Oracle® Database Gateway for Sybase User's Guide





Oracle Database Gateway for Sybase User's Guide, 23c

F47549-02

Copyright © 2007, 2023, Oracle and/or its affiliates.

Primary Author: Rhonda Day

Contributing Authors: Vira Goorah, Juan Pablo Ahues-Vasquez, Peter Castro, Charles Benet, Peter Wong, Govind Lakkoju

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, software documentation, data (as defined in the Federal Acquisition Regulation), 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 embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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®, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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

	tation Accessibility	x xi
Convention		χi
Conventio	nio	Al
Introdu	ction to the Oracle Database Gateway for Sybase	
1.1 Ove	erview of Oracle Database Gateways	1-1
1.2 Abo	out Heterogeneous Services Technology	1-2
1.3 Abo	out Oracle Database Gateway for Sybase	1-2
Sybase	e Gateway Features and Restrictions	
2.1 Rer	note Insert Rowsource	2-2
2.2 Usii	ng the Pass-Through Feature	2-2
2.3 Exe	ecuting Stored Procedures and Functions	2-3
2.3.1	Return Values and Stored Procedures	2-3
2.3.2	Result Sets and Stored Procedures	2-4
2.	3.2.1 OCI Program Fetching from Result Sets in Sequential Mode	2-6
2.	3.2.2 PL/SQL Program Fetching from Result Sets in Sequential Mode	2-7
2.4 CH	AR Semantics	2-8
2.5 Mul	ti-byte Character Sets Ratio Suppression	2-8
2.6 IPv	6 Support	2-8
2.7 Gat	eway Session IDLE Timeout	2-9
2.8 Dat	abase Compatibility Issues for Sybase	2-9
2.8.1	Chained Mode	2-9
2.8.2	Column Definitions	2-9
2.8.3	Naming Rules	2-10
2.	8.3.1 Rules for Naming Objects	2-10
2.	8.3.2 Case Sensitivity	2-10
2.8.4	Data Types	2-11



	2.8	3.4.2	Data Type Conversion	2-11
	2.8.5	Que	ries	2-12
	2.8	3.5.1	Row Selection	2-12
	2.8	3.5.2	Empty Strings	2-12
	2.8	3.5.3	Empty Bind Variables	2-13
	2.8.6	Lock	ing	2-13
	2.8.7	Syba	ase Identifiers Length Limit	2-13
2.9	Knov	vn Re	strictions	2-13
	2.9.1	Tran	sactional Integrity	2-14
	2.9.2	Tran	saction Capability	2-15
	2.9.3	COM	MMIT or ROLLBACK in PL/SQL Cursor Loops Closes Open Cursors	2-15
	2.9.4	Store	ed Procedures	2-15
	2.9.5	Pass	s-Through Feature	2-15
	2.9.6	Syba	ase NCHAR and NVARCHAR Data Types	2-16
	2.9.7	SQL	Syntax	2-16
	2.9	.7.1	WHERE CURRENT OF Clause	2-16
	2.9	.7.2	CONNECT BY Clause	2-16
	2.9	.7.3	ROWID	2-16
	2.9	.7.4	Subqueries in INSERT Statement	2-17
	2.9	.7.5	EXPLAIN PLAN Statement	2-17
	2.9.8	SQL	*Plus COPY Command with Lowercase Table Names	2-17
	2.9.9	Data	abase Links	2-17
2.10	0 Knc	wn P	roblems	2-17
	2.10.1	End	crypted Format Login	2-18
	2.10.2	Dat	te Arithmetic	2-18
	2.10.3	Syb	pase IMAGE and TEXT Data Type	2-18
	2.10.4	Stri	ing Functions	2-19
	2.10.5	Sch	nema Names and PL/SQL	2-19
	2.10.6	Dat	ta Dictionary Views and PL/SQL	2-19
	2.10.7	Sto	ored Procedures	2-19
Ca	se St	udie	2S	
3.1	Distri	ibutior	n Media Contents	3-1
3.2	Acce	ssing	the Demonstration Files	3-2
3.3	Dem	onstra	ation Requirements	3-2
3.4	Crea	ting D	Demonstration Tables	3-2
	3.4.1	Dem	nonstration Table Definitions	3-3
	3.4.2	Dem	nonstration Table Contents	3-4
3.5	Case	1: Si	mple Queries	3-5
3.6	Case	2: A	More Complex Query	3-5



3

3.7 Case 3: Joining Sybase Tables	3-5
3.8 Case 4: Write Capabilities	3-5
3.8.1 DELETE Statement	3-6
3.8.2 UPDATE Statement	3-6
3.8.3 INSERT Statement	3-6
3.9 Case 5: Data Dictionary Query	3-6
3.10 Case 6: The Pass-Through Feature	3-6
3.10.1 UPDATE Statement	3-7
3.10.2 SELECT Statement	3-7
3.11 Case 7: Executing Stored Procedures	3-7
Data Type Conversion	
Compared COL Compared Forestings	
Supported SQL Syntax and Functions	
B.1 Supported SQL Statements	B-1
B.1.1 DELETE B.1.2 INSERT	B-2
	B-2
B.1.3 SELECT B.1.4 UPDATE	B-2 B-2
B.2 Oracle Functions	B-3
B.2.1 Functions Not Supported by Sybase B.2.2 Functions Supported by Sybase	B-3 B-3
	в-з В-3
·	
B.2.2.2 Comparison Operators B.2.2.3 Group Functions	B-4 B-4
B.2.2.4 String Functions	B-4
B.2.2.5 Other Functions	B-5
B.2.2.5 Other Functions	B-3
Data Dictionary	
C.1 Sybase System Tables	C-3
C.2 Accessing the Gateway Data Dictionary	C-3
C.3 Direct Queries to Sybase Tables	C-4
C.4 Supported Views and Tables	C-4
C.5 Data Dictionary Mapping	C-5
C.5.1 Default Column Values	C-6
C.6 Gateway Data Dictionary Descriptions	C-6
C.7 ALL_CATALOG	C-6
C.8 ALL_COL_COMMENTS	C-7
C.9 ALL_CONS_COLUMNS	C-7



