

**Oracle® Health Sciences Clinical Development
Analytics**

Administrator's Guide

Release 3.1 for Standard Configuration

E48102-02

January 2015

Oracle® Health Sciences Clinical Development Analytics Administrator's Guide, Release 3.1 for Standard Configuration

E48102-02

Copyright © 2010, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Finding Information and Patches on My Oracle Support	vi
Finding Documentation on Oracle Technology Network	vii
Related Documents	viii
Conventions	ix
1 Maintaining the OBIEE Repository and Warehouse	
1.1 Maintaining the Oracle Health Sciences Clinical Development Analytics OBIEE Repository	1-1
1.1.1 Modifying the Repository	1-2
1.1.1.1 Details for Selected Modifications	1-2
1.1.2 Merging Changes into a New Oracle-supplied Repository	1-3
1.2 Maintaining the Oracle Health Sciences Clinical Development Analytics Data Warehouse	1-3
1.2.1 Extensions	1-3
1.2.2 Substitutions	1-5
1.2.3 Modifying Data Warehouse Tables	1-6
1.3 Derivations	1-7
2 Extract Transform Load Programs	
2.1 ETL Architecture	2-1
2.1.1 Adding a New Data Source	2-6
2.2 Executing the ETL Load Plans	2-7
2.2.1 Executing CDA - Complete Warehouse Load Plan	2-7
2.2.2 Executing CDA - Inform Automerge Load Plan	2-9
2.3 Customizing an ETL Load Plan	2-9
2.4 Creating an ETL Load Plan	2-9
2.5 Modifying an ETL Load Plan	2-10
2.6 Scheduling an ETL Load Plan	2-11
A Troubleshooting	
A.1 Sorting and Displaying of Null Values in Reports	A-1
A.2 Aborting a Load Plan	A-2

Preface

This guide provides information on the configuration of Oracle Health Sciences Clinical Development Analytics (OHSCDA).

This preface contains the following topics:

- [Audience](#) on page v
- [Documentation Accessibility](#) on page v
- [Finding Information and Patches on My Oracle Support](#) on page vi
- [Finding Documentation on Oracle Technology Network](#) on page vii
- [Related Documents](#) on page viii
- [Conventions](#) on page ix

Audience

This guide is intended for:

- Data Warehouse Administrators, ETL Developers and Operators
- System Administrators
- Deduplication Administrators and Data Stewards

This guide assumes that you have the following general skills:

- Knowledge of Oracle Business Intelligence Enterprise Edition Plus.
- Knowledge of Oracle Business Intelligence Data Warehouse Administration Console.
- Knowledge of Informatica PowerCenter.
- Familiarity with Oracle Clinical.
- Familiarity with Oracle's Siebel Clinical.
- Familiarity with Oracle Healthcare Master Person Index

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Finding Information and Patches on My Oracle Support

Your source for the latest information about Oracle Health Sciences Clinical Development Analytics is Oracle Support's self-service Web site, My Oracle Support (formerly MetaLink).

Before you install and use an Oracle software release, always visit the My Oracle Support Web site for the latest information, including alerts, release notes, documentation, and patches.

Creating a My Oracle Support Account

You must register at My Oracle Support to obtain a user name and password account before you can enter the Web site.

To register for My Oracle Support:

1. Open a Web browser to <http://support.oracle.com>.
2. Click the **Register here** link to create a My Oracle Support account. The registration page opens.
3. Follow the instructions on the registration page.

Signing In to My Oracle Support

To sign in to My Oracle Support:

1. Open a Web browser to <http://support.oracle.com>.
2. Click **Sign In**.
3. Enter your user name and password.
4. Click **Go** to open the My Oracle Support home page.

Searching for Knowledge Articles by ID Number or Text String

The fastest way to search for product documentation, release notes, and white papers is by the article ID number.

To search by the article ID number:

1. Sign in to My Oracle Support at <http://support.oracle.com>.
2. Locate the Search box in the upper right corner of the My Oracle Support page.
3. Click the sources icon to the left of the search box, and then select Article ID from the list.
4. Enter the article ID number in the text box.
5. Click the magnifying glass icon to the right of the search box (or press the Enter key) to execute your search.

The Knowledge page displays the results of your search. If the article is found, click the link to view the abstract, text, attachments, and related products.

In addition to searching by article ID, you can use the following My Oracle Support tools to browse and search the knowledge base:

- **Product Focus** — On the Knowledge page, you can drill into a product area through the Browse Knowledge menu on the left side of the page. In the Browse any Product, By Name field, type in part of the product name, and then select the product from the list. Alternatively, you can click the arrow icon to view the complete list of Oracle products and then select your product. This option lets you focus your browsing and searching on a specific product or set of products.
- **Refine Search** — Once you have results from a search, use the Refine Search options on the right side of the Knowledge page to narrow your search and make the results more relevant.
- **Advanced Search** — You can specify one or more search criteria, such as source, exact phrase, and related product, to find knowledge articles and documentation.

Finding Patches on My Oracle Support

Be sure to check My Oracle Support for the latest patches, if any, for your product. You can search for patches by patch ID or number, or by product or family.

To locate and download a patch:

1. Sign in to My Oracle Support at <http://support.oracle.com>.
2. Click the **Patches & Updates** tab.

The Patches & Updates page opens and displays the Patch Search region. You have the following options:

- In the Patch ID or Number is field, enter the primary bug number of the patch you want. This option is useful if you already know the patch number.
 - To find a patch by product name, release, and platform, click the Product or Family link to enter one or more search criteria.
3. Click **Search** to execute your query. The Patch Search Results page opens.
 4. Click the patch ID number. The system displays details about the patch. In addition, you can view the Read Me file before downloading the patch.
 5. Click **Download**. Follow the instructions on the screen to download, save, and install the patch files.

Finding Documentation on Oracle Technology Network

The Oracle Technology Network Web site contains links to all Oracle user and reference documentation. To find user documentation for Oracle products:

1. Go to the Oracle Technology Network at <http://www.oracle.com/technetwork/index.html> and log in.
2. Mouse over the Support tab, then click the **Documentation** hyperlink.
Alternatively, go to Oracle Documentation page at <http://www.oracle.com/technology/documentation/index.html>
3. Navigate to the product you need and click the link.
For example, scroll down to the Applications section and click Oracle Health Sciences Applications.
4. Click the link for the documentation you need.

