATE_FOLDER 'FOLDER1'
```

See Also

OMBDROP, OMBCREATE ACTIVITY_TEMPLATE_FOLDER, OMBALTER
ACTIVITY_TEMPLATE_FOLDER

OMBDROP ADVANCED_QUEUE

Purpose

Delete the Advanced Queue.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
dropAQCommand = OMBDROP ADVANCED_QUEUE "QUOTED_STRING"
```

Keywords And Parameters

dropAQCommand

Drops the Advanced Queue with the given name.

Examples

```
OMBDROP ADVANCED_QUEUE 'SOME_ADVANCED_QUEUE'
```

This will delete the "SOME_ADVANCED_QUEUE" Advanced Queue.

See Also

OMBDROP, OMBCREATE ADVANCED_QUEUE, OMBALTER ADVANCED_QUEUE, OMBRETRIEVE ADVANCED_QUEUE

OMBDROP ALTERNATIVE_SORT_ORDER

Purpose

To drop an Alternative Sort Order.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropAlternativeSortOrderCommand = OMBDROP ALTERNATIVE_SORT_ORDER  
    "QUOTED_STRING"
```

Keywords And Parameters

dropAlternativeSortOrderCommand

To drop an Alternative Sort Order.

Examples

```
OMBDROP ALTERNATIVE_SORT_ORDER 'customer_item'.
```

See Also

OMBRETRIEVE ALTERNATIVE_SORT_ORDER, OMBCREATE ALTERNATIVE_SORT_ORDER, OMBALTER ALTERNATIVE_SORT_ORDER

OMBDROP ANALYZE_ACTION_PLAN

Purpose

To drop a profile action plan.

Prerequisites

An action plan for profiling must already exist.

Syntax

```
DropActionPlanCommand = ( OMBDROP ( ( DEPLOYMENT_ACTION_PLAN |  
ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |  
CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" )
```

Keywords And Parameters

DropActionPlanCommand

Drop profile action plan

QUOTED_STRING

Name of profile action plan.

Examples

```
OMBDROP ANALYZE_ACTION_PLAN 'ANALYZE_PLAN'
```

alterAnalyzeActionPlanPreTag = An action plan for profiling must already exist.

See Also

OMBCREATE ANALYZE_ACTION_PLAN, OMBPROFILE

OMBDROP BUSINESS_AREA

Purpose

To drop a Business Area.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropBusinessAreaCommand = OMBDROP BUSINESS_AREA "QUOTED_STRING"
```

Keywords And Parameters

dropBusinessAreaCommand
To drop a Business Area.

Examples

```
OMBDROP BUSINESS_AREA 'sales'.
```

See Also

OMBRETRIEVE BUSINESS_AREA, OMBCREATE BUSINESS_AREA, OMBALTER BUSINESS_AREA

OMBDROP BUSINESS_DEFINITION_MODULE

Purpose

Delete the business definition module.

Prerequisites

Should be in the context of project.

Syntax

```
dropEULModuleCommand = OMBDROP ( BUSINESS_DEFINITION_MODULE  
    "QUOTED_STRING" )
```

Keywords And Parameters

dropEULModuleCommand

Remove an existing business definition module.

QUOTED_STRING

Name of the existing business definition module in quotes.

Examples

```
OMBDROP BUSINESS_DEFINITION_MODULE 'src_module'
```

This will delete the "src_module" business definition module.

See Also

OMBDROP, OMBCREATE BUSINESS_DEFINITION_MODULE, OMBALTER
BUSINESS_DEFINITION_MODULE

OMBDROP BUSINESS_PRESENTATION_MODULE

Purpose

Delete the presentation module.

Prerequisites

Should be in the context of project.

Syntax

```
dropReportModuleCommand = OMBDROP ( BUSINESS_PRESENTATION_MODULE  
    "QUOTED_STRING" )
```

Keywords And Parameters

dropReportModuleCommand

Remove an existing presentation module.

QUOTED_STRING

Name of the existing presentation module in quotes.

Examples

```
OMBDROP BUSINESS_PRESENTATION_MODULE 'salesrep_module'
```

This will delete the "salesrep_module" presentation module.

See Also

OMBDROP, OMBCREATE BUSINESS_PRESENTATION_MODULE, OMBALTER
BUSINESS_PRESENTATION_MODULE

OMBDROP CALENDAR

Purpose

To drop a calendar.

Prerequisites

Should be in the context of a Calendar Folder.

Syntax

```
parseDropCalendarCommand = OMBDROP CALENDAR "QUOTED_STRING"
```

Examples

```
OMBDROP CALENDAR 'CAL1'
```

See Also

OMBDROP, OMBCREATE CALENDAR, OMBALTER CALENDAR

OMBDROP CALENDAR_MODULE

Purpose

To drop a calendar module.

Prerequisites

Should be in the context of a Project.

Syntax

```
parseDropCalendarModuleCommand = OMBDROP CALENDAR_MODULE "QUOTED_STRING"
```

Examples

```
OMBDROP CALENDAR_MODULE 'CAL_MOD'
```

See Also

OMBDROP, OMBCREATE CALENDAR_MODULE, OMBALTER CALENDAR_MODULE

OMBDROP CHANGE_DATA_CAPTURE

Purpose

This command is used to drop a change data capture.

Prerequisites

This command can only be executed in the context of a change data capture and operates only on already existing change data capture.

Syntax

```
dropChangeSetCommand = OMBDROP ( CHANGE_DATA_CAPTURE "QUOTED_STRING" )
```

Examples

```
OMBDROP CHANGE_DATA_CAPTURE 'EMPLOYEE_CHANGES'
```

This deletes the change data capture.

See Also

OMBRETRIEVE CHANGE_DATA_CAPTURE, OMBCREATE CHANGE_DATA_CAPTURE, OMBALTER CHANGE_DATA_CAPTURE

OMBDROP CMI_DEFINITION

Purpose

Delete the CMI definition.

Prerequisites

Should be in the root context.

Syntax

```
dropMIVDefinitionCommand = OMBDROP ( CMI_DEFINITION "QUOTED_STRING" )
```

Keywords And Parameters

dropMIVDefinitionCommand

Remove an existing CMI definition.

QUOTED_STRING

Name of the existing CMI definition in quotes.

Examples

```
OMBDROP CMI_DEFINITION 'src_definition'
```

This will delete the "src_definition" CMI definition.

See Also

OMBDROP, OMBCREATE CMI_DEFINITION

OMBDROP CMI_MODULE

Purpose

Delete the CMI module.

Prerequisites

Should be in the context of project.

Syntax

```
dropMIVModuleCommand = OMBDROP ( CMI_MODULE "QUOTED_STRING" )
```

Keywords And Parameters

dropMIVModuleCommand

Remove an existing CMI module.

QUOTED_STRING

Name of the existing CMI module in quotes.

Examples

```
OMBDROP CMI_MODULE 'src_module'
```

This will delete the "src_module" CMI module.

See Also

OMBDROP, OMBCREATE CMI_MODULE, OMBALTER CMI_MODULE

OMBDROP COLLECTION

Purpose

Drop the collection object from this project.

Prerequisites

Should be in the context of a project, before dropping a collection.

Syntax

```
dropCollectionCommand = OMBDROP ( COLLECTION "QUOTED_STRING" )
```

Keywords And Parameters

dropCollectionCommand

Drop a collection of objects.

Examples

```
OMBDROP COLLECTION 'PURCHASING_WAREHOUSE'
```

See Also

OMBDROP, OMBALTER COLLECTION, OMBCREATE COLLECTION

OMBDROP CONFIGURATION

Purpose

Delete the Configuration.

Prerequisites

Should be in the context of a project.

Syntax

```
dropConfigurationCommand = OMBDROP ( CONFIGURATION "QUOTED_STRING" )
```

Keywords And Parameters

dropConfigurationCommand

Remove an existing Configuration.

Examples

```
OMBDROP CONFIGURATION 'QA_CONFIGURATION'
```

This will delete the "QA_CONFIGURATION" Configuration.

See Also

OMBDROP, OMBALTER CONFIGURATION, OMBCREATE CONFIGURATION

OMBDROP CONNECTOR

Purpose

Delete a connector.

Prerequisites

Can be in any context; the name is a name of the connector's owning location and a connector name separated by slash.

Syntax

```
dropConnectorCommand = OMBDROP ( CONNECTOR "QUOTED_STRING" )
```

Keywords And Parameters

dropConnectorCommand

Drops the named connector from the repository.

Examples

```
OMBDROP CONNECTOR 'A_LOCATION/A_CONNECTOR'
```

This will delete the "A_CONNECTOR" connector of "A_LOCATION".

See Also

OMBDROP, OMBCREATE CONNECTOR, OMBALTER CONNECTOR

OMBDROP CONTROL_CENTER

Purpose

Delete the control center.

Prerequisites

Can be in any context.

Syntax

```
dropControlCenterCommand = OMBDROP ( CONTROL_CENTER "QUOTED_STRING" )
```

Keywords And Parameters

dropControlCenterCommand

Drop the control center from the repository.

Examples

```
OMBDROP CONTROL_CENTER 'MY_CONNECTION'
```

This will delete the "MY_CONNECTION" control center.

See Also

OMBDROP, OMBCREATE CONTROL_CENTER, OMBALTER CONTROL_CENTER

OMBDROP CORRECTION_MAPS_ACTION_PLAN

Purpose

Drop an action plan for creating a correction map.

Prerequisites

In the context of a data profile.

Syntax

```
DropActionPlanCommand = ( OMBDROP ( ( DEPLOYMENT_ACTION_PLAN |
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" )
```

Keywords And Parameters

DropActionPlanCommand

Drop a correction map action plan.

QUOTED_STRING

Name of correction map action plan.

Examples

```
OMBDROP CORRECTION_MAPS_ACTION_PLAN 'CORRECT_INV_LOC_MAP'
```

See Also

OMBCREATE CORRECTION_MAPS_ACTION_PLAN, OMBPROFILE

OMBDROP CORRECTION_SCHEMA_ACTION_PLAN

Purpose

Drop an action plan for creating a correction schema

Prerequisites

In the context of a data profile.

Syntax

```
DropActionPlanCommand = ( OMBDROP ( ( DEPLOYMENT_ACTION_PLAN |  
ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |  
CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" )
```

Keywords And Parameters

DropActionPlanCommand

drop correction schema action plan

QUOTED_STRING

Name of correction schema action plan.

Examples

```
OMBDROP CORRECTION_SCHEMA_ACTION_PLAN 'CORRECT_INV_LOC'
```

See Also

OMBCREATE CORRECTION_SCHEMA_ACTION_PLAN, OMBPROFILE

OMBDROP CUBE

Purpose

This command drops a Cube.

Prerequisites

Should be in Oracle Module context.

Syntax

```
dropCubeCommand = OMBDROP CUBE "cubeName"  
cubeName = "QUOTED_STRING"
```

Examples

```
OMBDROP CUBE 'CUBE1'
```

See Also

OWBDROP, OMBCREATE CUBE, OMBALTER CUBE, OMBRETRIEVE CUBE

OMBDROP DATA_AUDITOR

Purpose

Drop an existing data auditor.

Prerequisites

The current context of scripting must be an Oracle

Syntax

```
dropDataAuditorCommand = OMBDROP DATA_AUDITOR "dataAuditorName"  
dataAuditorName = "QUOTED_STRING"
```

Keywords And Parameters

dropDataAuditorCommand

Drop an existing data auditor.

dataAuditorName

Name of data auditor.

Examples

```
OMBDROP DATA_AUDITOR 'MAP1'
```

See Also

OWBDROP, OMBCREATE DATA_AUDITOR, OMBALTER DATA_AUDITOR,
OMBRETRIEVE DATA_AUDITOR

OMBDROP DATA_PROFILE

Purpose

Delete the Data Profile.

Prerequisites

Should be in the context of project.

Syntax

```
dropDataProfileCommand = OMBDROP ( DATA_PROFILE "QUOTED_STRING" )
```

Keywords And Parameters

`dropDataProfileCommand`

Remove an existing Data Profile.

`QUOTED_STRING`

Name of the existing Data Profile in quotes.

Examples

```
OMBDROP DATA_PROFILE 'src_profile'
```

This will delete the "src_profile" Data Profile.