C.10	ALL_CONSTRAINTS	C-7
C.11	ALL_IND_COLUMNS	C-8
C.12	ALL_INDEXES	C-8
C.13	ALL_OBJECTS	C-9
C.14	ALL_TAB_COLUMNS	C-10
C.15	ALL_TAB_COMMENTS	C-11
C.16	ALL_TABLES	C-11
C.17	ALL_USERS	C-12
C.18	ALL_VIEWS	C-13
C.19	DBA_CATALOG	C-13
C.20	DBA_COL_COMMENTS	C-13
C.21	DBA_OBJECTS	C-14
C.22	DBA_TAB_COLUMNS	C-14
C.23	DBA_TAB_COMMENTS	C-15
C.24	DBA_TABLES	C-15
C.25	DICT_COLUMNS	C-16
C.26	DICTIONARY	C-17
C.27	DUAL	C-17
C.28	TABLE_PRIVILEGES	C-17
C.29	USER_CATALOG	C-18
C.30	USER_COL_COMMENTS	C-18
C.31	USER_CONS_COLUMNS	C-18
C.32	USER_CONSTRAINTS	C-18
C.33	USER_IND_COLUMNS	C-19
C.34	USER_INDEXES	C-19
C.35	USER_OBJECTS	C-21
C.36	USER_TAB_COLUMNS	C-21
C.37	USER_TAB_COMMENTS	C-22
C.38	USER_TABLES	C-22
C.39	USER_USERS	C-23
C.40	USER_VIEWS	C-24
Initia	lization Parameters	
D.1	Initialization Parameter File Syntax	D-3
D.2	HS_CALL_NAME	D-4
D.3	HS_DB_DOMAIN	D-5
D.4	HS_DB_INTERNAL_NAME	D-5
D.5	HS_DB_NAME	D-6
D.6	HS_DESCRIBE_CACHE_HWM	D-6
D.7	HS_FDS_ARRAY_EXEC	D-6



D

D.8	HS_FDS_AUTHENTICATE_METHOD	D-7
D.9	HS_FDS_CONNECT_INFO	D-7
D.10	HS_FDS_DATE_MAPPING	D-8
D.11	HS_FDS_ENCRYPT_SESSION	D-8
D.12	HS_FDS_FETCH_ROWS	D-8
D.13	HS_FDS_PROC_IS_FUNC	D-9
D.14	HS_FDS_QUOTE_IDENTIFIER	D-9
D.15	HS_FDS_RECOVERY_ACCOUNT	D-9
D.16	HS_FDS_RECOVERY_PWD	D-10
D.17	HS_FDS_REMOTE_DB_CHARSET	D-10
D.18	HS_FDS_REPORT_REAL_AS_DOUBLE	D-10
D.19	HS_FDS_RESULTSET_SUPPORT	D-11
D.20	HS_FDS_RSET_RETURN_ROWCOUNT	D-11
D.21	HS_FDS_SQLLEN_INTERPRETATION	D-11
D.22	HS_FDS_SUPPORT_STATISTICS	D-12
D.23	HS_FDS_TIMESTAMP_MAPPING	D-12
D.24	HS_FDS_TRACE_LEVEL	D-12
D.25	HS_FDS_TRANSACTION_ISOLATION	D-13
D.26	HS_FDS_TRANSACTION_LOG	D-13
D.27	HS_FDS_TRUSTSTORE_FILE	D-14
D.28	HS_FDS_TRUSTSTORE_PASSWORD	D-14
D.29	HS_FDS_VALIDATE_SERVER_CERT	D-14
D.30	HS_IDLE_TIMEOUT	D-15
D.31	HS_KEEP_REMOTE_COLUMN_SIZE	D-15
D.32	HS_LANGUAGE	D-16
D	0.32.1 Character Sets	D-16
D	0.32.2 Language	D-17
D	0.32.3 Territory	D-17
D.33	HS_LONG_PIECE_TRANSFER_SIZE	D-17
D.34	HS_NLS_LENGTH_SEMANTICS	D-17
D.35	HS_OPEN_CURSORS	D-18
D.36	HS_RPC_FETCH_REBLOCKING	D-18
D.37	HS_RPC_FETCH_SIZE	D-19
D.38	HS_TIME_ZONE	D-19
D.39	HS_TRANSACTION_MODEL	D-19
D.40	IFILE	D-20

Index





List of Tables

A-1	Data Type Mapping and Restrictions	A-1
C-1	Oracle Data Dictionary View Names and Sybase Equivalents	C-5



Preface

This manual describes the Oracle Database Gateway for Sybase, which enables Oracle client applications to access Sybase data through Structured Query Language (SQL). The gateway, with the Oracle database, creates the appearance that all data resides on a local Oracle database, even though the data can be widely distributed.

This preface covers the following topics:

- Audience
- · Documentation Accessibility
- Related Documents
- Conventions

Audience

This manual is intended for Oracle database administrators who perform the following tasks:

- Installing and configuring the Oracle Database Gateway for Sybase
- Diagnosing gateway errors
- Using the gateway to access Sybase data



Note:

You should understand the fundamentals of Oracle Database Gateways and the UNIX based platform before using this guide to install or administer the gateway.

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



Related Documents

For more information, see the following documents:

- Oracle Database New Features Guide
- Oracle Call Interface Programmer's Guide
- Oracle Database Administrator's Guide
- Oracle Database Development Guide
- Oracle Database Concepts
- Oracle Database Performance Tuning Guide
- Oracle Database Error Messages
- Oracle Database Globalization Support Guide
- Oracle Database Reference
- Oracle Database SQL Language Reference
- Oracle Database Net Services Administrator's Guide
- SQL*Plus User's Guide and Reference
- Oracle Database Heterogeneous Connectivity User's Guide
- Oracle Database Security Guide

Many of the examples in this book use the sample schemas of the seed database, which is installed by default when you install Oracle. Refer to *Oracle Database Sample Schemas* for information on how these schemas were created and how you can use them yourself.

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.
italic	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.



1

Introduction to the Oracle Database Gateway for Sybase

Oracle Database Gateways provide the ability to transparently access data residing in a non-Oracle system from an Oracle environment. The following sections briefly covers Heterogeneous Services, the technology on which the Oracle Database Gateway for Sybase is based.

To get a good understanding of generic gateway technology, Heterogeneous Services, and how Oracle Database Gateways fit in the picture, refer to the *Oracle Database Heterogeneous Connectivity User's Guide* first.

- Overview of Oracle Database Gateways
 - Heterogeneous data access is a problem that affects a lot of companies. A lot of companies run several different database systems. Each of these systems stores data and has a set of applications that run against it.
- About Heterogeneous Services Technology
 - Heterogeneous Services provides the generic technology for connecting to non-Oracle systems. As an integrated component of the database, Heterogeneous Services can exploit features of the database, such as the powerful SQL parsing and distributed optimization capabilities.
- About Oracle Database Gateway for Sybase
 - The capabilities, SQL mappings, data type conversions, and interface to the remote non-Oracle system are contained in the gateway. The gateway interacts with Heterogeneous Services to provide the transparent connectivity between Oracle and non-Oracle systems.