Related Documents

For more information, see the following documents in the Oracle Business Intelligence Enterprise Edition, Oracle Clinical, Siebel Clinical, Oracle Health Sciences InForm, and Oracle Data Integrator documentation sets:

Oracle Business Intelligence Enterprise Edition Documentation

The Oracle Business Intelligence Suite Enterprise Edition (OBIEE) documentation set includes:

- *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Developer's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*

Oracle Clinical Documentation

The Oracle Clinical documentation set includes:

- *Oracle Clinical Administrator's Guide*
- *Oracle Clinical Getting Started*
- *Interfacing from Oracle Clinical*
- *Oracle Clinical Conducting a Study*
- *Oracle Clinical Creating a Study*
- *Oracle Clinical Installation Guide*

Siebel Clinical Documentation

The Siebel Clinical documentation set includes:

- *Siebel Data Model Reference for Industry Applications*
- *Siebel Life Sciences Guide*

Oracle Health Sciences InForm Documentation

The Oracle Health Sciences InForm documentation set includes:

- *Oracle Health Sciences InForm Upgrade and Migration Guide*
- *Oracle Health Sciences InForm Installation Guide*
- *Oracle Health Sciences InForm Study and Reporting Setup Guide*
- *Oracle Health Sciences InForm User Guide*

- *Oracle Health Sciences InForm Reporting and Analysis*
- *Oracle Health Sciences InForm Utilities Guide*
- *Oracle Health Sciences InForm Clinical Data API Guide*
- *Oracle Health Sciences InForm Reporting Database Schema*
- *Oracle Health Sciences InForm Secure Configuration Guide*
- *Oracle Health Sciences InForm Third Party Licenses and Notices*

Oracle Data Integrator (ODI) Documentation

The Oracle Data Integrator documentation set includes:

- *Release Notes for Oracle Data Integrator*
- *Understanding Oracle Data Integrator*
- *Administering Oracle Data Integrator*
- *Installing and Configuring Oracle Data Integrator*
- *Application Adapters Guide for Oracle Data Integrator*
- *Developing Knowledge Modules with Oracle Data Integrator*
- *Connectivity and Knowledge Modules Guide for Oracle Data Integrator*
- *Oracle Data Integrator Tool Reference*
- *Java API Reference for Oracle Data Integrator*
- *Oracle Data Integrator Online Help*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Maintaining the OBIEE Repository and Warehouse

This chapter contains the following topics:

- [Maintaining the Oracle Health Sciences Clinical Development Analytics OBIEE Repository](#) on page 1-1
- [Maintaining the Oracle Health Sciences Clinical Development Analytics Data Warehouse](#) on page 1-3
- [Derivations](#) on page 1-7

1.1 Maintaining the Oracle Health Sciences Clinical Development Analytics OBIEE Repository

Each release of Oracle Health Sciences Clinical Development Analytics (OHSCDA) contains an OBIEE Repository (RPD) file. The Repository is the data store for the Oracle BI Server. It maintains the mapping of the physical tables comprising the data warehouse to the Presentation Layer, which holds the columns and tables available for use in OBIEE Analyses. As shipped, the RPD corresponds to the OHSCDA data warehouse, and can be used without any modification.

Note: OHSCDA ships with more than one Repository (Oracle Data Integrator (ODI) and OBIEE have Repositories). This section, however, is concerned only with the OBIEE Repository that ships with OHSCDA. Therefore, throughout this section, the terms Repository and RPD should be taken to mean the OBIEE Repository shipped with OHSCDA.

However, you might find it desirable to modify the Oracle-supplied OHSCDA Repository file (RPD), for any of the following reasons:

- You want to add a column or table to the data warehouse, and propagate that addition into the layers of the Repository.
- You want to add a calculated column in the Presentation Layer as a function of some set of Physical layer columns.
- You want to modify a Repository variable value, or add a new repository variable, for use in some Presentation Catalog calculation. For instance, you may want to modify the frequency with which the value of the dynamic repository variable `CURRENT_DAY` is refreshed. For more information about why OHSCDA must

refresh this variable, refer to the Note in step 13 of the Executing the ETL Programs section.

This section describes the procedures you must follow to carry out these types of modifications.

You should be aware that, once you have modified the Oracle-supplied Repository, it is your responsibility to merge these modifications into Repositories supplied by Oracle in subsequent patches and releases of OHSCDA. Details on how to re-apply your modifications are provided below.

Caution: Changes to the Repository should be made with care.

Privileges to make changes in the Repository should be granted only to a limited set of users who need to make such changes and also know how to make them correctly.

Changes should be tested on a side copy of the Repository before being released for production use.

1.1.1 Modifying the Repository

In OHSCDA Standard configuration, maintaining versions of the Repository is the responsibility of the administrator. Oracle recommends that you:

- Make a backup copy of each state of the repository before you modify it
- Attempt all modifications on a side copy of the Repository before putting it into production use
- Verify all changes to the Repository by running requests from the Presentation Service against the modified Repository

Therefore, Oracle recommends that you do not modify the deployed OHSCDA Repository directly.

If you need to modify the Repository, use the OBIEE Administration Tool.

1.1.1.1 Details for Selected Modifications

This section contains details on how to perform certain modifications to the RPD.

To modify the frequency with which CURRENT_DAY is refreshed:

1. In Oracle BI Administration Tool, click **Manage > Variables**.
2. Expand Repository and click **Initialization Block > ETL_Refresh_Ranges**.
3. In the Repository Variable Init Block - ETL_Refresh_Ranges screen, modify the value of **Refresh interval**.

Refresh interval indicates how often you want to refresh the value of CURRENT_DAY dynamic repository variable. By default, this value is set to 5 minutes. That is, the CURRENT_DAY dynamic repository variable is refreshed every five minutes. Modify Refresh interval to a suitable value.

See Also:

- *Oracle® Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1* for more information about modifying the RPD.