See Also

OMBDROP, OMBCREATE DATA_PROFILE, OMBALTER DATA_PROFILE

OMBDROP DATA_RULE

Purpose

Drop an existing data rule.

Prerequisites

The current context of scripting must be a data rule module.

Syntax

```
dropDataRuleCommand = OMBDROP DATA_RULE "QUOTED_STRING"
```

Keywords And Parameters

dropDataRuleCommand

This clause drops a data rule.

QUOTED_STRING

Data rule name.

Examples

```
OMBDROP DATA_RULE 'RULE1'
```

See Also

OMBRETRIEVE DATA_RULE, OMBCREATE DATA_RULE, OMBALTER DATA_RULE

OMBDROP DEPLOYMENT

Purpose

Delete the Deployment.

Prerequisites

Should be in the context of a Configuration.

Syntax

```
dropDeploymentCommand = OMBDROP ( DEPLOYMENT "QUOTED_STRING" )
```

Keywords And Parameters

dropDeploymentCommand

Remove an existing Deployment.

Examples

```
OMBDROP DEPLOYMENT 'QA_DEPLOYMENT'
```

This will delete the "QA_DEPLOYMENT" Deployment.

See Also

OMBDROP, OMBALTER DEPLOYMENT, OMBCREATE DEPLOYMENT

OMBDROP DEPLOYMENT_ACTION_PLAN

Purpose

Remove an existing deployment action plan.

Prerequisites

There must be a current working project.

Syntax

```
DropActionPlanCommand = ( OMBDROP ( ( DEPLOYMENT_ACTION_PLAN |  
    ANALYZE_ACTION_PLAN | CORRECTION_SCHEMA_ACTION_PLAN |  
    CORRECTION_MAPS_ACTION_PLAN ) ) "QUOTED_STRING" )
```

Keywords And Parameters

DropActionPlanCommand

Remove an existing deployment action plan.

Examples

```
OMBDROP DEPLOYMENT_ACTION_PLAN 'MY_PLAN'
```

See Also

OMBCREATE DEPLOYMENT_ACTION_PLAN, OMBDEPLOY

OMBDROP DIMENSION

Purpose

This command drops a dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
dropDimensionCommand = OMBDROP DIMENSION "dimensionName"  
dimensionName = "QUOTED_STRING"
```

Examples

```
OMBDROP DIMENSION 'PRODUCTS'
```

See Also

OWBDROP, OMBCREATE DIMENSION, OMBALTER DIMENSION, OMBRETRIEVE DIMENSION

OMBDROP DRILL_PATH

Purpose

To drop a Drill Path.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropDrillPathCommand = OMBDROP DRILL_PATH "QUOTED_STRING"
```

Keywords And Parameters

dropDrillPathCommand

To drop a Drill Path.

Examples

```
OMBDROP DRILL_PATH 'product_rollup'.
```

See Also

OMBRETRIEVE DRILL_PATH, OMBCREATE DRILL_PATH, OMBALTER DRILL_PATH

OMBDROP DRILL_TO_DETAIL

Purpose

To drop a Drill to Detail.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropDrillToDetailCommand = OMBDROP DRILL_TO_DETAIL "QUOTED_STRING"
```

Keywords And Parameters

dropDrillToDetailCommand

To drop a Drill to Detail.

Examples

```
OMBDROP DRILL_TO_DETAIL 'customer_item'.
```

See Also

OMBRETRIEVE DRILL_TO_DETAIL, OMBCREATE DRILL_TO_DETAIL,
OMBALTER DRILL_TO_DETAIL

OMBDROP EXPERT

Purpose

To drop an expert.

Prerequisites

In the context of an expert module.

Syntax

```
dropExpertCommand = OMBDROP EXPERT "QUOTED_STRING"
```

Keywords And Parameters

dropExpertCommand

Drop the specified expert.

Examples

This command will drop the expert EXP1:

```
OMBDROP EXPERT 'EXP1'
```

See Also

OMBDROP, OMBCREATE EXPERT, OMBALTER EXPERT, OMBRETRIEVE EXPERT

OMBDROP EXPERT_MODULE

Purpose

To drop an expert module.

Prerequisites

In the context of a project.

Syntax

```
dropExpertModuleCommand = OMBDROP ( EXPERT_MODULE "QUOTED_STRING" )
```

Keywords And Parameters

dropExpertModuleCommand

Delete an expert module.

Examples

This command will drop the expert module EM1:

```
OMBDROP EXPERT_MODULE 'EM1'
```

See Also

OMBDROP, OMBCREATE EXPERT_MODULE, OMBALTER EXPERT_MODULE

OMBDROP EXTERNAL_TABLE

Purpose

Delete the external table.

Prerequisites

Should be in the context of an Oracle module.

Syntax

```
dropExternalTableCommand = OMBDROP EXTERNAL_TABLE "QUOTED_STRING"
```

Keywords And Parameters

dropExternalTableCommand

Drop an external table from the repository.

QUOTED_STRING

The name of the external table to drop.

Examples

```
OMBDROP EXTERNAL_TABLE 'SRC_TABLE'
```

This will delete the external table "SRC_TABLE".

See Also

OMBDROP, OMBCREATE EXTERNAL_TABLE, OMBALTER EXTERNAL_TABLE,
OMBRETRIEVE EXTERNAL_TABLE

OMBDROP FLAT_FILE

Purpose

Delete a flat file.

Prerequisites

Should be in the context of a flat file module.

Syntax

```
dropFlatFileCommand = OMBDROP FLAT_FILE "QUOTED_STRING"
```

Keywords And Parameters

`dropFlatFileCommand`

Drop a flat file.

`QUOTED_STRING`

The name of the flat file to drop.

Examples

```
OMBDROP FLAT_FILE 'SRC_FILE'
```

This will delete the flat file "SRC_FILE".

See Also

OMBDROP, OMBCREATE FLAT_FILE, OMBALTER FLAT_FILE

OMBDROP FLAT_FILE_MODULE

Purpose

Delete a flat file module.

Prerequisites

Should be in the context of a project.

Syntax

```
dropFlatFileModuleCommand = OMBDROP ( FLAT_FILE_MODULE "QUOTED_STRING" )
```

Keywords And Parameters

`dropFlatFileModuleCommand`

Drop a flat file module.

`QUOTED_STRING`

The name of the flat file module to drop.

Examples

```
OMBDROP FLAT_FILE_MODULE 'src_module'
```

This will delete the "src_module" flat file module.

See Also

OMBDROP, OMBCREATE FLAT_FILE_MODULE, OMBALTER FLAT_FILE_MODULE

OMBDROP FUNCTION

Purpose

Delete the Function.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
dropFunctionCommand = OMBDROP ( FUNCTION "QUOTED_STRING" )
```

Keywords And Parameters

dropFunctionCommand

Remove an existing Function.

QUOTED_STRING

Name of the existing Function in quotes.

Examples

```
OMBDROP FUNCTION 'func'
```

This will delete the "func" Function.

If Packaged Function is overloaded, first find the Signature by using

OMBLIST command, and then use OMBALTER command using appropriate signature.

Example, if OMBLIST FUNCTIONS gives following two signatures,

```
FUNC_1 (NUMBER) RETURN NUMBER
```

```
FUNC_1 (VARCHAR2, NUMBER) RETURN NUMBER
```

The OMBDROP Syntax to drop the first one will be as follows

```
OMBDROP FUNCTION 'FUNC_1 \ (NUMBER\ ) RETURN NUMBER'
```

See Also

OMBDROP, OMBCREATE FUNCTION, OMBALTER FUNCTION

OMBDROP GATEWAY_MODULE

Purpose

Delete the Gateway module.

Prerequisites

Should be in the context of project.

Syntax

```
dropGatewayModuleCommand = OMBDROP ( GATEWAY_MODULE "QUOTED_STRING" )
```

Keywords And Parameters

dropGatewayModuleCommand

Remove an existing Gateway module.

QUOTED_STRING

Name of the existing Gateway module in quotes.

Examples

```
OMBDROP GATEWAY_MODULE 'db2_module'
```

This will delete the "db2_module" Gateway module.

See Also

OMBDROP, OMBCREATE GATEWAY_MODULE, OMBALTER GATEWAY_MODULE

OMBDROP ICONSET

Purpose

To remove an iconset from the repository.

Prerequisites

Any context.

Syntax

```
dropIconSetCommand = OMBDROP ICONSET "QUOTED_STRING"
```

Keywords And Parameters

`dropIconSetCommand`

This command removes an iconset from the repository.

`QUOTED_STRING`

The name of the iconset to be removed.

Examples

```
OMBDROP ICONSET 'ICON1'
```

See Also

OMBCREATE ICONSET, OMBALTER ICONSET

OMBDROP IMPORT_ACTION_PLAN

Purpose

To drop a transient import action plan.

Prerequisites

In the context of a project

Syntax

```
dropImportActionPlanCommand = ( OMBDROP ( IMPORT_ACTION_PLAN )  
    "QUOTED_STRING" )
```

Keywords And Parameters

dropImportActionPlanCommand

This command is for dropping an import action plan.

QUOTED_STRING

The name of the import action plan to be dropped.

Examples

```
OMBDROP IMPORT_ACTION_PLAN 'PLAN1'
```

This command will erase the transient import action plan PLAN1 from memory.

See Also

OMBCREATE IMPORT_ACTION_PLAN, OMBIMPORT

OMBDROP ITEM_FOLDER

Purpose

To drop an Item Folder.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropItemFolderCommand = OMBDROP ITEM_FOLDER "QUOTED_STRING"
```

Keywords And Parameters

dropItemFolderCommand
To drop an Item Folder.

Examples

```
OMBDROP ITEM_FOLDER 'customer'.
```

See Also

OMBRETRIEVE ITEM_FOLDER, OMBCREATE ITEM_FOLDER, OMBALTER ITEM_FOLDER

OMBDROP LIST_OF_VALUES

Purpose

To drop a List Of Values.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropListOfValuesCommand = OMBDROP LIST_OF_VALUES "QUOTED_STRING"
```

Keywords And Parameters

dropListOfValuesCommand

To drop a List Of Values.

Examples

```
OMBDROP LIST_OF_VALUES 'customer_item'.
```

See Also

OMBRETRIEVE LIST_OF_VALUES, OMBCREATE LIST_OF_VALUES, OMBALTER LIST_OF_VALUES

OMBDROP LOCATION

Purpose

Delete the location.

Prerequisites

Can be in any context.

Syntax

```
dropLocationCommand = OMBDROP ( LOCATION "QUOTED_STRING" )
```

Keywords And Parameters

dropLocationCommand

Drop a location from the repository.

Examples

```
OMBDROP LOCATION 'OLD_LOCATION'
```

This will delete the location "OLD_LOCATION".

See Also

OMBDROP, OMBCREATE LOCATION, OMBALTER LOCATION

OMBDROP MAPPING

Purpose

Drop an existing mapping.

Prerequisites

The current context of scripting must be an Oracle

Syntax

```
dropMappingCommand = OMBDROP MAPPING "mappingName"  
mappingName = "QUOTED_STRING"
```

Keywords And Parameters

dropMappingCommand

Drop an existing mapping.

mappingName

Name of the mapping.

Examples

```
OMBDROP MAPPING 'MAP1'
```

See Also

OMBDROP, OMBCREATE MAPPING, OMBALTER MAPPING, OMBRETRIEVE MAPPING

OMBDROP MATERIALIZED_VIEW

Purpose

To drop a materialized view.

Prerequisites

In the context of an Oracle Module.

Syntax

```
dropMaterializedViewCommand = OMBDROP MATERIALIZED_VIEW "QUOTED_STRING"
```

Keywords And Parameters

dropMaterializedViewCommand

This clause drops a MaterializedView.

Examples

```
OMBDROP MATERIALIZED_VIEW 'NEW_MATERIALIZED_VIEW'.
```

See Also

OMBDROP, OMBCREATE MATERIALIZED_VIEW, OMBALTER MATERIALIZED_VIEW, OMBRETRIEVE MATERIALIZED_VIEW

OMBDROP MDL_ACTION_PLAN

Purpose

Remove an existing metadata loader action plan.

Prerequisites

Connection must be established to the repository.