1.1 Overview of Oracle Database Gateways

Heterogeneous data access is a problem that affects a lot of companies. A lot of companies run several different database systems. Each of these systems stores data and has a set of applications that run against it.

Consolidation of this data in one database system is often hard - in large part because many of the applications that run against one database may not have an equivalent that runs against another. Until such time as migration to one consolidated database system is made feasible, it is necessary for the various heterogeneous database systems to interoperate.

Oracle Database Gateways provide the ability to transparently access data residing in a non-Oracle system from an Oracle environment. This transparency eliminates the need for application developers to customize their applications to access data from different non-Oracle systems, thus decreasing development efforts and increasing the mobility of the application. Applications can be developed using a consistent Oracle interface for both Oracle and Sybase.

Gateway technology is composed of two parts: a component that has the generic technology to connect to a non-Oracle system, which is common to all the non-Oracle systems, called Heterogeneous Services, and a component that is specific to the non-Oracle system that the

gateway connects to. Heterogeneous Services, in conjunction with the Oracle Database Gateway agent, enables transparent access to non-Oracle systems from an Oracle environment.

1.2 About Heterogeneous Services Technology

Heterogeneous Services provides the generic technology for connecting to non-Oracle systems. As an integrated component of the database, Heterogeneous Services can exploit features of the database, such as the powerful SQL parsing and distributed optimization capabilities.

Heterogeneous Services extend the Oracle SQL engine to recognize the SQL and procedural capabilities of the remote non-Oracle system and the mappings required to obtain necessary data dictionary information. Heterogeneous Services provides two types of translations: the ability to translate Oracle SQL into the proper dialect of the non-Oracle system as well as data dictionary translations that displays the metadata of the non-Oracle system in the local format. For situations where no translations are available, native SQL can be issued to the non-Oracle system using the pass-through feature of Heterogeneous Services.

Heterogeneous Services also maintains the transaction coordination between Oracle and the remote non-Oracle system, such as providing the two-phase commit protocol to ensure distributed transaction integrity, even for non-Oracle systems that do not natively support two-phase commit.



Oracle Database Heterogeneous Connectivity User's Guide for more information about Heterogeneous Services.

1.3 About Oracle Database Gateway for Sybase

The capabilities, SQL mappings, data type conversions, and interface to the remote non-Oracle system are contained in the gateway. The gateway interacts with Heterogeneous Services to provide the transparent connectivity between Oracle and non-Oracle systems.

The gateway can be installed on any machine independent of the Oracle or non-Oracle database. It can be the same machine as the Oracle database or on the same machine as the Sybase database or on a third machine as a standalone. Each configuration has its advantages and disadvantages. The issues to consider when determining where to install the gateway are network traffic, operating system platform availability, hardware resources and storage.



Sybase Gateway Features and Restrictions

After the gateway is installed and configured, you can use the gateway to access Sybase data, pass Sybase commands from applications to the Sybase database, perform distributed queries, and copy data.

Remote Insert Rowsource

A remote insert rowsource feature allows remote insert requiring local Oracle data to work through the Oracle database and Oracle Database Gateway. This functionality requires that the Oracle database and the Oracle Database Gateway to be version 12.2 or later.

Using the Pass-Through Feature

The gateway can pass Sybase commands or statements from the application to the Sybase database using the DBMS HS PASSTHROUGH package.

Executing Stored Procedures and Functions

Using the procedural feature, the gateway can execute stored procedures that are defined in the Sybase database. It is not necessary to relink the gateway or define the procedure to the gateway, but the procedure's access privileges must permit access by the gateway.

CHAR Semantics

This feature allows the gateway to optionally run in CHAR Semantics mode.

Multi-byte Character Sets Ratio Suppression

This feature optionally suppresses the ratio expansion from Sybase database to Oracle database involving multi-byte character set.

IPv6 Support

Besides full IPv6 support between Oracle databases and the gateway, IPv6 is also supported between this gateway and Sybase database.

Gateway Session IDLE Timeout

You can optionally choose to terminate long idle gateway sessions automatically with the gateway parameter HS IDLE TIMEOUT.

Database Compatibility Issues for Sybase

Sybase and Oracle databases function differently in some areas, causing compatibility problems.

Known Restrictions

The following sections describe the known restrictions and includes suggestions for dealing with them when possible.

Known Problems

This section describes known problems and includes suggestions for correcting them when possible for Oracle Database Gateway for Sybase.

2.1 Remote Insert Rowsource

A remote insert rowsource feature allows remote insert requiring local Oracle data to work through the Oracle database and Oracle Database Gateway. This functionality requires that the Oracle database and the Oracle Database Gateway to be version 12.2 or later.

By Oracle Database design, some distributed statement must be executed at the database link site. But in certain circumstances, there is data needed to execute these queries that must be fetched from the originating Oracle Database. Under homogeneous connections, the remote Oracle database would call back the source Oracle database for such data. But in heterogeneous connections, this is not viable, because this means that the Foreign Data Store would have to query call back functions, or data, that can only be provided by the Oracle instance that issued the query. In general, these kinds of statements are not something that can be supported through the Oracle Database Gateway.

The following categories of SQL statements results in a callback:

- Any DML with a sub-select, which refers to a table in Oracle database.
- Any DELETE, INSERT, UPDATE or "SELECT... FOR UPDATE..." SQL statement
 containing SQL functions or statements that needs to be executed at the
 originating Oracle database.
 - These SQL functions include USER, USERENV, and SYSDATE; and involve the selection of data from the originating Oracle database.
- Any SQL statement that involves a table in Oracle database, and a LONG or LOB column in a remote table.

An example of a remote INSERT statement that can work through the remote insert rowsource feature is as follows:

```
INSERT INTO gateway_table@gateway_link select * from local_table;
```

2.2 Using the Pass-Through Feature

The gateway can pass Sybase commands or statements from the application to the Sybase database using the <code>DBMS_HS_PASSTHROUGH</code> package.