1.1.2 Merging Changes into a New Oracle-supplied Repository

Each OHSCDA release, and some patches, includes a copy of the OHSCDA Repository. If you do modify your copy of the OHSCDA Repository, you must merge your changes into the Oracle-supplied Repository each time you receive a release or patch of OHSCDA that includes a repository. At upgrade time, use the OBIEE Merge Repository Wizard (accessed through File > Merge) in the Repository Administration Tool to merge your modified RPD with the Oracle-supplied RPD.

For information on comparing and merging Repositories, refer to Chapter 15, *Managing Oracle BI Repository Files*, in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1*.

1.2 Maintaining the Oracle Health Sciences Clinical Development Analytics Data Warehouse

You may need to modify the OHSCDA data warehouse, typically for one of the following reasons:

- *Derivation*: Calculation of a new measure as a function of some supplied measures.
- *Extension*: Adding data that was not delivered with OHSCDA.
- *Substitution*: Swapping data from a different source for a column that was delivered with OHSCDA.

Caution: Exercise caution when you modify the data warehouse. Please conform to the recommendations mentioned in [Section 1.2.3, "Modifying Data Warehouse Tables"](#) on page 1-6.

1.2.1 Extensions

An *extension* is a new column added to the data warehouse for data not extracted from the transactional sources by the SDE supplied with OHSCDA.

Example: Adding the study manager's name as an attribute of the study dimension for each study. The following are the assumptions:

- This information is available in a non-Oracle Clinical database, in a table named STUDY_MANAGERS. The column in that table holding the Study Manager name is STUDY_MANAGER_NAME.
- This table has a foreign key to the primary key in Oracle Clinical table OCL_STUDIES.

Note: When you add extension columns to the OHSCDA data model, the column name must start with "X_". This avoids collisions with names of columns added by Oracle in later releases of OHSCDA.

You can add the column either to the target table, or to an extension table. To minimize the level of effort required when implementing a release with a new repository, Oracle recommends that you add extensions to the warehouse through user-defined extension tables, rather than by adding new columns directly into the relevant staging and target tables.

Note: For each user-defined extension tables, its error table E\$_<user-defined extension tables name> must be created. Also, provide the select, insert, update, and delete grants to the ODI warehouse work schema.

Perform the following tasks to add the study manager name to the study dimension for each study:

Important: Create a backup of the Repository and Views before modifying it. Retain a log of changes you make to the Repository to ensure that the Merge Wizard correctly reapplies them to the next repository that Oracle ships.

1. Modify staging table W_RXI_STUDY_DS, adding the X_STUDY_MANAGER_NAME column. To modify the staging table, perform the following tasks in ODI:
 - a. Copy the warehouse datastore W_RXI_STUDY_DS into another model.
This is to ensure that the changes are not overwritten in the next OHSCDA upgrade.
 - b. Add a new column X_STUDY_MANAGER_NAME and save it.
 - c. Navigate to the **Design > Tables** subtab and add the new column X_STUDY_MANAGER_NAME under W_RXI_STUDY_DS and save it.
2. Modify the SDE that populates W_RXI_STUDY_DS, as indicated in the following steps:
 - a. Copy the SDE into another folder (to ensure that the changes are not overwritten in the next OHSCDA upgrade) or use an existing customized SDE.
 - b. Duplicate the SRO view (VW_SDE_STUDY_DIM), used for this SDE as VW_SDE_STUDY_DIM_X.
 - c. Add the STUDIES_MANAGERS table in the SQL to select STUDY_MANAGER_NAME.
 - d. Grant select on SRO view (VW_SDE_STUDY_DIM_X) to the ODI source work schema.
 - e. Reverse-engineer the new customized SRO schema view into another model.
 - f. Open the copied SDE mapping in ODI.
 - g. Click the **Logical** tab.
 - h. Select the source to view its properties.
 - i. Under the **General** section, change the Datastore to point to the newly created customized SRO view (VW_SDE_STUDY_DIM_X).
 - j. Select the mapping target to view its properties.
 - k. Under the **General** section, change the Datastore to point to the customized stage table.
 - l. Map the new column STUDY_MANAGER_NAME from source to target.
 - m. Validate and save the mapping.

- n. Generate scenario and schedule it in the load plan.
3. If `W_RXI_STUDY_D` does not already exist to support other extension, create the extension table `W_RXI_STUDY_DX`, containing one column `[STUDY_WID]` to function as a foreign key that joins to the primary key in `W_RXI_STUDY_D`. This table is populated with one row for each row in `W_RXI_STUDY_D` when the Study SIL executes.
4. Add the extension table `W_RXI_STUDY_DX` in ODI. To add a table, perform the following task in ODI:
 - a. Reverse-engineer `W_RXI_STUDY_DX` into a new model (to ensure that the changes are not overwritten in the next OHSCDA upgrade) or use an existing customized table definition, if already exists.
5. Modify the SIL view and mapping that populate `W_RXI_STUDY_D`.
6. Add instructions to create a record in `W_RXI_STUDY_DX` for each record in `W_RXI_STUDY_D` and to copy `W_RXI_STUDY_DS.STUDY_MANAGER_NAME` into `W_RXI_STUDY_DX.STUDY_MANAGER_NAME` for each record.
7. Generate scenario and schedule it in the load plan.
8. Using `W_RXI_DISCREPANCY_FX` as an example, propagate the extension table and its content to the Business and Presentation layers.

1.2.2 Substitutions

A substitution occurs if you have a preferred alternative source of data for a column that OHSCDA populates from Oracle Clinical or Siebel Clinical. For example, you have a system for defining what data collection instruments (DCIs) are mandatory for a given study, subject, or subject visit, and you prefer that over the OHSCDA calculation that is based on expected data collection modules (DCMs) and subject visit schedules. In this case, your column will be present in a table, and the SDE that extracts the data to a staging table already exists. Perform the following tasks:

1. Create a view that joins the Oracle Clinical table and the table containing your mandatory DCI information, in which your values replace the Oracle-supplied values for the column of interest. Call this the Substitution view. Also, provide the select grant on Substitution view to the source work schema.
2. Modify the SDE to read from the Substitution view, rather than the Oracle-supplied view.