Syntax

```
dropMDLActionPlanCommand = ( OMBDROP ( MDL_ACTION_PLAN ) "QUOTED_STRING" )
```

Keywords And Parameters

dropMDLActionPlanCommand

Remove an existing metadata loader action plan.

QUOTED_STRING

Enclose the name of the metadata loader action plan in single quotes.

Examples

```
OMBDROP MDL_ACTION_PLAN 'MY_PLAN'
```

See Also

OMBCREATE MDL_ACTION_PLAN, OMBALTER MDL_ACTION_PLAN,
OMBRETRIEVE MDL_ACTION_PLAN, OMBEXPORT MDL_FILE

OMBDROP MINING_MODEL

Purpose

Drop an existing mining model.

Prerequisites

The current context of scripting must be an Oracle Module.

Syntax

```
dropMiningModelCommand = OMBDROP MINING_MODEL "miningModelName"  
miningModelName = "QUOTED_STRING"
```

Examples

```
OMBDROP MINING_MODEL 'MODEL1'
```

See Also

OMBDROP, OMBCREATE MINING_MODEL, OMBALTER MINING_MODEL,
OMBRETRIEVE MINING_MODEL

OMBDROP NESTED_TABLE

Purpose

Delete the Nested Table.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
dropNestedTableCommand = OMBDROP NESTED_TABLE "QUOTED_STRING"
```

Examples

```
OMBDROP NESTED_TABLE 'SOME_NESTED_TABLE'
```

This will delete the "SOME_NESTED_TABLE" Nested Table.

See Also

OMBRETRIEVE NESTED_TABLE, OMBCREATE NESTED_TABLE, OMBALTER NESTED_TABLE

OMBDROP OBJECT_TYPE

Purpose

Delete the Object Type.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
dropObjectTypeCommand = OMBDROP OBJECT_TYPE "QUOTED_STRING"
```

Keywords And Parameters

dropObjectTypeCommand

Drops the Object Type with given name.

Examples

```
OMBDROP OBJECT_TYPE 'SOME_OBJECT_TYPE'
```

This will delete the "SOME_OBJECT_TYPE" Object Type.

See Also

OMBDROP, OMBCREATE OBJECT_TYPE, OMBALTER OBJECT_TYPE

OMBDROP ORACLE_MODULE

Purpose

Delete the Oracle module.

Prerequisites

Should be in the context of project.

Syntax

```
dropOracleModuleCommand = OMBDROP ( ORACLE_MODULE "QUOTED_STRING" )
```

Keywords And Parameters

dropOracleModuleCommand

Remove an existing Oracle module.

QUOTED_STRING

Name of the existing Oracle module in quotes.

Examples

```
OMBDROP ORACLE_MODULE 'src_module'
```

This will delete the "src_module" Oracle module.

See Also

OMBDROP, OMBCREATE ORACLE_MODULE, OMBALTER ORACLE_MODULE

OMBDROP PACKAGE

Purpose

Delete the Package.

Prerequisites

Should be in the context of a Oracle Module or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
dropPackageCommand = OMBDROP ( PACKAGE "QUOTED_STRING" )
```

Keywords And Parameters

dropPackageCommand

Remove an existing Package.

QUOTED_STRING

Name of the existing Package in quotes.

Examples

```
OMBDROP PACKAGE 'package_1'
```

This will delete the "package_1" Package.

See Also

OMBDROP, OMBCREATE PACKAGE, OMBALTER PACKAGE

OMBDROP PLSQL_RECORD_TYPE

Purpose

Delete the PLSQL Record Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
dropPlSqlRecordTypeCommand = OMBDROP PLSQL_RECORD_TYPE "QUOTED_STRING"
```

Keywords And Parameters

dropPlSqlRecordTypeCommand

Drops the PLSQL Record Type with given name.

Examples

```
OMBDROP PLSQL_RECORD_TYPE 'SOME_PLSQL_RECORD_TYPE'
```

This will delete the "SOME_PLSQL_RECORD_TYPE" PLSQL Record Type.

See Also

OMBDROP, OMBCREATE PLSQL_RECORD_TYPE, OMBALTER PLSQL_RECORD_TYPE

OMBDROP PLSQL_REF_CURSOR_TYPE

Purpose

Delete the Ref-cursor Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
dropPLSqlRefCursorTypeCommand = OMBDROP PLSQL_REF_CURSOR_TYPE  
    "QUOTED_STRING"
```

Keywords And Parameters

dropPLSqlRefCursorTypeCommand

Drops the Ref-cursor Type with given name.

Examples

```
OMBDROP PLSQL_REF_CURSOR_TYPE 'SOME_PLSQL_REF_CURSOR_TYPE'
```

This will delete the "SOME_PLSQL_REF_CURSOR_TYPE" Ref-cursor Type.

See Also

OMBDROP, OMBCREATE PLSQL_REF_CURSOR_TYPE, OMBALTER PLSQL_REF_CURSOR_TYPE

OMBDROP PLSQL_TABLE_TYPE

Purpose

Delete the Table Type.

Prerequisites

Should be in the context of a Package.

Syntax

```
dropPlSqlTableTypeCommand = OMBDROP PLSQL_TABLE_TYPE "QUOTED_STRING"
```

Keywords And Parameters

dropPlSqlTableTypeCommand

Drops the Table Type with given name.

Examples

```
OMBDROP PLSQL_TABLE_TYPE 'SOME_PLSQL_TABLE_TYPE'
```

This will delete the "SOME_PLSQL_TABLE_TYPE" Table Type.

See Also

OMBDROP, OMBCREATE PLSQL_TABLE_TYPE, OMBALTER PLSQL_TABLE_TYPE

OMBDROP PLUGGABLE_MAPPING

Purpose

Drop an existing pluggable mapping.

Prerequisites

The current context of scripting must be a project or pluggable map folder.

Syntax

```
dropPluggableMappingCommand = OMBDROP PLUGGABLE_MAPPING "pluggableMapName"  
pluggableMapName = "QUOTED_STRING"
```

Keywords And Parameters

`dropPluggableMappingCommand`

Drop an existing pluggable map folder.

`pluggableMapName`

Name of the pluggable map.

Examples

```
OMBDROP PLUGGABLE_MAP 'PLUGGABLE_MAP1'
```

See Also

OMBDROP, OMBCREATE PLUGGABLE_MAPPING, OMBALTER PLUGGABLE_MAPPING, OMBRETRIEVE PLUGGABLE_MAPPING

OMBDROP PLUGGABLE_MAPPING_FOLDER

Purpose

Drop an existing pluggable map folder.

Prerequisites

The current context of scripting must be a project.

Syntax

```
dropPluggableMappingFolderCommand = OMBDROP PLUGGABLE_MAPPING_FOLDER  
    "pluggableMapFolderName"  
pluggableMapFolderName = "QUOTED_STRING"
```

Keywords And Parameters

`dropPluggableMappingFolderCommand`

Drop an existing pluggable map folder.

`pluggableMapFolderName`

Name of the pluggable map folder.

Examples

```
OMBDROP PLUGGABLE_MAP_FOLDER 'PLUGGABLE_MAP_FOLDER1'
```

See Also

OMBDROP

OMBDROP PRESENTATION_TEMPLATE

Purpose

To drop a presentation template.

Prerequisites

Should be in the context of a business presentation module or use the full path.

Syntax

```
dropReportCommand = OMBDROP PRESENTATION_TEMPLATE "QUOTED_STRING"
```

Keywords And Parameters

dropReportCommand

To drop a presentation template.

Examples

```
OMBDROP PRESENTATION_TEMPLATE 'sales'.
```

See Also

OMBRETRIEVE PRESENTATION_TEMPLATE, OMBCREATE PRESENTATION_TEMPLATE, OMBALTER PRESENTATION_TEMPLATE

OMBDROP PROCEDURE

Purpose

Delete the Procedure.

Prerequisites

Should be in the context of a Oracle Module or Package or Transformation Module.

A Transformation Module may be WB_CUSTOM_TRANS for Public Transformations.

WB_CUSTOM_TRANS may be modified by an administrator.

WB_CUSTOM_TRANS is not dependent on any project.

Syntax

```
dropProcedureCommand = OMBDROP ( PROCEDURE "QUOTED_STRING" )
```

Keywords And Parameters

dropProcedureCommand

Remove an existing Procedure.

QUOTED_STRING

Name of the existing Procedure in quotes.

Examples

```
OMBDROP PROCEDURE 'proc'
```

This will delete the "proc" Procedure.

If Packaged Function is overloaded, first find the Signature by using

OMBLIST command, and then use OMBALTER command using appropriate signature.

Example, if OMBLIST PROCEDURES gives following two signatures,

```
PROC_1 (NUMBER)
```

```
PROC_1 (VARCHAR2, NUMBER)
```

The OMBDROP Syntax to drop the first one will be as follows

```
OMBDROP PROCEDURE 'PROC_1 \ (NUMBER\ )'
```

See Also

OMBDROP, OMBCREATE PROCEDURE, OMBALTER PROCEDURE

OMBDROP PROCESS_FLOW

Purpose

Delete the Process Flow.

Prerequisites

Should be in the context of a Process Flow Package.

Syntax

```
dropProcessFlowCommand = OMBDROP ( PROCESS_FLOW "QUOTED_STRING" )
```

Keywords And Parameters

dropProcessFlowCommand

Delete a process flow.

Examples

```
OMBDROP PROCESS_FLOW 'process_flow'
```

This will delete the "process_flow" Process Flow.

See Also

OMBDROP, OMBCREATE PROCESS_FLOW, OMBALTER PROCESS_FLOW

OMBDROP PROCESS_FLOW_MODULE

Purpose

Delete the Process Flow Module.

Prerequisites

Should be in the context of a project.

Syntax

```
dropProcessFlowModuleCommand = OMBDROP ( PROCESS_FLOW_MODULE  
    "QUOTED_STRING" )
```

Keywords And Parameters

dropProcessFlowModuleCommand

Drop an existing process flow module.

Examples

```
OMBDROP PROCESS_FLOW_MODULE 'process_module'
```

This will delete the "process_module" Process Flow Module.

See Also

OMBDROP, OMBCREATE PROCESS_FLOW_MODULE, OMBALTER PROCESS_FLOW_MODULE

OMBDROP PROCESS_FLOW_PACKAGE

Purpose

Delete the Process Flow Package.

Prerequisites

Should be in the context of a Process Flow Module.

Syntax

```
dropProcessFlowPackageCommand = OMBDROP ( PROCESS_FLOW_PACKAGE  
    "QUOTED_STRING" )
```

Keywords And Parameters

dropProcessFlowPackageCommand

Delete a process flow package.

Examples

```
OMBDROP PROCESS_FLOW_PACKAGE 'process_package'
```

This will delete the "process_package" Process Flow Package.

See Also

OMBDROP, OMBCREATE PROCESS_FLOW_PACKAGE, OMBALTER PROCESS_FLOW_PACKAGE

OMBDROP PROJECT

Purpose

Delete the project.

Prerequisites

Should be in the top level context.

Syntax

```
dropProjectCommand = OMBDROP ( PROJECT "QUOTED_STRING" )
```

Keywords And Parameters

dropProjectCommand

Remove an existing project.

QUOTED_STRING

Name of the existing project in quotes.

Examples

```
OMBDROP PROJECT 'New Project'
```

This will delete the "New Project" project.

See Also

OMBDROP, OMBCREATE PROJECT, OMBALTER PROJECT

OMBDROP QUEUE_PROPAGATION

Purpose

Delete the Queue Propagation.

Prerequisites

Should be in the context of an Advanced Queue.

Syntax

```
dropQPCommand = OMBDROP QUEUE_PROPAGATION "QUOTED_STRING"
```

Keywords And Parameters

dropQPCommand

Drops the Queue Propagation with the given name.

Examples

```
OMBDROP QUEUE_PROPAGATION 'SOME_QUEUE_PROPAGATION'
```

This will delete the "SOME_QUEUE_PROPAGATION" Queue Propagation.

See Also

OMBDROP, OMBCREATE QUEUE_PROPAGATION, OMBALTER QUEUE_PROPAGATION, OMBRETRIEVE QUEUE_PROPAGATION

OMBDROP QUEUE_TABLE

Purpose

Delete the Queue Table.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
dropQTCommand = OMBDROP QUEUE_TABLE "QUOTED_STRING"
```

Keywords And Parameters

dropQTCommand

Drops the Queue Table with the given name.