Use the <code>DBMS_HS_PASSTHROUGH</code> package in a PL/SQL block to specify the statement to be passed to the Sybase database, as follows:

```
num_rows INTEGER;
BEGIN
   num_rows := DBMS_HS_PASSTHROUGH.EXECUTE_IMMEDIATE@SYBS('command');
END;
/
```

Where command cannot be one of the following:

- BEGIN TRANSACTION
- COMMIT
- ROLLBACK



- SAVE
- SHUTDOWN
- Sybase tool commands

The DBMS_HS_PASSTHROUGH package supports passing bind values and executing SELECT statements.



TRUNCATE cannot be used in a pass-through statement.

See Also:

Oracle Database PL/SQL Packages and Types Reference and Chapter 3, Features of Oracle Database Gateways, of Oracle Database Heterogeneous Connectivity User's Guide for more information about the DBMS HS PASSTHROUGH package.

2.3 Executing Stored Procedures and Functions

Using the procedural feature, the gateway can execute stored procedures that are defined in the Sybase database. It is not necessary to relink the gateway or define the procedure to the gateway, but the procedure's access privileges must permit access by the gateway.

Standard PL/SQL statements are used to execute a stored procedure.

The gateway supports stored procedures in three mutually exclusive modes:

- Normal mode: Have access to IN/OUT arguments only
- Return value mode: Have a return value for all stored procedures
- Resultset mode: Out values are available as last result set
- Return Values and Stored Procedures
 By default, all stored procedures and functions do not return a return value to the user.
- Result Sets and Stored Procedures
 The Oracle Database Gateway for Sybase provides support for stored procedures which return result sets.

2.3.1 Return Values and Stored Procedures

By default, all stored procedures and functions do not return a return value to the user.

To enable return values, set the ${\tt HS_FDS_PROC_IS_FUNC}$ parameter in the initialization parameter file.



See Also:

Initialization Parameters for information about both editing the initialization parameter file and the <code>HS_FDS_PROC_IS_FUNC</code> parameter.

Note:

If you set the <code>HS_FDS_PROC_IS_FUNC</code> gateway initialization parameter, you must change the syntax of the procedure execute statement for all existing stored procedures.

In the following example, the employee name JOHN SMYTHE is passed to the Sybase stored procedure REVISE_SALARY. The stored procedure retrieves the salary value from the Sybase database to calculate a new yearly salary for JOHN SMYTHE. The revised salary returned in RESULT is used to update EMP in a table of an Oracle database:

```
DECLARE
INPUT VARCHAR2(15);
RESULT NUMBER(8,2);
BEGIN
INPUT := 'JOHN SMYTHE';
RESULT := REVISE_SALARY@SYBS(INPUT);
UPDATE EMP SET SAL = RESULT WHERE ENAME =: INPUT;
END;
```

The procedural feature automatically converts non-Oracle data types to and from PL/SQL data types.

2.3.2 Result Sets and Stored Procedures

The Oracle Database Gateway for Sybase provides support for stored procedures which return result sets.

By default, all stored procedures and functions do not return a result set to the user. To enable result sets, set the ${\tt HS_FDS_RESULTSET_SUPPORT}$ parameter in the initialization parameter file.

See Also:

Initialization Parameters for information about both editing the initialization parameter file and the HS_FDS_RESULTSET_SUPPORT parameter. For further information about Oracle support for result sets in non-Oracle databases see Oracle Database Heterogeneous Connectivity User's Guide.



Note:

If you set the <code>HS_FDS_RESULTSET_SUPPORT</code> gateway initialization parameter, you must change the syntax of the procedure execute statement for all existing stored procedures or errors will occur.

When accessing stored procedures with result sets through the Oracle Database Gateway for Sybase, you will be in the sequential mode of Heterogeneous Services.

The Oracle Database Gateway for Sybase returns the following information to Heterogeneous Services during procedure description:

- All the input arguments of the remote stored procedure
- None of the output arguments
- One out argument of type ref cursor (corresponding to the first result set returned by the stored procedure)

Client programs have to use the virtual package function

dbms_hs_result_set.get_next_result_set to get the ref cursor for subsequent result sets. The last result set returned is the out argument from the procedure.

The limitations of accessing result sets are the following:

- Result sets returned by a remote stored procedure have to be retrieved in the order in which they were placed on the wire
- On execution of a stored procedure, all result sets returned by a previously executed stored procedure will be closed (regardless of whether the data has been completely

In the following example, the Sybase stored procedure is executed to fetch the contents of the emp and dept tables from Sybase:

```
create procedure REFCURPROC (@arg1 varchar(255), @arg2 varchar(255) output)
as
select @arg2 = @arg1
select * from EMP
select * from DEPT
go
```

This stored procedure assigns the input parameter arg1 to the output parameter arg2, opens the query <code>SELECT * FROM EMP</code> in ref cursor rc1, and opens the query <code>SELECT * FROM DEPT</code> in ref cursor rc2.



Chained mode must be set before creating the stored procedure. Issue the following command in Sybase: set chained on

- OCI Program Fetching from Result Sets in Sequential Mode
 An example of an OCI program fetching from result sets in sequential mode.
- PL/SQL Program Fetching from Result Sets in Sequential Mode
 An example of a PL/SQL program fetching from result sets in sequential mode.



2.3.2.1 OCI Program Fetching from Result Sets in Sequential Mode

An example of an OCI program fetching from result sets in sequential mode.