To modify the SDE, perform the following tasks in ODI:

- a. Copy the SDE into another folder (to ensure that the changes are not overwritten in the next OHSCDA upgrade) or use an existing customized SDE.
- b. Reverse-engineer the Substitution view into another model.
- c. Select the source to view its properties.
- d. Under the **General** section, change the Datastore so that the mapping reads from the Substitution view.
3. Validate and save the mapping.

If you make changes to a source table, you must propagate that change forward as far as necessary. Some of the scenarios and the related necessary adjustments are described in the [Table 1-1](#):

Table 1–1 Scenarios Requiring Necessary Adjustments

Scenario	Adjustments Required
New table has the same layout as the old table, but is passed through from a different source	Change the SDE view that reads the old table to instead read the new table.
Modified table has modified layout	<ol style="list-style-type: none"> 1. Modify the SDE view to read the modified layout. 2. Modify the staging table populated by the SDE to include the modified layout. 3. Modify the SIL view and mapping to read the modified layout. 4. Modify the target table to include the modified layout. 5. Modify the RPD to accept the changed data warehouse table.
New table	<ol style="list-style-type: none"> 1. Add a new staging table and its error table to accept the new input. 2. Add a new SDE view in the SRO schema. 3. Add a new SDE mapping to read from the new view and write to the new staging table. 4. Add a new warehouse table and its error table to make the new data available to the BI Server. 5. Add a new SIL view in the warehouse. 6. Add an SIL mapping to populate the new data warehouse table from the new view. 7. Modify the RPD to accept the new warehouse table.

1.2.3 Modifying Data Warehouse Tables

Depending on what changes are required to the data warehouse, it is necessary to modify SRO and warehouse views either in SRO and/or warehouse schema and in ODI, or in the source, staging, and target tables.

Managing Indexes

OHSCDA is delivered with a set of indexes. You can create additional indexes to meet your query requirements.

Note: Oracle recommends that you Drop and re-create all indexes only for full load. This should be disabled for incremental load.

It is useful to drop all indexes on the warehouse tables before loading large volumes of data, and to recreate them afterward. ODI can automate this process for you. To drop indexes before a load, and recreate them afterward, perform the following tasks in ODI:

1. Execute the **CDA-Complete Warehouse** load plan.
2. Set the variable `VAR_OCDA_DWH_LOAD_FLAG` value as 1 while running the load plan.

Note: Setting the variable VAR_OCDA_DWH_LOAD_FLAG value as 1, while running load plan, executes for full load, truncates all warehouse tables, drops indexes, loads data, and recreates indexes.

1.3 Derivations

A *derivation* is a calculation of a new measure as a function of some supplied measures. You can use any of the following approaches to calculate derivations:

- Calculate the derivation as part of the creation of an analysis.

In this approach, only the Presentation Catalog is modified. However, you must specify the calculation for each analysis, and the calculation is executed every time the analysis is executed.

- Calculate the derivation in the physical or business layer of the RPD and propagate it to the presentation layer. This makes the derivation you created appear in Answers as a column.

Using this approach, you can specify the calculation once and use it for multiple analyses. Once it has been propagated to the Presentation Layer, the derived value looks the same as any other Answers column to an end user.

- Calculate the derivation in the data warehouse.

In this approach, you add an extension column to the warehouse to hold the derived values. To do so:

1. Add the result column to the staging and target table of the fact or dimension in which the derivation is being created. It is an extension column, so its name must begin with "X_".
2. Modify the SDE view and mapping to calculate the derived value during extract of records from the sources.
3. Modify the SIL view and mapping to transfer the derived value from the Staging to the Target table.
4. Create and execute a script to provide values for the derived column in rows already present in the Target.
5. Add the column to all layers of the RPD.

In carrying out these steps, follow the rules described above for changes to the RPD and warehouse.

Extract Transform Load Programs

This chapter contains the following topics:

- [ETL Architecture](#) on page 2-1
- [Executing the ETL Load Plans](#) on page 2-7
- [Customizing an ETL Load Plan](#) on page 2-9
- [Creating an ETL Load Plan](#) on page 2-9
- [Modifying an ETL Load Plan](#) on page 2-10
- [Scheduling an ETL Load Plan](#) on page 2-11

To load data from the source systems to the data warehouse, OHSCDA uses Extract Transform and Load (ETL) programs that

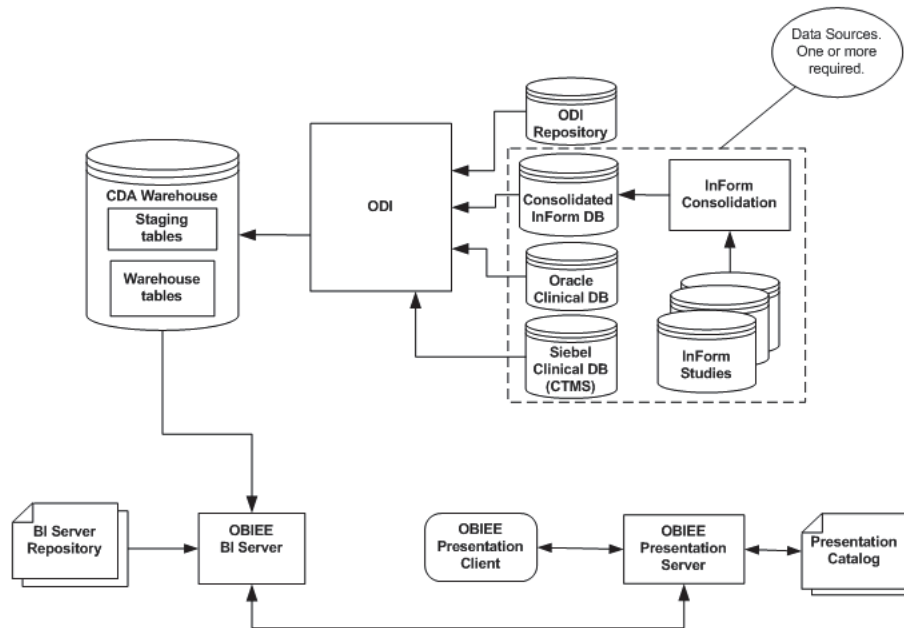
- Identify and read desired data from different data source systems,
- Clean and format data uniformly, and
- Write it to the target data warehouse.