Examples

```
OMBDROP QUEUE_TABLE 'SOME_QUEUE_TABLE'
```

This will delete the "SOME_QUEUE_TABLE" Queue Table.

See Also

OMBDROP, OMBCREATE QUEUE_TABLE, OMBALTER QUEUE_TABLE,
OMBRETRIEVE QUEUE_TABLE

OMBDROP REAL_TIME_MAPPING

Purpose

Drop an existing Real Time mapping.

Prerequisites

The current context of scripting must be an Oracle

Syntax

```
dropRealTimeMappingCommand = OMBDROP REAL_TIME_MAPPING "mappingName"  
mappingName = "QUOTED_STRING"
```

Keywords And Parameters

mappingName

Name of the mapping.

Examples

```
OMBDROP REAL_TIME_MAPPING 'MAP1'
```

See Also

OMBDROP, OMBCREATE REAL_TIME_MAPPING, OMBALTER REAL_TIME_MAPPING, OMBRETRIEVE REAL_TIME_MAPPING

OMBDROP REGISTERED_FUNCTION

Purpose

To drop an I/O Function.

Prerequisites

Should be in the context of a Business Definition Module or use the full path.

Syntax

```
dropRegisteredFunctionCommand = OMBDROP REGISTERED_FUNCTION  
    "QUOTED_STRING"
```

Keywords And Parameters

dropRegisteredFunctionCommand

To drop an I/O Function.

Examples

```
OMBDROP REGISTERED_FUNCTION 'sum'.
```

See Also

OMBRETRIEVE REGISTERED_FUNCTION, OMBCREATE REGISTERED_FUNCTION, OMBALTER REGISTERED_FUNCTION

OMBDROP ROLE

Purpose

To drop a Warehouse Builder role.

Prerequisites

Must be connected to a OWB repository.

Syntax

```
parseDropRoleCommand = ( OMBDROP ROLE "QUOTED_STRING" )
```

Keywords And Parameters

parseDropRoleCommand

This clause drops a Warehouse Builder role.

Examples

OMBDROP ROLE 'DEVELOPMENT_ROLE'
will drop role 'DEVELOPMENT_ROLE'.

See Also

OMBCREATE ROLE, OMBALTER ROLE, OMBRETRIEVE ROLE, OMBLIST ROLES

OMBDROP SAP_MODULE

Purpose

Remove an existing SAP module.

Prerequisites

You must open a project to drop a SAP module.

Syntax

```
dropSAPModuleCommand = OMBDROP ( SAP_MODULE "QUOTED_STRING" )
```

Keywords And Parameters

dropSAPModuleCommand

Remove an existing SAP module.

Examples

```
OMBDROP SAP_MODULE 'src_module'
```

This will delete the "src_module" SAP module.

See Also

OMBDROP

OMBDROP SEQUENCE

Purpose

To drop a sequence.

Prerequisites

In the context of an Oracle Module.

Syntax

```
dropSequenceCommand = OMBDROP SEQUENCE "QUOTED_STRING"
```

Keywords And Parameters

dropSequenceCommand

This clause drops a Sequence.

Examples

```
OMBDROP SEQUENCE 'new_sequence'.
```

See Also

OMBDROP, OMBCREATE SEQUENCE, OMBALTER SEQUENCE, OMBRETRIEVE SEQUENCE

OMBDROP SNAPSHOT

Purpose

A snapshot can be dropped.

Prerequisites

The snapshot to be dropped should already exist. This command can be executed for any snapshot regardless of current context.

Syntax

```
parseDropCommand = OMBDROP ( SNAPSHOT "QUOTED_STRING" )
```

Keywords And Parameters

parseDropCommand

To drop a snapshot.

QUOTED_STRING

Name of snapshot to be dropped.

Examples

```
OMBDROP SNAPSHOT 'S1'
```

See Also

OMBCREATE SNAPSHOT, OMBALTER SNAPSHOT, OMBRESTORE SNAPSHOT, OMBCOMPARE SNAPSHOT, OMBLIST SNAPSHOT, OMBRETRIEVE SNAPSHOT

OMBDROP STREAMS_CAPTURE_PROCESS

Purpose

Delete the Streams Capture Process.

Prerequisites

Should be in the context of a Streams Queue.

Syntax

```
dropCaptureCommand = OMBDROP STREAMS_CAPTURE_PROCESS "QUOTED_STRING"
```

Keywords And Parameters

dropCaptureCommand

Drops the Streams Capture Process with the given name.

Examples

```
OMBDROP STREAMS_CAPTURE_PROCESS 'SOME_CAPTURE_PROCESS'
```

This will delete the "SOME_CAPTURE_PROCESS" Streams Capture Process.

See Also

OMBDROP, OMBCREATE STREAMS_CAPTURE_PROCESS, OMBALTER STREAMS_CAPTURE_PROCESS, OMBRETRIEVE STREAMS_CAPTURE_PROCESS

OMBDROP STREAMS_QUEUE

Purpose

Delete the Streams Queue.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
dropANYQCommand = OMBDROP STREAMS_QUEUE "QUOTED_STRING"
```

Keywords And Parameters

dropANYQCommand

Drops the Streams Queue with the given name.

Examples

```
OMBDROP STREAMS_QUEUE 'SOME_STREAMS_QUEUE'
```

This will delete the "SOME_STREAMS_QUEUE" Streams Queue.

See Also

OMBDROP, OMBCREATE STREAMS_QUEUE, OMBALTER STREAMS_QUEUE, OMBRETRIEVE STREAMS_QUEUE

OMBDROP TABLE

Purpose

To drop a table.

Prerequisites

In the context of an Oracle Module.

Syntax

```
dropTableCommand = OMBDROP TABLE "QUOTED_STRING"
```

Keywords And Parameters

dropTableCommand

This clause drops a table.

Examples

```
OMBDROP TABLE 'old_table'.
```

See Also

OMBDROP, OMBCREATE TABLE, OMBALTER TABLE, OMBRETRIEVE TABLE

OMBDROP TABLE_FUNCTION

Purpose

Delete the Table Function.

Prerequisites

Should be in the context of Oracle Module or Package.

Syntax

```
dropTableFunctionCommand = OMBDROP ( TABLE_FUNCTION "QUOTED_STRING" )
```

Keywords And Parameters

dropTableFunctionCommand

Drops a Table Function

Examples

```
OMBDROP TABLE_FUNCTION 'table_function'
```

This will delete the "table_function" Table Function.

See Also

OMBDROP, OMBCREATE TABLE_FUNCTION, OMBALTER TABLE_FUNCTION

OMBDROP TIME_DIMENSION

Purpose

This command drops a time dimension.

Prerequisites

Should be in Oracle Module context.

Syntax

```
dropTimeDimensionCommand = OMBDROP TIME_DIMENSION "TimeDimensionName"  
TimeDimensionName = "QUOTED_STRING"
```

Keywords And Parameters

TimeDimensionName

The name of the time dimension.

Examples

```
OMBDROP TIME_DIMENSION 'YR2005'
```

See Also

OMBCREATE TIME_DIMENSION, OMBALTER TIME_DIMENSION,
OMBRETRIEVE TIME_DIMENSION

OMBDROP TRANSPORTABLE_MODULE

Purpose

To drop a transportable module.

Prerequisites

In the context of a project.

Syntax

```
dropTMCommand = OMBDROP TRANSPORTABLE_MODULE "QUOTED_STRING"
```

Keywords And Parameters

dropTMCommand

This command is for dropping a transportable module.

QUOTED_STRING

The transportable module to be dropped.

Examples

```
OMBDROP TRANSPORTABLE_MODULE 'TM101'
```

This command will drop the transportable module TM101 and all its content.

See Also

OMBCREATE TRANSPORTABLE_MODULE

OMBDROP VARYING_ARRAY

Purpose

Delete the Varying Array.

Prerequisites

Should be in the context of an Oracle Module.

Syntax

```
dropVaryingArrayCommand = OMBDROP VARYING_ARRAY "QUOTED_STRING"
```

Examples

```
OMBDROP VARYING_ARRAY 'SOME_VARRAY'
```

This will delete the "SOME_VARRAY" Varying Array.

See Also

OMBRETRIEVE VARYING_ARRAY, OMBCREATE VARYING_ARRAY, OMBALTER VARYING_ARRAY

OMBDROP VIEW

Purpose

To drop a view.

Prerequisites

In the context of an Oracle Module.

Syntax

```
dropViewCommand = OMBDROP VIEW "QUOTED_STRING"
```

Keywords And Parameters

dropViewCommand

This clause drops a View.

Examples

```
OMBDROP VIEW 'NEW_VIEW'.
```

See Also

OMBDROP, OMBCREATE VIEW, OMBALTER VIEW, OMBRETRIEVE VIEW

OMU Commands

This chapter lists commands associated with OMU commands in alphabetical order.

OMUALTER

Purpose

Invoke the editing GUI from OWB for the specified object.

Prerequisites

Must be in the parent context of the object.

Syntax

```
parseAlterCommand = OMUALTER "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Keywords And Parameters

`parseAlterCommand`

Invoke the corresponding editing GUI for the object. This command returns the type and name of the object being altered, or empty string if the GUI is cancelled.

`objectType`

The type of object to be edited. This can be any component or folder type. For example, TABLE, VIEW, or MAPPING.

Examples

This command will launch the data object editor for editing a table T1 under the current oracle module context:

```
OMUALTER TABLE 'T1'
```

See Also

OMUCREATE

OMUANALYZEIMPACT

Purpose

omushowimpactPurposeTag??

Prerequisites

omushowimpactPreTag??

Syntax

```
parseShowImpactCommand = OMUANALYZEIMPACT "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

omushowimpactExampleTag??

OMUANALYZELINEAGE

Purpose

omushowlineagePurposeTag??

Prerequisites

omushowlineagePreTag??

Syntax

```
parseShowLineageCommand = OMUANALYZELINEAGE "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

omushowlineageExampleTag??

OMUCOMPILE

Purpose

Invoke the generation GUI from OWB for the specified object.

Prerequisites

Must be in the parent context of the object.

Syntax

```
parseCompileCommand = OMUCOMPILE "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Keywords And Parameters

`parseCompileCommand`

Invoke the generation GUI for the specified object.

`QUOTED_STRING`

The object name.

`objectType`

The type of object to be compiled. For example, TABLE, VIEW, or MAPPING.

Examples

This command will launch the generation results dialog for table T1 under the current oracle module context:

```
OMUCOMPILE TABLE 'T1'
```

OMUCONFIGURE

Purpose

To launch the graphical version of the configuration inspector.

Prerequisites

The object specified must be configurable within the current project.

Syntax

```
parseConfigureCommand = OMuConfigure "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

```
OMuConfigure TABLE 'MY_TABLE'
```

OMUCONNECT

Purpose

To connect to repository using a graphical interface.

Prerequisites

Must not be connected to a repository.

Syntax

```
parseConnectCommand = OMUCONNECT
```

Keywords And Parameters

parseConnectCommand

Provides a graphical interface to connect to repository.

Examples

```
OMUCONNECT
```

OMUCONTROLCENTER

Purpose

To show the Control Center Manager.

Prerequisites

A Control Center connection. Note, a Design Repository connection is not required.

Syntax

```
parseControlCenterCommand = OMUCONTROLCENTER "QUOTED_STRING"
```

Examples

```
OMUCONTROLCENTER 'DEFAULT_CONTROL_CENTER'
```


OMUCONTROLCENTERJOBS

Purpose

To show jobs running in the Control Center.

Prerequisites

A Control Center connection. Note, a Design Repository connection is not required.

Syntax

```
parseControlCenterJobsCommand = OMUCONTROLCENTERJOBS [ "QUOTED_STRING" ]
```

Examples

```
OMUCONTROLCENTERJOBS 'DEFAULT_CONTROL_CENTER'
```

See Also

OMUALTER

OMUCREATE

Purpose

Invoke the creation GUI from OWB for the specified object type.

Prerequisites

Must be in the parent context for the object type.