The following example shows OCI program fetching from result sets in sequential mode:

```
OCIEnv *ENVH;
OCISvcCtx *SVCH;
OCIStmt *STMH;
OCIError *ERRH;
OCIBind *BNDH[3];
OraText arg1[20];
OraText arg2[255];
OCIResult *rset;
OCIStmt *rstmt;
ub2 rcode[3];
ub2 rlens[3];
sb2 inds[3];
OraText *stmt = (OraText *) "begin refcurproc@SYBS(:1,:2,:3); end;";
OraText *n_rs_stm = (OraText *)
  "begin :ret := DBMS HS RESULT SET.GET NEXT RESULT SET@SYBS; end;";
/* Prepare procedure call statement */
/* Handle Initialization code skipped */
OCIStmtPrepare(STMH, ERRH, stmt, strlen(stmt), OCI NTV SYNTAX, OCI DEFAULT);
/* Bind procedure arguments */
inds[0] = 0;
strcpy((char *) arg1, "Hello World");
rlens[0] = strlen(arg1);
OCIBindByPos(STMH, &BNDH[0], ERRH, 1, (dvoid *) arg1, 20, SQLT CHR,
             (dvoid *) &(inds[0]), &(rlens[0]), &(rcode[0]), 0, (ub4 *) 0,
             OCI DEFAULT);
inds[1] = -1;
OCIBindByPos(STMH, &BNDH[1], ERRH, 1, (dvoid *) arg2, 20, SQLT CHR,
             (dvoid *) &(inds[1]), &(rlens[1]), &(rcode[1]), 0, (ub4 *) 0,
             OCI DEFAULT);
inds[2] = 0;
rlens[2] = 0;
OCIDescriptorAlloc(ENVH, (dvoid **) &rset, OCI_DTYPE_RSET, 0, (dvoid **) 0);
OCIBindByPos(STMH, &BNDH[2], ERRH, 2, (dvoid *) rset, 0, SQLT RSET,
             (dvoid *) &(inds[2]), &(rlens[2]), &(rcode[2]),
             0, (ub4 *) 0, OCI DEFAULT);
/* Execute procedure */
OCIStmtExecute(SVCH, STMH, ERRH, 1, 0, (CONST OCISnapshot *) 0,
               (OCISnapshot *) 0, OCI DEFAULT);
/* Convert result set to statement handle */
OCIResultSetToStmt(rset, ERRH);
rstmt = (OCIStmt *) rset;
/* After this the user can fetch from rstmt */
/* Issue get next result set call to get handle to next_result set */
/* Prepare Get next result set procedure call */
OCIStmtPrepare(STMH, ERRH, n rs stm, strlen(n rs stm), OCI NTV SYNTAX,
```



2.3.2.2 PL/SQL Program Fetching from Result Sets in Sequential Mode

An example of a PL/SQL program fetching from result sets in sequential mode.

Assume that the table <code>loc_emp</code> is a local table exactly like the Sybase emp table. The same assumption applies for <code>loc_dept.outargs</code> is a table with columns corresponding to the out arguments of the Sybase stored procedure.

```
create or replace package rcpackage is
  type RCTYPE is ref cursor;
end rcpackage;
declare
 rc1 rcpackage.rctype;
  rec1 loc emp%rowtype;
 rc2 rcpackage.rctype;
  rec2 loc dept%rowtype;
  rc3 rcpackage.rctype;
  rec3 outargs%rowtype;
  out arg varchar2(255);
begin
  -- Execute procedure
  out arg := null; refcurproc@SYBS('Hello World', out arg, rc1);
  -- Fetch 20 rows from the remote emp table and insert them into loc_emp
  for i in 1 .. 20 loop
    fetch rcl into recl;
    insert into loc emp (recl.empno, recl.ename, recl.job,
    recl.mgr, recl.hiredate, recl.sal, recl.comm, recl.deptno);
  end loop;
  -- Close ref cursor
  close rc1;
  -- Get the next result set returned by the stored procedure
  rc2 := dbms hs result set.get next result set@SYBS;
  -- Fetch 5 rows from the remote dept table and insert them into loc dept
```

```
for i in 1 .. 5 loop
   fetch rc2 into rec2;
   insert into loc dept values (rec2.deptno, rec2.dname, rec2.loc);
  end loop;
  --Close ref cursor
  close rc2;
  -- Get the output arguments from the remote stored procedure
  -- Since we are in sequential mode, they will be returned in the
  -- form of a result set
  rc3 := dbms hs result set.get next result set@SYBS;
  -- Fetch them and insert them into the outarguments table
  fetch rc3 into rec3;
  insert into outargs (rec3.outarg, rec3.retval);
  -- Close ref cursor
  close rc3:
end;
```

2.4 CHAR Semantics

This feature allows the gateway to optionally run in CHAR Semantics mode.

Rather than always describing Sybase CHAR columns as CHAR (n BYTE), this feature describes them as CHAR (n CHAR) and VARCHAR (n CHAR). The concept is similar to Oracle database CHAR Semantics. You need to specify

HS_NLS_LENGTH_SEMANTICS=CHAR gateway parameter to activate this option. Refer to Initialization Parameters for more detail.

2.5 Multi-byte Character Sets Ratio Suppression

This feature optionally suppresses the ratio expansion from Sybase database to Oracle database involving multi-byte character set.

By default, Oracle gateways assume the worst ratio to prevent data being truncated or insufficient buffer size situation. However, if you have specific knowledge of your Sybase database and do not want the expansion to occur, you can specify HS_KEEP_REMOTE_COLUMN_SIZE parameter to suppress the expansion. Refer to Initialization Parameters for more detail.

2.6 IPv6 Support

Besides full IPv6 support between Oracle databases and the gateway, IPv6 is also supported between this gateway and Sybase database.

Refer to the ${\tt HS_FDS_CONNECT_INFO}$ parameter in Initialization Parameters for more detail.



2.7 Gateway Session IDLE Timeout

You can optionally choose to terminate long idle gateway sessions automatically with the gateway parameter HS IDLE TIMEOUT.

Specifically, when a gateway session is idle for more than the specified time limit, the gateway session is terminated with any pending update rolled back.

2.8 Database Compatibility Issues for Sybase

Sybase and Oracle databases function differently in some areas, causing compatibility problems.

Chained Mode

The gateway supports the ANSI-standard chained mode.

Column Definitions

By default, a Sybase table column cannot contain null values unless \mathtt{NULL} is specified in the column definition.

Naming Rules

The following topics discuss the naming rules issues.

Data Types

The following topics discuss data type issues.

Oueries

The following topics discuss query issues.

Locking

The locking model for a Sybase database differs significantly from the Oracle model.

Sybase Identifiers Length Limit

By default, the gateway will always quote identifiers. However, certain Sybase releases have a limit of 30 characters for identifiers such as table or column names and quotes are considered part of the names when checking against this limit.

2.8.1 Chained Mode

The gateway supports the ANSI-standard chained mode.

Sybase stored procedures must be written for this mode. Running in chained mode allows the gateway to extend the Oracle two-phase commit protection to transactions updating Oracle and Sybase databases.

2.8.2 Column Definitions

By default, a Sybase table column cannot contain null values unless \mathtt{NULL} is specified in the column definition.

In compliance with the ANSI standard, the Sybase database option "allow nulls by default" can be set to true to change the default column definition to NULL.

For an Oracle table, null values are allowed in a column unless ${\tt NOT}$ ${\tt NULL}$ is specified in the column definition.



2.8.3 Naming Rules

The following topics discuss the naming rules issues.

- Rules for Naming Objects
 Oracle and Sybase use different database object naming rules.
- Case Sensitivity

The Oracle database defaults to uppercase unless you surround identifiers with double quote characters.

2.8.3.1 Rules for Naming Objects

Oracle and Sybase use different database object naming rules.