In OHSCDA, Oracle Clinical, Siebel Clinical, and InForm are the source systems for which Oracle provides predefined ETL.

2.1 ETL Architecture

ETL is the process by which data is copied from a source database and placed in warehouse tables, available for analytic queries.

Figure 2–1 ETL Architecture

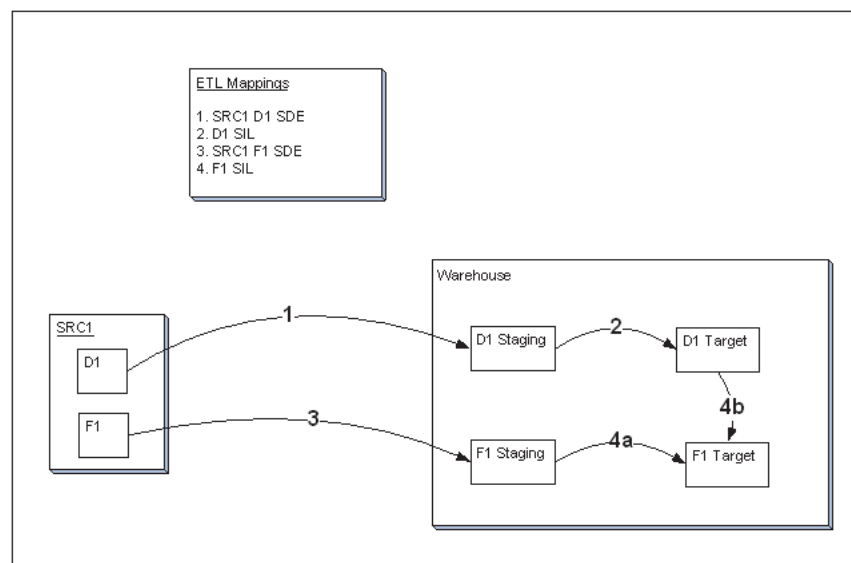


OHSCDA supports the extraction of data from one or more transactional application source databases. In OHSCDA, ETL is performed by execution of ODI Mappings. A mapping is a program that selects data from one or more tables, performs transformations and derivations on the data, and inserts or updates the results in a target table. Mappings are stored in the ODI Repository. Oracle uses ODI to schedule and supervise the execution of the entire set of ETL necessary to populate the OHSCDA warehouse.

Because data can be extracted from databases managed by different applications, the ETL mappings to load any given warehouse table are composed of two parts. The first part is a mapping that reads from a specific application's database and writes to a common warehouse Staging table. This mapping is referred to as SDE, for Source-Dependent Extract. Records in the Staging table have a common format, regardless of the source from which they were extracted. For a given warehouse table, there must be one SDE for each supported source application. The second mapping for the given warehouse table reads from its Staging table and writes to the final warehouse table. This part is called as the SIL, for Source-Independent Load.

Figure 2–2 shows the ETL process for extracting data from two tables (one dimension D1 and one fact F1) in one source application database.

Figure 2–2 ETL for One Source Database



Each numbered arrow represents the execution of a Mapping. The Mappings are executed in the order indicated by the numbers. Mapping 4, the SIL that loads fact F1 into the warehouse target table for the fact, has two components. The first part (4a) represents the copy, transformation, and derivation of the staged data for the fact. The second part (4b) represents the setting of the values of each record's foreign key to dimension D1. The result is that dimension D1 can be queried as one of the dimensions in the star-schema centered on fact F1.

OHSCDA provides SIL for each of its warehouse tables. It provides SDE to stage data for these tables from InForm, Oracle Clinical, and Siebel Clinical database. The execution of the mappings is under the control of ODI.

The OHSCDA ETL architecture is flexible. It allows you to integrate data from multiple databases instances for one application (for example, multiple Oracle Clinical databases), or databases from different source applications (for example, one Oracle Clinical and one Siebel Clinical), or any combination of these.

The OHSCDA ETL architecture is extensible. OHSCDA provides SDE mappings for InForm, Oracle Clinical, and Siebel Clinical. You can add data from another application to the OHSCDA warehouse by creating the SDE to map its database tables to the OHSCDA Staging tables.

For each source application there is one SDE mapping for each warehouse table. This SDE extracts data from the source system and loads it to the staging tables. SDEs have the following feature:

- **Incremental submission mode:** OHSCDA-supplied ETL uses timestamps and journal tables in the source transactional system to optimize periodic loads.

There is one SIL mapping for each warehouse target table. The SIL extracts the normalized data from the staging table and inserts it into the data warehouse star-schema target table. SILs have the following attributes:

- Concerning changes to dimension values over time, OHSCDA overwrites old values with new ones. This strategy is termed as *Slowly Changing Dimension approach 1*.

- OHSCDA's data model includes aggregate tables and a number of indexes, designed to minimize query time.
- The results of each ETL execution are logged by ODI. The logs hold information about errors encountered, during execution.

Every table in OHSCDA warehouse that is populated through ODI has a corresponding error log table.

The naming convention for error log table is as follows:

`E$_<TABLE_NAME>`

If the length of the table name is greater than 26, a different naming convention is used for error log table. The characters until the second instance of underscore (__) in the table name are replaced with E\$. The naming convention in this case will be as follows:

`E$_<remaining characters of table name>`

For example, if the table name is `W_HS_MDM_APPLICATION_USER_DS`, the characters `W_HS` are replaced with `E$`. Hence, the name of the error log table will be `E$_MDM_APPLICATION_USER_DS`.

During ETL execution, records which fail to be inserted in the target table (for example, some records violate a constraint) are placed in the corresponding error tables. You can review which records did not make it into the data warehouse, and decide on appropriate action with respect to them.

Adding Data Source Information

As you read data from different database instances, you need to specify the source of the data. OHSCDA provides the `OCDA_CONTROL_S`, `OCDA_CONTROL_LOG_S`, and `W_RXI_DATASOURCE_S` tables (in source-specific SRO schema) that stores information about data sources from which data is extracted for OHSCDA. These tables are populated by procedure trigger or seed data.