Syntax

```
parseCreateCommand = OMUCREATE "objectType" [ SET "setPropertiesClause" ]
objectType = "UNQUOTED_STRING"
setPropertiesClause = PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
propertyNameList = "UNQUOTED_STRING" { "," "UNQUOTED_STRING" }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parseCreateCommand

Invoke the corresponding creation GUI for the object type. This command returns the type and name of the object(s) being created, or empty string if the GUI is cancelled.

objectType

The type of object to be created. This can be any component or folder. For examples, TABLE, VIEW, MAPPING, and so on

setPropertiesClause

This clause sets the properties needed for launching the GUI.

propertyNameList

The list of property names to set.

Properties for LOCATION:

Name: TYPE

Type: STRING

Valid Values: See OMBCREATE LOCATION for valid values.

Default: 'ORACLE_DATABASE'

The type of system the location represents.

Properties for CONNECTOR:

Name: TYPE

Type: STRING

Valid Values: 'DB_CONNECTOR', 'DIRECTORY'

Default: 'DB_CONNECTOR'

The type of connector to create.

Properties for GATEWAY_MODULE:

Name: GATEWAY_TYPE

Type: STRING

Valid Values: See OMBCREATE GATEWAY_MODULE for valid values.

Default: N/A

The type of gateway module to create.

propertyValueList

The list of property values.

propertyValue

The value of a property.

Examples

This command will launch the data object editor for creating a table under the current oracle module context:

```
OMUCREATE TABLE
```

See Also

```
OMUALTER
```

OMUDATAVIEWER

Purpose

To display Dataviewer for relational and dimensional objects to business definition objects.

Prerequisites

A relational object or dimensional object must exist.

Syntax

```
parseDataviewerCommand = OMUDATAVIEWER ( DIMENSION | CUBE | TABLE | VIEW )  
    "QUOTED_STRING"
```

Examples

```
OMUDATAVIEWER TABLE 'MY_TABLE'
```

OMUDEPLOY

Purpose

To Deploy Action Plans to Control Centers.

Prerequisites

A Control Center connection and a named Deployment Action Plan are required. Also, the current Context must be either an Oracle Module, a Process Flow Module or a Location.

Syntax

```
parseDeployCommand = OMUDEPLOY "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

```
OMUDEPLOY DEPLOYMENT_ACTION_PLAN 'MY_DEPLOY_PLAN'
```

See Also

OMUALTER

OMUDERIVE

Purpose

To derive relational and dimensional objects to business definition objects.

Prerequisites

A relational object or dimensional object must exist.

Syntax

```
parseDeriveCommand = OMUDERIVE ( ORACLE_MODULE | DIMENSION | CUBE | TABLE  
    | VIEW | EXTERNAL_TABLE | FUNCTION | COLLECTION ) "QUOTED_STRING"
```

Keywords And Parameters

parseDeriveCommand

To derive relational and dimensional objects to business definition objects.

Examples

```
OMUDERIVE TABLE 'MY_TABLE'
```

OMUEXPORT MDL_FILE

Purpose

Exports the metadata from the repository based on the metadata loader action plan.

Prerequisites

Connection must be established to repository to be exported from.

Syntax

```
parseMDLExportCommand = OMUEXPORT MDL_FILE [ "QUOTED_STRING" ]  
                        MDL_ACTION_PLAN "QUOTED_STRING"
```

Keywords And Parameters

parseMDLExportCommand

Exports the metadata from the repository

QUOTED_STRING

Enclose the name of the export metadata file in single quotes.

MDL_ACTION_PLAN

Specify the MDL_ACTION_PLAN to be used by the export. Enclose the metadata loader action plan in single quotes.

Examples

```
OMUEXPORT MDL_FILE 'd:/mdl/exp1.mdl' MDL_ACTION_PLAN 'MY_ACTION_  
PLAN'
```

```
OMUEXPORT MDL_FILE MDL_ACTION_PLAN 'MY_ACTION_PLAN'
```

See Also

OMBCREATE MDL_ACTION_PLAN, OMBALTER MDL_ACTION_PLAN,
OMBDROP MDL_ACTION_PLAN, OMBRETRIEVE MDL_ACTION_PLAN,
OMUIMPORT MDL_FILE

OMUIMPACT

Purpose

To perform impact analysis on the specified object. Metadata Dependency Manager will show up with an impact graph of the specified object.

Prerequisites

The object specified can be source or target object.

Syntax

```
parseShowImpactCommand = OMUIMPACT "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Keywords And Parameters

`parseShowImpactCommand`
Shows the impact of an object.

`objectType`
Type of the object whose impact graph needs to be shown.

Examples

```
OMUIMPACT TABLE 'MY_TABLE'
```


OMUIMPORT

Purpose

To launch the graphical version of the import wizard.

Prerequisites

Must be in the context of a module.

Syntax

```
parseDatabaseImportCommand = METADATA_LOCATION [ "objectType" [ (
    MINIMAL_MODE | FULL_MODE ) ] ]
objectType = "UNQUOTED_STRING"
```

Examples

OMUIMPORT FROM METADATA_LOCATION TABLE FULL_MODE will invoke
import wizard
with only tables.

OMUIMPORT MDL_FILE

Purpose

Imports the metadata from a file into the repository

Prerequisites

Must be connected to the repository where the import is to be performed.

Syntax

```
parseMDLImportCommand = MDL_FILE [ "QUOTED_STRING" ]
```

Keywords And Parameters

parseMDLImportCommand

Import metadata from a file.

QUOTED_STRING

Enclose the name of the import metadata file in single quotes.

Examples

```
OMUIMPORT MDL_FILE 'd:/mdl/exp1.mdl'
```

```
OMUIMPORT MDL_FILE
```

See Also

```
OMUEXPORT MDL_FILE
```

OMULINEAGE

Purpose

To perform lineage analysis on the specified object. Metadata Dependency Manager will show up with a lineage graph of the specified object.

Prerequisites

The object specified can be source or target object.

Syntax

```
parseShowLineageCommand = OMULINEAGE "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Keywords And Parameters

parseShowLineageCommand

Shows the lineage of an object.

objectType

Type of the object whose lineage graph needs to be shown.

Examples

```
OMULINEAGE TABLE 'MY_TABLE'
```

OMULIST

Purpose

Run the object selector dialog for selecting some objects.

Prerequisites

Connect to the repository and in the parent context.

Syntax

```
parseListCommand = OMULIST "objectType"  
objectType = "UNQUOTED_STRING"
```

Keywords And Parameters

`parseListCommand`

Show an object selector dialog for choosing the specified object type.

`objectType`

Specify the object type that the object selector dialog should show. This object type can be any component or folder object type name in plural. For example, TABLES, VIEWS, MAPPINGS, and so on

Examples

This command will show a list of all tables to be selected under the current oracle module context:

```
OMULIST TABLES
```

OMUPROMPT

Purpose

To construct a custom UI in order to prompt for some information from the end user.

Prerequisites

None.

Syntax

```
parsePromptCommand = OMUPROMPT "type" "name" [ "setPropertiesClause" ] {
    "addComponentClause" }
type = "UNQUOTED_STRING"
name = "QUOTED_STRING"
setPropertiesClause = SET PROPERTIES "(" "propertyNameList" ")" VALUES "("
    "propertyValueList" ")"
addComponentClause = ADD "type" "name" [ "setPropertiesClause" ]
propertyNameList = PROPERTY_NAME { "," PROPERTY_NAME }
propertyValueList = "propertyValue" { "," "propertyValue" }
propertyValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parsePromptCommand

Construct a custom UI to prompt for information. This clause specifies the name of the dialog and the properties for the dialog.

type

The type of custom dialog or its component. Currently only DIALOG is valid custom UI type. For the type of a component UI, the valid types include CHECK_BOX, CHOICE_BOX, COMBO_BOX, FILE_CHOOSER, HYPERLINK, LABEL, LIST_BOX, PASSWORD_FIELD, RADIO_BUTTONS, SEPARATOR, TEXT_FIELD.

name

The name of the custom UI or UI component.

setPropertiesClause

This clause sets the properties for the custom UI or the UI component.

Basic properties for CHECK_BOX :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The text to be displayed by the side of the check box.

Name: SELECTION

Type: BOOLEAN

Valid Values: true, false

Default: false

Whether the check box should be selected by default.

Basic properties for CHOICE_BOX :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed by the side of the choice box that describes what the selection is about.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The different choices shown in the choice box. Needs to be a

comma-delimited list.

Name: SELECTION

Type: STRING

Valid Values: N/A

Default: "

The default selection of the choices.

Basic properties for COMBO_BOX :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed by the side of the combo box that describes what the selection is about.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The different choices shown in the combo box. Needs to be a comma-delimited list.

Name: SELECTION

Type: STRING

Valid Values: N/A

Default: "

The default selection of the choices.

Basic properties for DIALOG :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: TITLE

Type: STRING

Valid Values: N/A

Default: "

Title of the custom dialog.

Name: WIDTH

Type: NUMBER

Valid Values: N/A

Default: "

The width of the custom dialog. If either the height or the width is unspecified or set to 0, then the dialog will be auto-sized.

Name: HEIGHT

Type: NUMBER

Valid Values: N/A

Default: "

The height of the custom dialog. If either the height or the width is unspecified or set to 0, then the dialog will be auto-sized.

Name: LINE_WRAP

Type: NUMBER

Valid Values: N/A

Default: 2

CUSTOMUL_DIALOG_LINE_WRAP_DESC??

Name: OPTIONS

Type: STRING

Valid Values: N/A

Default: "

The options for the dialog. These translate into buttons at the bottom of the dialog. A comma-delimited list of values is expected here. For example, a dialog with a YES and No button will need an OPTION of 'YES,NO'.

Name: DEFAULT_OPTION

Type: STRING

Valid Values: N/A

Default: "

The default option of the dialog. This means the default button that will be pressed when the ENTER key is pressed.

Name: CANCEL_OPTION

Type: STRING

Valid Values: N/A

Default: "

The option that represents cancellation of the dialog.

Basic properties for FILE_CHOOSER :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: .

The default directory to be shown in the file chooser.

Name: SELECTION

Type: STRING

Valid Values: DIRECTORIES_ONLY, FILES_ONLY, FILES_AND_DIRECTORIES

Default: FILES_AND_DIRECTORIES

Tells whether the file chooser should allow selection of files only, directories only, or both files and directories.

Basic properties for HYPERLINK :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed as text for the link.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The URL of the link target location.

Basic properties for LABEL :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The content of the label. This can be HTML-formatted text.

Basic properties for LIST_BOX :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed on top of the list box that describes what the selection is about.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The different choices shown in the list box. Needs to be a comma-delimited list.

Name: SELECTION

Type: STRING

Valid Values: N/A

Default: "

The default selection of the choices.

Basic properties for PASSWORD_FIELD :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed by the side of the password field that describes what the field is about.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The default content of the password field.

Basic properties for RADIO_BUTTONS :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed on top of the radio buttons that describes what the selection is about.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The different choices shown for the radio buttons. Needs to be a comma-delimited list.

Name: SELECTION

Type: STRING

Valid Values: N/A

Default: "

The default selection of the choices.

Basic properties for SEPARATOR :

Basic properties for TEXT_FIELD :

Name: RETURN_VALUE

Type: STRING

Valid Values: N/A

Default: "

The return value for this component.

Name: MESSAGE_TEXT

Type: STRING

Valid Values: N/A

Default: "

The message text to be displayed by the side of the text field that describes what the field is about.

Name: CONTENT

Type: STRING

Valid Values: N/A

Default: "

The default content of the text field.

addComponentClause

Add a UI component to the custom UI.

propertyNameList

The list of property names.

propertyValueList

The list of property values to set.

propertyValue

The value of the property.

Examples

This command will prompt for a password from the end user:

```
OMUPROMPT DIALOG 'DLG1' \  
SET PROPERTIES (TITLE, WIDTH, HEIGHT, OPTIONS) VALUES ('Title', 0, 0,  
'OK,Cancel') \  
ADD LABEL 'LBL1' SET PROPERTIES (CONTENT) VALUES ('Please enter the  
password below') \  
ADD PASSWORD_FIELD 'PWD1' SET PROPERTIES (MESSAGE_TEXT, CONTENT)  
VALUES  
( 'Password:', 'default_password')
```

OMUPROPAGATECHANGE

Purpose

omupropagateChangePurposeTag??

Prerequisites

omupropagateChangePreTag??

Syntax

```
parsePropagateChangeCommand = OMUPROPAGATECHANGE "objectType"  
    "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

omupropagateChangeExampleTag??