For example, the maximum number of characters allowed for each object name can be different. Also, the use of single and double quotation marks, case sensitivity, and the use of alphanumeric characters can all be different.



Oracle Database Reference and Sybase documentation.

2.8.3.2 Case Sensitivity

The Oracle database defaults to uppercase unless you surround identifiers with double quote characters.

For example, to refer to the Sybase table called emp, enter the name with double quote characters, as follows:

```
SQL> SELECT * FROM "emp"@SYBS;
```

However, to refer to the Sybase table called emp owned by SCOTT from an Oracle application, enter the following:

```
SQL> SELECT * FROM "Scott"."emp"@SYBS;
```

If the Sybase table called emp is owned by SCOTT, a table owner name in uppercase letters, you can enter the owner name without double quote characters, as follows:

```
SQL> SELECT * FROM SCOTT."emp"@SYBS;
Or:
SQL> SELECT * FROM scott."emp"@SYBS;
```

Oracle recommends that you surround all Sybase object names with double quote characters and use the exact letter case for the object names as they appear in the Sybase data dictionary. This convention is not required when referring to the supported Oracle data dictionary tables or views listed in Data Dictionary.

If existing applications cannot be changed according to these conventions, create views in Oracle to associate Sybase names to the correct letter case. For example, to

refer to the Sybase table emp from an existing Oracle application by using only uppercase names, define the following view:

```
SQL> CREATE VIEW EMP (EMPNO, ENAME, SAL, HIREDATE)
   AS SELECT "empno", "ename", "sal", "hiredate"
   FROM "emp"@SYBS;
```

With this view, the application can issue statements such as the following:

```
SQL> SELECT EMPNO, ENAME FROM EMP;
```

Using views is a workaround solution that duplicates data dictionary information originating in the Sybase data dictionary. You must be prepared to update the Oracle view definitions whenever the data definitions for the corresponding tables are changed in the Sybase database.

2.8.4 Data Types

The following topics discuss data type issues.

- Binary Literal Notation
 Oracle SQL uses hexadecimal digits surrounded by single quotes to express literal values being compared or inserted into columns defined as data type RAW.
- Data Type Conversion
 Sybase does not support implicit date conversions. Such conversions must be explicit.

2.8.4.1 Binary Literal Notation

Oracle SQL uses hexadecimal digits surrounded by single quotes to express literal values being compared or inserted into columns defined as data type RAW.

This notation is not converted to syntax compatible with the Sybase VARBINARY and BINARY data types (a 0x followed by hexadecimal digits, surrounded by single quotes).

For example, the following statement is not supported:

```
SQL> INSERT INTO BINARY TAB@SYBS VALUES ('0xff')
```

Where BINARY_TAB contains a column of data type VARBINARY or BINARY. Use bind variables when inserting into or updating VARBINARY and BINARY data types.

2.8.4.2 Data Type Conversion

Sybase does not support implicit date conversions. Such conversions must be explicit.

For example, the gateway issues an error for the following SELECT statement:

```
SELECT DATE COL FROM TEST@SYBS WHERE DATE COL = "1-JAN-2001";
```

To avoid problems with implicit conversions, add explicit conversions, as in the following:

```
SELECT DATE COL FROM TEST@SYBS WHERE DATE COL = TO DATE("1-JAN-2001")
```



See Also:

Data Type Conversion for more information about restrictions on data types.

2.8.5 Queries

The following topics discuss query issues.

Row Selection

Sybase evaluates a query condition for all selected rows before returning any of the rows.

Empty Strings

Oracle processes an empty string in a SQL statement as a null value. Sybase processes an empty string as an empty string.

Empty Bind Variables

For VARCHAR bind variables, the gateway passes empty bind variables to the Sybase database as a \mathtt{NULL} value.

2.8.5.1 Row Selection

Sybase evaluates a query condition for all selected rows before returning any of the rows.

If there is an error in the evaluation process for one or more rows, no rows are returned even though the remaining rows satisfy the condition.

Oracle evaluates the query condition row-by-row and returns a row when the evaluation is successful. Rows are returned until a row fails the evaluation.

2.8.5.2 Empty Strings

Oracle processes an empty string in a SQL statement as a null value. Sybase processes an empty string as an empty string.

When comparing an empty string, the gateway passes literal empty strings to the Sybase database without any conversion. If you intended an empty string to represent a null value, Sybase does not process the statement that way; it uses the empty string.

You can avoid this problem by using NULL or IS NULL in the SQL statement instead of the empty string syntax, as in the following example:

```
SELECT * from "emp"@SYBS where "ename" IS NULL;
```

Selecting an empty string

For VARCHAR columns, the gateway returns an empty string to the Oracle database as NULL value.

For CHAR columns, the gateway returns the full size of the column with each character as empty space (' ').



2.8.5.3 Empty Bind Variables

For VARCHAR bind variables, the gateway passes empty bind variables to the Sybase database as a NULL value.

2.8.6 Locking

The locking model for a Sybase database differs significantly from the Oracle model.

The gateway depends on the underlying Sybase behavior, so Oracle applications that access Sybase through the gateway can be affected by the following possible scenarios:

- Read access might block write access
- · Write access might block read access
- Statement-level read consistency is not guaranteed



Sybase documentation for information about the Sybase locking model.

2.8.7 Sybase Identifiers Length Limit

By default, the gateway will always quote identifiers. However, certain Sybase releases have a limit of 30 characters for identifiers such as table or column names and quotes are considered part of the names when checking against this limit.

Therefore, when quotes are used, you can only specify 28 characters. In order to support the maximum length limit in those Sybase releases, you need to specify HS_FDS_QUOTE_IDENTIFIER=FALSE in your gateway initialization parameter file. Setting this initialization parameter will cause the gateway to send identifiers without quotes. However, it has the side effect of precluding the use of identifiers that contain dots (.) or spaces, and the identifiers will follow the case sensitivity of the Sybase database being used.

2.9 Known Restrictions

The following sections describe the known restrictions and includes suggestions for dealing with them when possible.

If you encounter incompatibility problems not listed in this section or in "Known Problems", contact Oracle Support Services.

The following restriction also applies:

 When negative numbers are used as the second parameter in a SUBSTR function, incorrect results are returned. This is due to incompatibility between the Oracle SUBSTR function and the equivalent in Sybase.



Note:

If you have any questions or concerns about the restrictions, contact Oracle Support Services.

Transactional Integrity

The gateway cannot guarantee transactional integrity in the following cases.