- **OCDA_CONTROL_S:** This table is present in the OHSCDA SRO schema. It has only one row per Enterprise ID present in the OHSCDA application. Initial data is populated on full load and subsequent ETL Incremental Runs update this table accordingly for the specific enterprises for which it is run, thereby tracking the last extract date appropriately. The following are some of the columns in these table:
 - **DATASOURCE_NUM_ID:** It is used to identify the data source.
 - **SOURCE_EXTRACTION_START_DT:** It stores the ETL start date.
 - **SOURCE_EXTRACTION_END_DT:** It stores the ETL end date.
 - **PROGRAM_NAME:** It stores information about load plan name.
 - **PROGRAM_RUN_DT:** It stores date on which the last ETL was successfully executed. This value is populated when ETL is executed successfully. Else, it is NULL.
 - **CONFIG_DAYS:** It stores value for the PRUNE days. By default, its value is 1. You can modify the value of this column. End date for incremental load is derived from this column as `SYSDATE - CONFIG_DAYS`.
 - **ETL_RUN_ID:** It is generated using a sequence.
- **OCDA_CONTROL_LOG_S:** Before every ETL run the information in the table `OCDA_CONTROL_S` will be copied to `OCDA_CONTROL_LOG_S` for auditing via a trigger.

- **W_RXI_DATASOURCE_S:** It is used for InForm source system to get DATASOURCE_NUM_ID and DATASOURCE_NAME. The following are some of the columns in these table:
 - **ROW_WID:** A unique ID for each record in the table.
 - **DATASOURCE_NUM_ID:** The ID for the database.
 - **DATASOURCE_NAME:** A meaningful name of the database.
 - **DATASOURCE_TYPE:** Application system that manages the database.
 - **DESC_TEXT:** Optional text describing the purpose of the database.
 - **INTEGRATION_ID:** Set this to the same values as DATASOURCE_NUM_ID
 - **INFORM_STUDY_GUID:** Study GUID of the InForm source.

See Also:

- *Oracle Health Sciences Clinical Development Analytics Electronic Technical Reference Manual*, for more information about these tables.
- [Section 2.1.1, "Adding a New Data Source"](#) on page 2-6, for more information about how to add a new data source to OHSCDA.

Handling Deletions in Siebel Clinical

OHSCDA provides an optional feature to manage hard deletion of records in Siebel Clinical. You create triggers in the source system to handle deletion of records. To do this:

1. Navigate to the temporary staging location where the OHSCDA installer copies the installation files.
2. Connect to the Siebel Clinical data source and run the `ocda_sc_del_trigger.sql` script delivered with OHSCDA. This script creates the triggers on tables provided as input. The following are the tables in Siebel Clinical for which OHSCDA supports creating triggers:
 - S_CL_PTCL_LS
 - S_PROD_INT
 - S_CL_SUBJ_LS
 - S_CONTACT
 - S_CL_PGM_LS
 - S_PTCL_SITE_LS
 - S_EVT_ACT
 - S_ORG_EXT

Provide a list of comma separated values of table names for which the triggers need to be created as the script's input. For example, S_CL_PTCL_LS, S_PROD_INT, and S_CL_SUBJ_LS. The tables names that you provide can only be a subset of the tables listed above.

Note: When you delete a record in the table, the primary key of the deleted record is inserted in the RXI_DELETE_LOG_S table on the Siebel source system.

3. Set the value of the **OHSCDA.DELETE_FLOW** variable to **Y** while executing the load plan in ODI.
4. Set the value of the **DELETE_FLOW** variable to **Y** while executing the load plan in ODI.
5. Execute the ETLs as listed in the [Executing the ETL Load Plans](#) section.

The Siebel Clinical related SDE mappings read the above instance of the **RXI_DELETE_LOG_S** table.

Note: Records that are deleted in the source system are soft deleted in the data warehouse.

2.1.1 Adding a New Data Source

OHSCDA provides predefined source-dependent extract (SDE) mappings for InForm, Oracle Clinical, and Siebel Clinical. It also provides **OCDA_CONTROL_S**, **OCDA_CONTROL_LOG_S**, and **W_RXI_DATASOURCE_S** tables, and procedure/trigger to populate these tables for each SRO schema.

For InForm source, enter your sources' system-related information in **W_RXI_DATASOURCE_S** table in [Section , "Adding Data Source Information"](#) on page 2-4 where structure of **w_rxi_datasource_s** table is described. For other source systems, update the procedure which populates **OCDA_CONTROL_S** with unique **DATASOURCE_NUM_ID** (set this value to a number greater than 100).

Perform the following tasks to add a new data source:

1. Install OHSCDA in new source system (InForm, Oracle Clinical, or Siebel Clinical). For information, see *Oracle Health Sciences Clinical Development Analytics Installation and Configuration Guide*.

This creates SRO, work schema, and related objects for the selected source system.

2. In ODI Topology, create a new physical connection pointing to the new source system.
3. Edit the context and change Physical Schemas to point to the new physical connection.

For example, if the new source system is Oracle Clinical, edit the context **CTX_OCDA_OC** and change **LS_OCDA_OracleClinical** to point to the new physical schema.

Important:

- When calling an SDE mapping to read from a particular database instance, ensure that its corresponding SRO view has value of **DATASOURCE_NUM_ID** that corresponds to that database and **ENTERPRISE_ID** (normally 0).
 - If you write new SDE, ensure that it sets the value of **DATASOURCE_NUM_ID** in the staging table to which it writes.
-
-

4. Make a copy of **OCDA_W_RXI_LOV_S_seed.sql**.

5. In the copy, modify the value of `datasource_num_id` to the value that you specified in the procedure that populates the `OCDA_CONTROL_S` table for the source database that you are adding.
6. Connect to the SRO account of the source.
7. Truncate `W_RXI_LOV_S`.
8. Run the modified copy of `OCDA_W_RXI_LOV_S_seed.sql`.