OMUPROPERTIES

Purpose

To launch the graphical version of the object property inspector.

Prerequisites

The object specified must be within the current project.

Syntax

```
parsePropertiesCommand = OMUPROPERTIES "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

```
OMUPROPERTIES TABLE 'MY_TABLE'
```


OMUSELECTSOURCE

Purpose

To select the source where data is going to move from: type (oracle, gateway or file) and location.

Prerequisites

Must be in the context of a project.

Syntax

```
parseSelectSrcCommand = OMUSELECTSOURCE
```

Examples

```
OMUSELECTSOURCE
```

OMUSELECTTARGET

Purpose

To select the target where data is going to move to: type (oracle only) and location.

Prerequisites

Must be in the context of a project.

Syntax

```
parseSelectTgtCommand = OMUSELECTTARGET
```

Examples

```
OMUSELECTTARGET
```

OMUSHOWCHANGECONSOLE

Purpose

To show the console containing all snapshots.

Prerequisites

Must be connected to the repository.

Syntax

```
parseShowChangeConsoleCommand = OMUSHOWCHANGECONSOLE
```

Keywords And Parameters

`parseShowChangeConsoleCommand`

Shows the console containing all snapshots.

Examples

```
OMUSHOWCHANGECONSOLE
```

OMUSHOWLIA

Purpose

omushowliaPurposeTag??

Prerequisites

omushowliaPreTag??

Syntax

```
parseUDMCommand = OMUSHOWLIA "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

omushowliaExampleTag??

OMUSTART

Purpose

To start running the expert assistant.

Prerequisites

Connect to a repository.

Syntax

```
parseStartExpertCommand = OMUSTART EXPERT "QUOTED_STRING" [
    "parameterClause" ]
parameterClause = WITH PARAMETERS "(" "parameterNameList" ")" VALUES "("
    "parameterValueList" ")"
parameterNameList = "UNQUOTED_STRING" { ", " "UNQUOTED_STRING" }
parameterValueList = "parameterValue" { ", " "parameterValue" }
parameterValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

parseStartExpertCommand

Start an expert.

parameterClause

Supply a list of input parameter values to be passed into the expert.

parameterNameList

The list of input parameter names.

parameterValueList

The list of parameter values to be passed in.

parameterValue

A parameter value in quoted string, can be STRING, NUMBER, BOOLEAN or ARRAY.

Examples

This command will launch expert assistant on the expert

'/MY_PROJECT/EM1/EXP1':

```
OMUSTART EXPERT '/MY_PROJECT/EM1/EXP1'
```

See Also

OMBCREATE EXPERT, OMBALTER EXPERT, OMUCREATE, OMUALTER

OMUSTARTJOB

Purpose

To launch the graphical version of object execution .

Prerequisites

'Best Practice' usage requires a Design Repository connection and the context to be set to that of executable object. The 'Control Center Only' usage does not require a Design Repository connection and therefore no context.

Syntax

```
parseStartJobCommand = OMUSTARTJOB "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Examples

```
OMUSTARTJOB MAPPING 'MY_MAP'
```

OMUSTART EXPERT

Purpose

To start running the expert assistant.

Prerequisites

Connect to a repository.

Syntax

```
parseStartExpertCommand = OMUSTART EXPERT "QUOTED_STRING" [  
    "parameterClause" ]  
parameterClause = WITH PARAMETERS "(" "parameterNameList" ")" VALUES "("  
    "parameterValueList" ")"  
parameterNameList = "UNQUOTED_STRING" { ",", "UNQUOTED_STRING" }  
parameterValueList = "parameterValue" { ",", "parameterValue" }  
parameterValue = ( "QUOTED_STRING" | "INTEGER_LITERAL" |  
    "FLOATING_POINT_LITERAL" )
```

Keywords And Parameters

`parseStartExpertCommand`

Start an expert.

`parameterClause`

Supply a list of input parameter values to be passed into the expert.

`parameterNameList`

The list of input parameter names.

`parameterValueList`

The list of parameter values to be passed in.

`parameterValue`

A parameter value in quoted string, can be STRING, NUMBER, BOOLEAN or ARRAY.

Examples

This command will launch expert assistant on the expert

```
'/MY_PROJECT/EM1/EXP1':
```

```
OMUSTART EXPERT '/MY_PROJECT/EM1/EXP1'
```


See Also

OMBCREATE EXPERT, OMBALTER EXPERT, OMUCREATE, OMUALTER

OMUVALIDATE

Purpose

Invoke the validation GUI from OWB for the specified object.

Prerequisites

Must be in the parent context of the object.

Syntax

```
parseValidateCommand = OMUVALIDATE "objectType" "QUOTED_STRING"  
objectType = "UNQUOTED_STRING"
```

Keywords And Parameters

`parseValidateCommand`

Invoke the validation GUI for the specified object.

`QUOTED_STRING`

The object name.

`objectType`

The type of object to be validated. For example, TABLE, VIEW, or MAPPING.

Examples

This command will launch the validation results dialog for table T1 under the current oracle module context:

```
OMUVALIDATE TABLE 'T1'
```

Additional and Optional Usages

This chapter includes the following topics:

- [Using Control Files to Import and Export Metadata](#)
- [Working with Mappings and Operators](#)
- [Accessing Transformation Modules](#)
- [Running OMB Plus in Oracle JDeveloper](#)

Using Control Files to Import and Export Metadata

Control files enable you to specify additional options while importing or exporting metadata using the OMB Plus commands OMBIMPORT and OMBEXPORT.

Creating MDL Control Files

An MDL control file is a text file that contains a set of parameters that are used by the Metadata Loader. These parameters specify the options to be used while exporting or importing metadata.

You can use any text editor to create a control file. Oracle recommends that you use the extension .ctl for control files. This helps identify it as an MDL control file.

The format for an export or import parameter is:

Keyword=Value

You can also form a parameter file by replacing the value with the wildcard character (*), which matches any string, or with a list of named objects:

Keyword=*

Keyword=(value-1, value-2, ..., -k)

For example, you can specify the keyword TABLES followed by the names of the tables to import as follows:

```
TABLES=(Customers, Products, Days)
```

You can use the comment indicator (#) to place comments in the control file. Put the comment indicator in the first column of a record and follow it with text.

Control File Example

An example of a control file is shown in [Example A-1](#). This control file contains a list of object types (keywords) and their object names (values) to import from the MDL

file. This is useful since the OMBIMPORT command does not provide an option to select the objects that you want to import from an MDL file.

Example A–1 Control File Format

```
PROJECT=MY_PROJECT
ORACLE_MODULES=DW1
TABLES=TABLE1
ORACLE_MODULES=DW2
DIMENSIONS=DIM1 , DIM2
```

Keywords Used to Import Metadata

You use keywords to form the parameters specified to selectively import objects using a control file. [Table A–1](#) provides a list of keywords for object types that you use to select objects to import.

Table A–1 Object Types Keywords for Importing Objects

Object Type Keyword	Description
ACTION	Used only for the Action Plan. The import mode must be MODE = ACTIONPLAN. The options are: <ul style="list-style-type: none"> ▪ CREATE ▪ REPLACE ▪ UPDATE ▪ INCREMENTALUPDATE (Merge mode) ▪ NONE (do not import the object) If a MODE parameter is not included, then the default is CREATE.
ACTIVITY_TEMPLATE_FOLDERS	Use this keyword to specify the activity template folders to be imported.
ACTIVITY_TEMPLATES	Use this keyword to specify the activity templates to be imported. Requires the ACTIVITY_TEMPLATE_FOLDERS to be specified.
ADVANCED_QUEUES	Use this keyword to specify the advanced queues to be imported. Requires the ORACLE_MODULES option to be specified.
BUSINESS_AREAS	Use this keyword to specify the business areas to be imported. Requires the BUSINESS_DEFINITION_MODULES option to be specified.
BUSINESS_DEFINITION_MODULES	Use this keyword to specify the business definition modules to be imported. Requires the PROJECT option to be specified.

Table A-1 (Cont.) Object Types Keywords for Importing Objects

Object Type Keyword	Description
BUSINESS_PRESENTATION_MODULES	Use this keyword to specify the business presentation modules to be imported. Requires the PROJECT option to be specified.
CALENDAR_MODULES	Use this keyword to specify the calendar modules to be imported.
CALENDARS	Use this keyword to specify the calendars to be imported. Requires the CALENDAR_MODULES option to be specified.
COLLECTIONS	Use this keyword to specify the collections to be imported. Requires the PROJECT option to be specified.
CONFIGURATIONS	Use this keyword to specify the configurations to be imported.
CONNECTORS	Use this keyword to specify the connectors to be imported. Requires the LOCATIONS option to be specified.
CONTROL_CENTERS	Use this keyword to specify the control centers to be imported. Requires the PROJECT option to be specified.
CUBES	Use this keyword to specify the cubes to be imported. Requires the ORACLE_MODULES option to be specified.
DATA_AUDITORS	Use this keyword to specify the data auditors to be imported. Requires the ORACLE_MODULES option to be specified.
DATA_RULE_MODULES	Use this keyword to specify the data rule modules to be imported. Requires the PROJECT option to be specified.
DATA_RULES	Use this keyword to specify the data rules to be imported. Requires the DATA_RULE_MODULES option to be specified.
DATA_PROFILES	Use this keyword to specify the data profiles to be imported. Requires the PROJECT option to be specified.
DEPLOYMENTS	Use this keyword to specify the deployments to be imported.

Table A-1 (Cont.) Object Types Keywords for Importing Objects

Object Type Keyword	Description
DIMENSIONS	Use this keyword to specify the dimensions to be imported. Requires the ORACLE_MODULES option to be specified.
DRILL_PATHS	Use this keyword to specify the drill paths to be imported. Requires the BUSINESS_DEFINITION_MODULES option to be specified.
EXPERT_MODULES	Use this keyword to specify the expert modules to be imported. Requires the PROJECT option to be specified.
EXPERTS	Use this keyword to specify the experts to be imported. Requires the EXPERT_MODULES option to be specified.
EXTERNAL_TABLES	Use this keyword to specify the external tables to be imported. Requires the ORACLE_MODULES option to be specified.
FLAT_FILE_MODULES	Use this keyword to specify the flat file modules to be imported. Requires the PROJECT option to be specified.
FLAT_FILES	Use this keyword to specify the flat files to be imported. Requires the FLAT_FILE_MODULES option to be specified.
FUNCTIONS	Use this keyword to specify the functions to be imported. Requires the module (for example, ORACLE_MODULES, BUSINESS_DEFINITION_MODULES, and so on) option to be specified.
GATEWAY_MODULES	Use this keyword to specify the Gateway modules to be imported. Requires the PROJECT option to be specified.
GENERIC_COMPONENTS	
GENERIC_FOLDERS	
GENERIC_MODULES	
ICONSETS	Use this keyword to specify the icon sets to be imported. Requires the ORACLE_MODULE option to be specified.

Table A-1 (Cont.) Object Types Keywords for Importing Objects

Object Type Keyword	Description
ITEM_CLASSES	Use this keyword to specify the item classes to be imported. Requires the BUSINESS_DEFINITION_MODULES option to be specified.
ITEM_FOLDERS	Use this keyword to specify the item folders to be imported. Requires the BUSINESS_DEFINITION_MODULES option to be specified.
LOCATIONS	Use this keyword to specify the locations to be imported. Requires the PROJECT option to be specified.
MAPPINGS	Use this keyword to specify the mappings to be imported. Requires the ORACLE_MODULES option to be specified.
MATERIALIZED_VIEWS	Use this keyword to specify the materialized views to be imported. Requires the ORACLE_MODULES option to be specified.
ORACLE_MODULES	Use this keyword to specify the Oracle modules to be imported. Requires the PROJECT option to be specified.
OBJECT_TYPES	Use this keyword to specify the object types to be imported. Requires the ORACLE_MODULES option to be specified.
PACKAGES	Use this keyword to specify the packages to be imported. Requires the ORACLE_MODULES option to be specified.
PLSQL_RECORD_TYPES	Use this keyword to specify the PL/SQL record types to be imported.
PLSQL_REF_CURSOR_TYPES	Use this keyword to specify the PL/SQL REF cursor types to be imported.
PLSQL_TABLE_TYPES	Use this keyword to specify the PL/SQL types to be imported. Requires the ORACLE_MODULES option to be specified.
PLUGGABLE_MAPPING_FOLDERS	Use this keyword to specify the pluggable mapping folders to be imported.
PLUGGABLE_MAPPINGS	Use this keyword to specify the pluggable mappings to be imported. Requires the PLUGGABLE_MAPPINGS_FOLDERS option to be specified.