Transaction Capability

The gateway does not support savepoints.

COMMIT or ROLLBACK in PL/SQL Cursor Loops Closes Open Cursors
 Any COMMIT or ROLLBACK issued in a PL/SQL cursor loop closes all open cursors,
 which can result in an error.

Stored Procedures

Oracle Database Gateway for Sybase and stored procedures.

Pass-Through Feature

DDL statements executed by Sybase using the gateway pass-through feature might fail if they are in a multi-statement transaction.

Sybase NCHAR and NVARCHAR Data Types
 The gateway cannot select a column defined with a Sybase NCHAR or NVARCHAR data type.

SQL Syntax

The following topics discuss the restrictions on SQL syntax.

- SQL*Plus COPY Command with Lowercase Table Names
 You need to use double guotes to wrap around lowercase table names.
- Database Links

The gateway is not multithreaded and cannot support shared database links.

2.9.1 Transactional Integrity

The gateway cannot guarantee transactional integrity in the following cases.

- When a statement that is processed by the gateway causes an implicit commit in the target database
- When the target database is configured to work in autocommit mode



Oracle strongly recommends the following:

- If you know that executing a particular statement causes an implicit commit in the target database, then ensure that this statement is executed in its own transaction.
- Do not configure the target database to work in autocommit mode.



2.9.2 Transaction Capability

The gateway does not support savepoints.

If a distributed update transaction is under way involving the gateway and a user attempts to create a savepoint, the following error occurs:

ORA-02070: database dblink does not support savepoint in this context

By default, the gateway is configured as COMMIT_CONFIRM and it is always the commit point site when the Sybase database is updated by the transaction.

2.9.3 COMMIT or ROLLBACK in PL/SQL Cursor Loops Closes Open Cursors

Any COMMIT or ROLLBACK issued in a PL/SQL cursor loop closes all open cursors, which can result in an error.

For example:

ORA-1002: fetch out of sequence

To prevent this error, move the COMMIT or ROLLBACK statement outside the cursor loop.

2.9.4 Stored Procedures

Oracle Database Gateway for Sybase and stored procedures.

Consider the following with regard to stored procedures:

- Changes issued through stored procedures that embed commits or rollbacks cannot be controlled by the Oracle transaction manager or Oracle COMMIT or ROLLBACK commands.
- When accessing stored procedures with result sets through the Oracle Database
 Gateway for Sybase, you must work in the sequential mode of Heterogeneous Services.
- When accessing stored procedures with multiple result sets through the Oracle Database Gateway for Sybase, you must read all the result sets before continuing.
- Output parameters of stored procedures must be initialized to a NULL value.
- Oracle Database Gateway for Sybase does not support output parameters or stored procedures with output parameters, inside a pass through query.

2.9.5 Pass-Through Feature

DDL statements executed by Sybase using the gateway pass-through feature might fail if they are in a multi-statement transaction.

Set the Sybase option "ddl in tran" to allow DDL statements in a transaction.

Oracle recommends that you place a DDL statement in its own transaction when executing such a statement with the pass-through feature. An explicit COMMIT must be issued after the DDL statement.



If the SQL statements being passed through the gateway result in an implicit commit at the Sybase database, the Oracle transaction manager is unaware of the commit and an Oracle ROLLBACK command cannot be used to roll back the transaction.

2.9.6 Sybase NCHAR and NVARCHAR Data Types

The gateway cannot select a column defined with a Sybase NCHAR or NVARCHAR data type.

2.9.7 SQL Syntax

The following topics discuss the restrictions on SQL syntax.

WHERE CURRENT OF Clause

UPDATE and DELETE statements with the WHERE CURRENT OF clause are not supported by the gateway because they rely on the Oracle ROWID implementation.

CONNECT BY Clause

The gateway does not support the CONNECT BY clause in a SELECT statement.

ROWID

The Oracle ROWID implementation is not supported.

Subqueries in INSERT Statement

Subqueries of INSERT statements cannot use multiple aliases for the same table.

EXPLAIN PLAN Statement

The EXPLAIN PLAN statement is not supported.



Supported SQL Syntax and Functions for more information about restrictions on SQL syntax.

2.9.7.1 WHERE CURRENT OF Clause

UPDATE and DELETE statements with the WHERE CURRENT OF clause are not supported by the gateway because they rely on the Oracle ROWID implementation.

To update or delete a specific row through the gateway, a condition style <code>WHERE</code> clause must be used.

2.9.7.2 CONNECT BY Clause

The gateway does not support the CONNECT BY clause in a SELECT statement.

2.9.7.3 ROWID

The Oracle ROWID implementation is not supported.



2.9.7.4 Subqueries in INSERT Statement

Subqueries of INSERT statements cannot use multiple aliases for the same table.

For example, the following statement is not supported:

2.9.7.5 EXPLAIN PLAN Statement

The EXPLAIN PLAN statement is not supported.

2.9.8 SQL*Plus COPY Command with Lowercase Table Names

You need to use double quotes to wrap around lowercase table names.

For example:

```
copy from tkhouser/tkhouser@inst1 insert loc_tkhodept using select * from
"tkhodept"@holink2;
```

2.9.9 Database Links

The gateway is not multithreaded and cannot support shared database links.

Each gateway session spawns a separate gateway process and connections cannot be shared.

2.10 Known Problems

This section describes known problems and includes suggestions for correcting them when possible for Oracle Database Gateway for Sybase.

If you have any questions or concerns about the problems, contact Oracle Support Services. A current list of problems is available online. Contact your local Oracle office for information about accessing the list.

The known problems are as follows:

- Encrypted Format Login
 Oracle database no longer supports the initialization parameter DBLINK ENCRYPT LOGIN.
- Date Arithmetic
 Oracle Patchase Cateway for Subase does not supplied.

Oracle Database Gateway for Sybase does not support some date arithmetic functions.

- Sybase IMAGE and TEXT Data Type
 There are restrictions when using IMAGE and TEXT data types with Oracle Database
 Gateway for Sybase.
- String Functions

If you concatenate numeric literals using the " \mid \mid " operator when using the gateway to query a Sybase database, the result is an arithmetic addition.



Schema Names and PL/SQL

If you do not prefix a Sybase database object with its schema name in a SQL statement within a PL/SQL block, an error is returned.

Data Dictionary Views and PL/SQL

You cannot refer to data dictionary views in SQL statements that are inside a PL/SQL block.

Stored Procedures

Return values of stored procedures that return result sets are incorrect.