2.2 Executing the ETL Load Plans

OHSCDA ships the following two load plans:

- **CDA - Complete Warehouse:** This is used to load warehouse with one or all of the three source systems (Oracle Clinical, Siebel Clinical, and InForm). For information on how to execute the CDA - Complete Warehouse load plan, see [Section 2.2.1, "Executing CDA - Complete Warehouse Load Plan"](#).
- **CDA- Inform Automerger:** This is used to load warehouse only from Inform data source and only when automerger feature is needed. For information on how to execute the CDA- Inform Automerger load plan, see [Section 2.2.2, "Executing CDA - Inform Automerger Load Plan"](#).

2.2.1 Executing CDA - Complete Warehouse Load Plan

To load data from the source to their target tables in the data warehouse, run the CDA - Complete Warehouse load plan packaged with OHSCDA. Perform the following tasks in ODI:

1. Click the **Operator** tab.
2. Select **Load Plans and Scenarios**.
3. Right-click the **CDA - Complete Warehouse** load plan and select **Run**.

The Start Load Plan dialog box appears.

Note: ■ The `VAR_INT_CONFIG_DAYS` variable is used to determine the extraction end date for the incremental load. For full load, set the value of this variable to 1. For incremental load, set the value of this variable to 0. This indicates that the source extraction end date is a day less than the ETL program run date. For example, if the ETL program run date is 28 July, 2010 the source extraction end date is 27 July, 2010.

- The `VAR_OCDA_DWH_LOAD_FLAG` variable is used to determine full load or incremental load. For Incremental load, set the value of this variable to 0. For full load, set the value of this variable to 1.
 - Setting `VAR_OCDA_DWH_LOAD_FLAG` value to 1 while executing load plan truncates the complete warehouse drop indexes on the DWH table, loads data, and creates indexes.
-

4. Select `CTX_OCDA` from the **Context** drop-down list.
5. Select `OCDA_LA` from the **Logical Agent** drop-down list.

6. Select <Use Session Task Log Level> from the **Log Level** drop-down list.
7. Assign startup values for the Load Plan Variables.
 - **OHSCDA.AUTO_MERGE_FLG**: Set value as **N**.
 - **OHSCDA.DELETE_FLOW**: Set value as **Y** to enable the Siebel Clinical deletion flow.
 - **OHSCDA.VAR_IF**: If InForm is your *one of the or only* data source(s), set value as **Y** to run load plan for the InForm source.
 - **OHSCDA.VAR_INT_CONFIG_DAYS**: Set numeric value (this variable is used to determine the extraction end date for the incremental load).
 - **OHSCDA.VAR_OC**: If Oracle Clinical is your *one of the or only* data source(s), set value as **Y** to run load plan for the Oracle Clinical source.
 - **OHSCDA.VAR_OCDA_DWH_LOAD_FLAG**: For full load, set value as **1** and for incremental load set value as **0**.
 - **OHSCDA.VAR_OCDA_ERROR_REJECT_LIMIT**: This variable is used to set the number of rows that will be tracked in the respective error tables prior to aborting the ETL in case of errors.
 You must enter positive integer for this variable. For example, 0, 10, 100, 1000, and so on.
 - **OHSCDA.VAR_SC**: If Siebel Clinical is your *one of the or only* data source(s), set value as **Y** to run load plan for the Siebel Clinical source.
8. Click **OK**.

Note: Execution of the ETL (specifically the `OCDA_ETL_RUN_S_POP` program) populates `W_ETL_RUN_S.LOAD_DT` with the timestamp for the execution of the ETL. This ETL execution timestamp is used in the calculation of OHSCDA measures concerning the amount of time that currently open discrepancies have been open.

While the timestamp is captured in `CURRENT_ETL_LOAD_DT`, it is only available for calculation of discrepancy intervals through the OBIEE Dynamic Repository Variable `CURRENT_DAY`. `CURRENT_DAY` is refreshed from `LOAD_DT` at a fixed interval, by default 5 minutes, starting each time the Oracle BI Service is started. Between the time that the ETL is run, and the time that `CURRENT_DAY` is refreshed, calculations of intervals that currently open discrepancies have been open will be inaccurate.

The following are two remedies:

- Restart the Oracle BI Server after every execution of the ETL. This will cause `CURRENT_DAY` to be refreshed to the correct value.
 - If this is inconvenient, you can modify the intervals between refreshes of the value of `CURRENT_DAY`. For more information on how to modify the refresh interval for `CURRENT_DAY`, see [Maintaining the Oracle Health Sciences Clinical Development Analytics OBIEE Repository](#) on page 1-1.
-

Tip: You can schedule the jobs to execute at regular intervals. For more information on scheduling jobs, refer to [Scheduling an ETL Load Plan](#) on page 2-11.

2.2.2 Executing CDA - Inform Automerge Load Plan

To load data from the source to their target tables in the data warehouse, run the CDA - Inform Automerge load plan packaged with OHSCDA. Perform the following tasks in ODI:

1. Click the **Operator** tab.
2. Select **Load Plans and Scenarios**.
3. Right-click the **CDA - Inform Automerge** load plan and select **Run**.
The Start Load Plan dialog box appears.
4. Select **CTX_OCDA** from the **Context** drop-down list.
5. Select **OCDA_LA** from the **Logical Agent** drop-down list.
6. Select **<Use Session Task Log Level>** from the **Log Level** drop-down list.
7. Assign startup values for the Load Plan Variables.
 - **OHSCDA.AUTO_MERGE_FLG:** Set value as **Y**.
 - **OHSCDA.VAR_INT_CONFIG_DAYS:** Set numeric value (this variable is used to determine the extraction end date for the incremental load).
 - **OHSCDA.VAR_OCDA_DWH_LOAD_FLAG:** For full load, set value as **1** and for incremental load set value as **0**.
 - **OHSCDA.VAR_OCDA_ERROR_REJECT_LIMIT:** This variable is used to set the number of rows that will be tracked in the respective error tables prior to aborting the ETL in case of errors.

You must enter positive integer for this variable. For example, 0, 10, 100, 1000, and so on.
8. Click **OK**.