Table A-1 (Cont.) Object Types Keywords for Importing Objects

Object Type Keyword	Description
PRESENTATION_TEMPLATES	Use this keyword to specify the presentation templates to be imported. Requires the BUSINESS_PRESENTATION_MODULES option to be specified.
PROCESS_FLOW_MODULES	Use this keyword to specify the process flow modules to be imported. Requires the PROJECT option to be specified.
PROCESS_FLOW_PACKAGES	Use this keyword to specify the process flow packages to be imported. Requires the PROCESS_FLOW_MODULES option to be specified.
PROCESS_FLOWS	Use this keyword to specify the process flows to be imported. Requires the PROCESS_FLOW_PACKAGES option to be specified.
PROFILE_PREFERENCES	
PROJECT	Wildcard format supported, but if used, no other object type keywords can follow. To import shared transformations, use PROJECT=PUBLIC_PROJECT.
QUERYOBJECTS	
QUEUE_TABLES	Use this keyword to specify the queue tables to be imported. Requires the ADVANCED_QUEUES option to be specified.
ROLES	Use this keyword to import the roles to be imported. Requires administrator privileges.
SAP_MODULES	Use this keyword to specify the SAP modules to be imported. Requires the PROJECT option to be specified.
SEQUENCES	Use this keyword to specify the sequences to be imported. Requires the module (for example, ORACLE_MODULES, SAP_MODULES, and so on) options to be specified.
SHARED_MODULES	Requires PROJECT=PUBLIC_PROJECT option to be specified.
SNAPSHOTS	Use this keyword to specify the snapshots to be imported. If this option is used, the no other object type keyword options can precede it.
SQLCOLLECTIONS	

Table A-1 (Cont.) Object Types Keywords for Importing Objects

Object Type Keyword	Description
TABLES	Use this keyword to specify the tables to be imported. Requires the module (for example, ORACLE_MODULES, SAP_MODULES, and so on) options to be specified.
TRANSPORTABLE_MODULE_TABLESPACES	Use this keyword to specify the transportable module tablespace to be imported.
TRANSPORTABLE_MODULES	Use this keyword to specify the transportable modules to be imported.
USERS	Use this keyword to import the users to be imported. Requires administrator privileges.
VIEWS	Use this keyword to specify the views to be imported. Requires the module (for example, ORACLE_MODULES, SAP_MODULES, and so on) options to be specified.

Examples of Control Files Used to Import Metadata

You can direct the MDL import utility to import objects from a file by creating a control file with a set of parameters. [Example A-2](#) shows a typical control file for importing objects from a MDL data file.

Example A-2 Control File format

```
PROJECT=WH_SALES
ORACLE_MODULES=SALES_SRC
EXTERNAL_TABLES=SALES_DATA_EXT
```

Control File Used to Create an Action Plan

You can specify an action plan in the control file that will allow you to specifically define what you want to do with each object in the imported file. First you need to indicate that the type of import is an action plan by specifying `MODE = ACTION PLAN`. Next, you need to specify the type of actions for objects that you want to import or skip. If you want to import the objects, you can set the Action to either `CREATE`, `UPDATE`, `REPLACE`, or `INCREMENTALUPDATE`. Otherwise, if you want to skip the object, specify `NONE` as the Action. For the list of object type keywords that you can use to import objects using the Action Plan, refer [Table A-1](#).

[Example A-3](#) shows an example of an MDL control file that contains an action plan.

Example A-3 MDL Action Plan

```
MODE=ACTIONPLAN
#
# User-Specified Action Plan
#
ACTION=NONE
ORACLE_MODULES=(DATAWAREHOUSE)
#
ACTION=CREATE
```

```
TABLES=(TABLE_3)
FACTS=(FACT1, FACT2, FACT3)
SEQUENCES=(SEQ_A, SEQ_B, SEQ_C)
#
ACTION=REPLACE
TABLES=(TABLE_1, TABLE_2)
DIMENSIONS=(DIM1, DIM2, DIM3)
#
# Switching to a different module
ACTION=REPLACE
FLAT_FILE_MODULES=(FLAT_FILE)
FILES=(FILE_1, FILE_2)
#
ACTION=CREATE
FILES=(FILE_3)
#
```

Exporting Metadata Using OMB Plus

You use the OMBEXPORT command to export metadata. To use control files in conjunction with the OMBEXPORT command, use the CONTROL_FILE clause. For more information on creating control files, see ["Creating MDL Control Files"](#) on page A-1.

Before you use the OMBEXPORT command, ensure that you are connected to the repository from which you want to export metadata. You use the OMBCONNECT command to connect to a repository. For more information on the OMBCONNECT and OMBEXPORT commands, refer to the OMBEXPORT command in the Oracle Warehouse Builder Scripting Reference.

Examples of Exporting Metadata Using Control Files

To use control files in conjunction with the OMBEXPORT command:

1. Create an MDL control file.
For more information on creating an MDL control file, see ["Creating MDL Control Files"](#) on page A-1.
2. Open OMB Plus by first selecting **Start**, then **Programs**, then **<OWB Home>**, then **Warehouse Builder**, and, finally, **OWB OMB Plus**.
3. Connect to the repository from which you want to export metadata.
4. Execute the OMBEXPORT command with the CONTROL_FILE clause that specifies the control file created in Step 1.

For example, to use a control file called `par.ctl` use the following OMBEXPORT command:

```
OMBEXPORT TO MDL_FILE 'd:/mdl/exp1.mdl' FROM PROJECT 'MY_PROJECT'
CONTROL_FILE 'd:/mdl/par.ctl' OUTPUT LOG TO 'd:/mdl/exp1.log'
```

Importing Metadata Using OMB Plus

You use the OMBIMPORT command to import metadata. You can also use control files in conjunction with the OMBIMPORT command. To do this, use the CONTROL_FILE clause of the OMBIMPORT command. For more information on creating control files, see ["Creating MDL Control Files"](#) on page A-1.

Before you use the OMBIMPORT command, ensure that you are connected to the repository into which you want to import metadata. You use the OMBCONNECT command to connect to a repository. For more information on the OMBCONNECT and OMBIMPORT commands, refer to Oracle Warehouse Builder Scripting Reference.

Examples of Importing Metadata Using Control Files

To use control files in conjunction with the OMBIMPORT command:

1. Create an MDL control file.
For more information on creating an MDL control file, see ["Creating MDL Control Files"](#) on page A-1.
2. Open OMB Plus by first selecting **Start**, then **Programs**, then **<OWB Home>**, then **Warehouse Builder**, and, finally, **OWB OMB Plus**
3. Connect to the repository from which you want to import metadata.
4. Execute the OMBIMPORT command with the CONTROL_FILE clause that specifies the control file created in Step 1.

For example, to use a control file called `par.ct1` use the following OMBIMPORT command:

```
OMBIMPORT FROM MDL_FILE 'd:/mdl/exp1.mdl'
CONTROL_FILE 'd:/mdl/par.ct1' OUTPUT LOG TO 'd:/mdl/exp1.log'
```

Accessing Transformation Modules Using OMBPlus

Transformation modules consist of a set of reusable transformations that you use to transform your source data. Transformations include functions, procedures, and packages.

There are two types of transformation modules:

- [Predefined Transformations](#)
- [Custom Transformations](#)

Predefined Transformations

Predefined transformations consist of built-in and seeded functions and procedures that are part of the Oracle Library. You can directly use these transformations in any project in your repository.

Predefined transformations are grouped into the following categories. Each category contains transformations that pertain to that category.

- Administration
- Character
- Control Center
- Conversion
- Date
- Numeric
- OLAP
- Other
- Spatial

- Streams
- SYS
- XML

Accessing Predefined Transformations Using OMBPlus

All predefined transformations belong to a transformation module called WB_PREDEFINED_TRANS in the project PUBLIC_PROJECT. Also, every project in the repository contains the WB_PREDEFINED_TRANS. To access predefined transformations, you must change the current context to the WB_PREDEFINED_TRANS transformation module either in PUBLIC_PROJECT or in your project.

Each category of predefined transformations is represented by a package in the WB_PREDEFINED_TRANS transformation module. The package contains the transformations, including functions and procedures, that are belong under it. For example, all the predefined numeric transformations belong to the package called NUMERIC under the WB_PREDEFINED_TRANS transformation module.

Examples

Use the following commands to list the types of public transformation modules.

```
OMB+> OMBCC '/PUBLIC_PROJECT/'
OMB+> OMBLIST TRANSFORMATION_MODULES
```

To view the types of predefined transformations, use the following command from the context of the WB_PREDEFINED_TRANS transformation module.

```
OMB+> OMBCC '/PUBLIC_PROJECT/WB_PREDEFINED_TRANS'
OMB+> OMBLIST PACKAGES
```

To view the procedures under the Date category of the predefined transformations, first change context to the DATE package.

```
OMB+> OMBCC '/PUBLIC_PROJECT/WB_PREDEFINED_TRANS/PACKAGES/DATE'
OMB+> OMBLIST PROCEDURES
```

Use the following command to use the Date transformation TRUNC in your mapping.

```
OMB+> OMBCREATE MAPPING 'MAP1' \
> ADD TRANSFORMATION OPERATOR 'TRUNC_OPER' \
> BOUND TO FUNCTION '/MY_PROJECT/WB_PREDEFINED_TRANS/DATE/TRUNC'
```

Custom Transformations

Custom transformations are transformations that are defined by the user. Custom transformations include functions, procedures, and packages.

Custom transformations are of two types:

- Public custom transformations
These are part of the global shared library that consists of predefined transformations.
- Custom transformations within a particular project
These are accessible only in the project in which they are defined.

Public Custom Transformations

Public custom transformations are accessible across all projects in your repository. They belong to the transformation module `WB_CUSTOM_TRANS` under the project `PUBLIC_PROJECT`. Also, every project in your repository contains a transformation module called `WB_CUSTOM_TRANS`. This transformation module contains the public custom transformations.

Examples

To list the types of custom transformations, you need to be in the context of the custom transformations module.

```
OMB+> OMBCC '/PUBLIC_PROJECT/WB_CUSTOM_TRANS'
OMB+> OMBLIST FUNCTIONS
```

To view the custom public procedures, use the following command from the context of the `WB_CUSTOM_TRANS` module in the `PUBLIC_PROJECT`.

```
OMB+> OMBLIST PACKAGES
```

To use a public custom function in a mapping, navigate to the context of the `WB_CUSTOM_TRANS` transformation module under the project in which you are defining the mapping.

```
OMB+> OMBCC '/MY_PROJECT/MOD1'
OMB+> OMBCREATE MAPPING 'MAP1'\
  > ADD TRANSFORMATION OPERATOR 'FUNC_OPER' \
  > BOUND TO FUNCTION '/PUBLIC_PROJECT/WB_CUSTOM_TRANS/FUNC1'
```

Custom Transformations that Belong to a Particular Project

You can create custom transformations whose scope is limited to the project in which they are defined. These custom transformations are defined in the context of a particular project and are accessible to all the modules within that project.

For example, the project `MY_PROJECT` contains two modules `MOD1` and `MOD2`. In `MOD1`, you define a function called `LOCAL_FUNC`. This function is accessible from the context of both `MOD1` and `MOD2`.

Custom transformations that belong to a particular project are part of the transformations in that project.