2.3 Customizing an ETL Load Plan

When you customize an ETL Load Plan, it is your responsibility to maintain version control over changes to ETL mappings.

Oracle recommends that you carefully track the changes you make to Oracle-supplied ETL so that you can re-apply these changes in subsequent releases.

2.4 Creating an ETL Load Plan

Though OHSCDA includes ETL Load Plans for extracting data from InForm, Oracle Clinical, and Siebel Clinical to OHSCDA data warehouse, you may want to create your own ETL to extract data from other data sources.

Note: The value of `DATASOURCE_NUM_ID` is set to 1 for Oracle Clinical, 2 for Siebel Clinical, and 3 for InForm. If you want to add your own data sources, set this value to a number greater than 100.

See Also:

- *Oracle Data Integrator Online Help*

2.5 Modifying an ETL Load Plan

You may also want to modify an existing ETL to meet your reporting requirements.

See Also:

- *Oracle Data Integrator Online Help*

To modify an ETL without any changes to the associated tables or columns, perform the following tasks:

1. Identify the load plan that needs to be modified in ODI.
2. Open and modify the ETLs.
3. Test and save the changes in repository.
4. Re-generate the scenario.
5. Navigate to the load plan.
6. Regenerate and execute ETL to verify the changes.

Note: The ETL Load Plans that extract data for the warehouse fact tables assume that the dimensions to which each fact is related are up-to-date at the time the fact ETL Load Plans are executed. This assumption is the basis for certain fact calculations that would provide erroneous results if the assumption were not true. For example, in the *received CRFs* fact, the value of the *pCRF entry complete measure* depends on whether or not the study requires second pass entry. But that piece of information -- second pass entry required -- is obtained from an attribute of the Study dimension. So, if the second-pass requirement for a study changes, and the change is not applied to the Study dimension, the Received CRF fact attributes will contain incorrect values.

As shipped, OHSCDA ETL load plan ensure this interlock by executing the ETL for related dimensions immediately before running the ETL for a fact. This is standard warehouse management practice, but especially important given the interdependence of the dimensions and the fact. The need to execute dimension ETL immediately before corresponding fact ETL, and the danger of not doing it, is emphasized here because it is possible (though discouraged) to modify these shipped load plan.

To modify one or more tables or columns without any changes to the associated ETL programs (typically to widen a column):

1. Change the table properties as needed.
2. Save the mapping and refresh the scenario.
3. Navigate to the load plan and verify the scenario to which it is referred.

Note: Ensure that changes to the tables or columns are compatible with the table that is installed in the data warehouse schema. For example, if you are reducing the length of a number column from 15 to 10, ensure that the change is compatible with the existing data in the table.

2.6 Scheduling an ETL Load Plan

When you submit a Load Plan for execution in ODI, you can schedule it to execute at regular intervals. To schedule a Load Plan, perform the following tasks:

1. Select **Load Plans and Scenarios**.
2. Expand the **CDA - Complete Warehouse** load plan.
3. Right-click **Scheduling** and select **New Scheduling**.
4. Select **Definition**.
5. Select the required values for the following:
 - **Context**
 - **Logical Agent**
 - **Log Level**
 - **Status**
 - **Execution**
6. Click **Save**.

Troubleshooting

This appendix contains the following topics:

- [Sorting and Displaying of Null Values in Reports](#) on page A-1
- [Aborting a Load Plan](#) on page A-2

A.1 Sorting and Displaying of Null Values in Reports

In order to understand results shown in OBIEE reports, it may be necessary to understand how null values are sorted and displayed in reports.

Oracle uses NULL as a pseudo-value for a table cell when there is no actual value. For example, if the number of documents awaiting completion for a site is unknown, the column containing that attribute of the site will be set to null in the database.

As null values can appear in among data, OBIEE has rules that determine how to display the null values. And as OBIEE supports sorting of data in a column, it has rules for how nulls should be sorted.

The following are the rules:

- Oracle's sorting order cause a null value to be treated as greater than any non-null value.
- In table views, OBIEE generally displays null values as empty cells.

The exception is when the request designer has specified that the user can navigate to a different request by clicking on a value in the column that contains null. In that case, in order to give the user something to click on, OBIEE displays the null value as a zero.

These rules can produce unexpected results. This following section describes how to interpret such unexpected results. It also describes actions you can take in creating OBIEE requests to override OBIEE's default rules.

The results of these rules are:

- If the data in a column contain nulls and non-nulls, and the column is sorted, and navigation is not enabled from cells in the column, then:
 - Nulls will display as blank cells
 - Blank cells will sort as larger than the largest non-null value
- If the data in a column contain nulls and non-nulls, and the column is sorted, and navigation is enabled from cells in the column, then:
 - Nulls will display as zeros

- Cells representing nulls (but now displaying as zeros) will sort as larger than the largest non-null value. If there are actual zeros in the column as well, they will sort as smaller than the smallest positive value in the column. So, if you have both real zero values and null values, and cell-based navigation is enabled, and you sort the column, you will get two clumps of zeros - one representing the nulls, the other representing the actual zeros - separated by the non-negative actual values.
- OBIEE does have a capability that can be used to make it easier to identify null values. In requests, you can use the IFNULL function to specify that NULL should be replaced by a large negative value that could not be a real value for the column. For instance, if "# Documents Outstanding" could be null in your data, and you want to include it in a request, you could change the functional definition of the column in the request from "# DocumentsOutstanding" to IFNULL("# Documents Outstanding", -99). This would cause nulls to sort and display as if their value was -99.

If you use IFNULL, it is important that you:

- Choose a value that could not also be a legitimate value (this may vary from column to column, though it is preferable to use the same IFNULL replacement across all columns).
- Communicate to your end users the meaning of the IFNULL values.

A.2 Aborting a Load Plan

Perform the following steps in ODI to abort a load plan:

1. Select **Load Plans and Scenarios** in the **Operator** dialog box.
2. In **Agents** section, expand **Agent used for Execution**.
3. Right-click the Load Plan you want to abort.
4. Click **Stop Normal** or **Stop Immediate**, as required.
5. Select **Physical Agent**.
6. Click **OK**.

See Also:

Oracle Data Integrator Online Help