To create a custom transformation in the module `MOD1`, use the following syntax.

```
OMB+> OMBCREATE FUNCTION 'LOCAL_FUNC'\
  > ADD PARAMETER PARAM_1\
  > SET PROPERTIES (IN_OUT,DATATYPE) VALUES('in','varchar2')
```

Use the following command to reference the function `LOCAL_FUNC` in a mapping that you defined in module `MOD2`.

```
OMB+> OMBCREATE MAPPING 'MAP1'\
  > ADD TRANSFORMATION OPERATOR 'TRUNC_OPER' \
  > BOUND TO FUNCTION '/MY_PROJECT/MOD1/LOCAL_FUNC'
```

Working with Mappings and Operators

This section includes the following topics for using OMB Plus to add operators to mappings:

- [Defining Expressions in Mappings](#) on page A-12
- [Default Group Names and Attribute Names](#) on page A-12

Defining Expressions in Mappings

When using the `OMBCREATE MAPPING` and `OMBALTER MAPPING` commands, you can create and edit expressions such as filter, join, and group by expressions. If you define the attributes of the operator before defining the expression, OMB Plus generates the expected code. However, if you define the expression incorrectly and without first defining the necessary attributes, OMB Plus interprets your entry as a string literal and generates unexpected code.

To correctly define an expression in a mapping, prefix each attribute name with a colon. OMB Plus recognizes the text following a colon as an attribute. For example, OMB Plus interprets

```
:INOUTGRP1.ATTR1
```

as an attribute `ATTR1` in a group named `INOUTGRP1`.

If you make it a practice when writing expressions to precede attribute names with a colon, OMB Plus returns an error message in the event that you defined the expression without first defining the operator attributes.

Default Group Names and Attribute Names

When you use OMB Plus to add an operator to a mapping, Warehouse Builder adds the operator and assigns default groups and parameters. [Table A-2](#) lists the default groups and parameters for each operator.

Table A-2 *Default Names for Groups and Attributes*

Operator Type	Default Operator Name	Default Group Name	Default Attribute Name
TABLE	Same as bound object name	INOUTGRP1	Same as column names
VIEW			
EXTERNAL TABLE			
MATERIALIZED VIEW			
CUBE	Object name	Same as file record name	Same as field names in each record
DIMENSION			
FLAT FILE	Same as sequence name	OUTGRP1	NEXTVAL
SEQUENCE			CURRVAL
DATAGENERATOR	DATAGENERATOR	OUTGRP1	RECNUM
			SYS_DATE
			SEQUENCE
CONSTANT	CONST	OUTGRP1	No defaults
KEY LOOKUP	Object name	INGRP1	In LOOKUP_OUT object column names
		OUTGRP1	

Table A-2 (Cont.) Default Names for Groups and Attributes

Operator Type	Default Operator Name	Default Group Name	Default Attribute Name
SET	SET	INGRP1 INGRP2 OUTGRP1	None
JOINER	JOIN	INGRP1 INGRP2 OUTGRP1	None
SPLITTER	SPLIT	INGRP1 OUTGRP1 OUTGRP2 REMAINING_ ROWS	None
DEDuplicATOR	DEDUP	INOUTGRP1	None
AGGREGATOR	AGG	INGRP1 OUTGRP1	None
FILTER	FLTR	INOUTGRP1	None
SORTER	SORT	INOUTGRP1	None
NAME AND ADDRESS	NAMEADDR	INGRP1 OUTGRP1	Within Group INGRP1: Line 1, Line 2, Line 3 Within Group OUTGRP1: First Name, Last Name, Primary Address, Secondary Address, City, State, Postal Code, Is Good Group
PROCEDURES	Procedure name	If input parameter exists, an input group will be created with the name INGRP1. If output parameter exists, an output group will be created with the name OUTGRP1. If inout parameter exists, an input-output group will be created with the name INGRP1.	Same as parameter names
FUNCTIONS	Same as function name	INGRP1, RETURN	An attribute
TRIGGER	PREMAP POSTMAP	Naming see PROCEDURES and FUNCTIONS	None
INPUT_PARAMETER	MAP_INPUTS	MAP_INPUTS	None

Table A–2 (Cont.) Default Names for Groups and Attributes

Operator Type	Default Operator Name	Default Group Name	Default Attribute Name
OUTPUT_PARAMETER	MAP_OUTPUTS	MAP_OUTPUTS	None
EXTERNAL_PROCESS	EXTERNALPROCES S	None	None
EXPRESSION	EXPR	INGRP1 OUTGRP1	None

Accessing Transformation Modules

Transformation modules consist of a set of reusable transformations that you use to transform your source data. Transformations include functions, procedures, and packages.

There are two types of transformation modules:

- [Predefined Transformations](#)
- [Custom Transformations](#)

Predefined Transformations

Predefined transformations consist of built-in and seeded functions and procedures that are part of the Oracle Library. You can directly use these transformations in any project in your repository.

Predefined transformations are grouped into the following categories. Each category contains transformations that pertain to that category.

- Administration
- Character
- Control Center
- Conversion
- Date
- Numeric
- OLAP
- Other
- Spatial
- Streams
- SYS
- XML

Accessing Predefined Transformations Using OMBPlus

All predefined transformations belong to a transformation module called WB_PREDEFINED_TRANS in the project PUBLIC_PROJECT. Also, every project in the repository contains the WB_PREDEFINED_TRANS. To access predefined transformations, you must change the current context to the WB_PREDEFINED_TRANS transformation module either in PUBLIC_PROJECT or in your project.

Each category of predefined transformations is represented by a package in the WB_PREDEFINED_TRANS transformation module. The package contains the transformations, including functions and procedures, that are belong under it. For example, all the predefined numeric transformations belong to the package called NUMERIC under the WB_PREDEFINED_TRANS transformation module.

Examples

Use the following commands to list the types of public transformation modules.

```
OMB+> OMBCC '/PUBLIC_PROJECT/'
OMB+> OMBLIST TRANSFORMATION_MODULES
```

To view the types of predefined transformations, use the following command from the context of the WB_PREDEFINED_TRANS transformation module.

```
OMB+> OMBCC '/PUBLIC_PROJECT/WB_PREDEFINED_TRANS'
OMB+> OMBLIST PACKAGES
```

To view the procedures under the Date category of the predefined transformations, first change context to the DATE package.

```
OMB+> OMBCC '/PUBLIC_PROJECT/WB_PREDEFINED_TRANS/PACKAGES/DATE'
OMB+> OMBLIST PROCEDURES
```

Use the following command to use the Date transformation TRUNC in your mapping.

```
OMB+> OMBCREATE MAPPING 'MAP1' \
> ADD TRANSFORMATION OPERATOR 'TRUNC_OPER' \
> BOUND TO FUNCTION '/MY_PROJECT/WB_PREDEFINED_TRANS/DATE/TRUNC'
```

Custom Transformations

Custom transformations are transformations that are defined by the user. Custom transformations include functions, procedures, and packages.

Custom transformations are of two types:

- Public custom transformations
 - These are part of the global shared library that consists of predefined transformations.
- Custom transformations within a particular project
 - These are accessible only in the project in which they are defined.

Public Custom Transformations

Public custom transformations are accessible across all projects in your repository. They belong to the transformation module WB_CUSTOM_TRANS under the project PUBLIC_PROJECT. Also, every project in your repository contains a transformation module called WB_CUSTOM_TRANS. This transformation module contains the public custom transformations.

Examples

To list the types of custom transformations, you need to be in the context of the custom transformations module.

```
OMB+> OMBCC '/PUBLIC_PROJECT/WB_CUSTOM_TRANS'
OMB+> OMBLIST FUNCTIONS
```

To view the custom public procedures, use the following command from the context of the WB_CUSTOM_TRANS module in the PUBLIC_PROJECT.

```
OMB+> OMBLIST PACKAGES
```

To use a public custom function in a mapping, navigate to the context of the WB_CUSTOM_TRANS transformation module under the project in which you are defining the mapping.

```
OMB+> OMBCC '/MY_PROJECT/MOD1'
OMB+> OMBCREATE MAPPING 'MAP1'\
  > ADD TRANSFORMATION OPERATOR 'FUNC_OPER' \
  > BOUND TO FUNCTION '/PUBLIC_PROJECT/WB_CUSTOM_TRANS/FUNC1
```

Custom Transformations that Belong to a Particular Project

You can create custom transformations whose scope is limited to the project in which they are defined. These custom transformations are defined in the context of a particular project and are accessible to all the modules within that project.

For example, the project MY_PROJECT contains two modules MOD1 and MOD2. In MOD1, you define a function called LOCAL_FUNC. This function is accessible from the context of both MOD1 and MOD2.

Custom transformations that belong to a particular project are part of the transformations in that project.

To create a custom transformation in the module MOD1, use the following syntax.

```
OMB+> OMBCREATE FUNCTION 'LOCAL_FUNC'\
  > ADD PARAMETER PARAM_1\
  > SET PROPERTIES (IN_OUT,DATATYPE) VALUES('in','varchar2')
```

Use the following command to reference the function LOCAL_FUNC in a mapping that you defined in module MOD2.

```
OMB+> OMBCREATE MAPPING 'MAP1'\
  > ADD TRANSFORMATION OPERATOR 'TRUNC_OPER' \
  > BOUND TO FUNCTION '/MY_PROJECT/MOD1/LOCAL_FUNC'
```

Running OMB Plus in Oracle JDeveloper

This section outlines the steps to install and run OMB Plus in Oracle Database JDeveloper. For more information, see the Oracle Database JDeveloper documentation.

This section includes the following topics:

- [Installing OMB Plus in Oracle JDeveloper](#) on page A-17
- [Opening the Syntax Highlighting Editor in JDeveloper](#) on page A-17
- [Invoking Keyword Auto Completion](#) on page A-17
- [Invoking the OMBPlus Interpreter](#) on page A-17
- [Viewing the OMBPlus Console](#) on page A-18
- [Viewing Help Documentation](#) on page A-18

Installing OMB Plus in Oracle JDeveloper

Use the following steps to install OMB Plus in Oracle JDeveloper. You must install Oracle JDeveloper and Warehouse Builder client on the same machine.

To install OMB Plus in JDeveloper:

1. Drop `shiphome/owb/lib/int/OMBPlus_jdev.jar` in the JDeveloper installation directory/`lib/ext` directory.
2. Start JDeveloper.
3. From the Tools menu, select **Preferences**, **OMBPlus**, and then **OMBPlus Installation** to set the Warehouse Builder installation directory. For example, `c:\oracle\ora81`. Do not include the Warehouse Builder directory at the end of this path.
4. Restart JDeveloper.

Repeat steps 3 and 4 each time you change the Warehouse Builder installation directory.

Opening the Syntax Highlighting Editor in JDeveloper

To open the syntax highlighting editor:

1. From the File menu, select **New** and then **OMBPlus**, if you want to create a new OMBPlus script. To open an existing OMBPlus script, select **File**, and then **Open**.
2. From the Tools menu, select **Preferences**, then **Editor**, and then **Syntax Colors**.
3. Select the OMBPlus Category to configure the syntax highlight styles.

Invoking Keyword Auto Completion

Use any of the following methods to invoke keyword auto completion:

- Wait for a predefined number of seconds after typing the first few characters of the keyword.

You can configure the predefined number of seconds for invoking the auto completion by changing the Auto-Popup Delay for Completion Insight scale. From the Tools menu, select **Preferences**, then **Editor**, and then **Code Insight**.

- Press **Ctrl** and **Space** after typing the first few characters of the keyword.
- Type any number of the first few characters of the keyword.

Invoking the OMBPlus Interpreter

To invoke the OMBPlus Interpreter:

1. From the View menu, select **Log Window** to view the OMBPlus Log.
2. From the Run menu, select **Run OMBPlus** to run the current OMBPlus script.

You can also invoke Run OMBPlus by right-clicking a specific script node on the system navigator and selecting **Run OMBPlus**.

You can terminate the OMBPlus program by selecting **Run**, then **Terminate**, and then **OMBPlus**.

3. If you want to provide input to the OMBPlus script, select **Project**, then **Project Settings**, then **Runner**, and then **Options** to set Allow Program Input.

Viewing the OMBPlus Console

To open the OMBPlus Console:

1. Select OMBPlus Console to open the OMBPlus Console window.
2. From the Tools menu, select **Preferences**, then **OMBPlus**, and then **OMBPlus Console** to configure the display in the console window.

Viewing Help Documentation

Use any of the following methods to invoke the help documentation:

- Invoke the help document for a specific keyword by placing the cursor anywhere in the keyword and then pressing F2.
- Invoke the help document for a specific keyword by selecting the specific keyword and then pressing F2.
- Locate the general help document for OMBPlus scripting language by selecting **OMBPlus**, and then **OMBPlus Help Topics**.