2.10.1 Encrypted Format Login

Oracle database no longer supports the initialization parameter ${\tt DBLINK\ ENCRYPT\ LOGIN}.$

Up to version 7.3, this parameter's default TRUE value prevented the password for the login user ID from being sent over the network (in the clear). Later versions automatically encrypt the password.

2.10.2 Date Arithmetic

Oracle Database Gateway for Sybase does not support some date arithmetic functions.

The following SQL expressions do not function correctly with the gateway:

```
date + numbernumber + datedate - numberdate1 - date2
```

Statements with the preceding expressions are sent to the Sybase database without any translation. Since Sybase does not support these date arithmetic functions, the statements return an error.

2.10.3 Sybase IMAGE and TEXT Data Type

There are restrictions when using ${\tt IMAGE}$ and ${\tt TEXT}$ data types with Oracle Database Gateway for Sybase.

The following restrictions apply when using IMAGE and TEXT data types:

- An unsupported SQL function cannot be used in a SQL statement that accesses a column defined as Sybase data type IMAGE or TEXT.
- You cannot use SQL*Plus to select data from a column defined as Sybase data type IMAGE or TEXT when the data is greater than 80 characters in length. Oracle recommends using Pro*C or Oracle Call Interface to access such data in a Sybase database.
- IMAGE and TEXT data cannot be read through pass-through queries.

The gateway does not support the PL/SQL function <code>COLUMN_VALUE_LONG</code> of the <code>DBMS SQL</code> package.





Supported SQL Syntax and Functions for more information about restrictions on SQL syntax.

2.10.4 String Functions

If you concatenate numeric literals using the "||" operator when using the gateway to query a Sybase database, the result is an arithmetic addition.

For example, the result of the following statement is 18:

```
SQL> SELECT 9 || 9 FROM DUAL@SYBS;
```

The result is 99 when using Oracle to query an Oracle database.

2.10.5 Schema Names and PL/SQL

If you do not prefix a Sybase database object with its schema name in a SQL statement within a PL/SQL block, an error is returned.

For example:

ORA-6550 PLS-201 Identifier table_name must be declared.

Change the SQL statement to include the schema name of the object.

2.10.6 Data Dictionary Views and PL/SQL

You cannot refer to data dictionary views in SQL statements that are inside a PL/SQL block.

2.10.7 Stored Procedures

Return values of stored procedures that return result sets are incorrect.



Case Studies

The following case studies for Sybase demonstrate some of the features of the Oracle Database Gateway.

You can verify that the gateway is installed and operating correctly by using the demonstration files included on the distribution media.

The demonstration files are automatically copied to disk when the gateway is installed.

Distribution Media Contents
 This topic discusses the distribution media contents.

Accessing the Demonstration Files

After a successful gateway installation, use the demonstration files stored in the directory $\protect\protec$

Demonstration Requirements

This topic discusses the requirements for the case studies.

Creating Demonstration Tables

This topic discusses how to create demonstration tables for Oracle Database Gateway for Sybase.

Case 1: Simple Queries

Demonstration using simple gueries.

Case 2: A More Complex Query

Demonstration using a more complex query.

Case 3: Joining Sybase Tables

Demonstration joining Sybase tables.

Case 4: Write Capabilities

Demonstration using write capabilities and the DELETE, UPDATE, and INSERT statements.

Case 5: Data Dictionary Query

Demonstration using a data dictionary query.

Case 6: The Pass-Through Feature

Demonstration using the pass-through feature.

• Case 7: Executing Stored Procedures

Demonstration executing stored procedures.

3.1 Distribution Media Contents

This topic discusses the distribution media contents.

The distribution media contains the following:

- Demonstration files.
- One SQL script file that creates the demonstration tables and stored procedures in the Sybase database.



 One SQL script file that drops the demonstration tables and stored procedures from the Sybase database.

3.2 Accessing the Demonstration Files

After a successful gateway installation, use the demonstration files stored in the directory $\protect\protec$

The directory contains the following demonstration files:

Demonstration Files	Demonstration Files
bldsybs.sql	case4c.sql
case1.sql	case5.sql
case2.sql	case6a.sql
case3.sql	case6b.sql
case4a.sql	case7.sql
case4b.sql	dropsybs.sql

3.3 Demonstration Requirements

This topic discusses the requirements for the case studies.

The case studies assume these requirements have been met:

- The gateway demonstration tables and stored procedures are installed in the Sybase database.
- The Oracle database has an account named SCOTT with a password of TIGER.
- The Oracle database has a database link called GTWLINK (set up as public or private to the user SCOTT) that connects the gateway to a Sybase database as SCOTT with password TIGER2. For example, you can create the database link as follows:

```
SQL> CREATE DATABASE LINK GTWLINK CONNECT TO SCOTT
2     IDENTIFIED BY TIGER2 USING 'GTWSID';
```

- Oracle Net Services is configured correctly and running.
- The Sybase environment variable, SYBASE, is set correctly.

3.4 Creating Demonstration Tables

This topic discusses how to create demonstration tables for Oracle Database Gateway for Sybase.

The case studies are based on the <code>GTW_EMP</code>, <code>GTW_DEPT</code>, and <code>GTW_SALGRADE</code> tables and the stored procedures <code>InsertDept</code> and <code>GetDept</code>. If the demonstration tables and stored procedures have not been created in the Sybase database, use the <code>bldsybs.sql</code> script to create them. Enter the following:

```
$ isql -USCOTT -PTIGER2 -ibldsybs.sql
```



The script creates the demonstration tables and stored procedures in the Sybase database accordingly:

```
CREATE TABLE GTW EMP (
EMPNO SMALLINT NOT NULL
         VARCHAR (10),
JOB VARCHAR(9),
MGR SMALLINT,
HIREDATE DATETIME,
SAL NUMERIC (7,2),
COMM
        NUMERIC(7,2),
DEPTNO SMALLINT)
CREATE TABLE GTW DEPT (
DEPTNO SMALLINT NOT NULL,
DEL 1.
          VARCHAR(14),
          VARCHAR (13))
LOC
go
CREATE TABLE GTW_SALGRADE (
GRADE MONEY,
LOSAL NUMERIC (9,4),
HISAL NUMERIC (9,4))
SET CHAINED ON
go
DROP PROCEDURE InsertDept
CREATE PROCEDURE InsertDept (@dno INTEGER,
  @dname VARCHAR(14), @loc VARCHAR(13))
AS INSERT INTO GTW DEPT VALUES (@dno, @dname, @loc)
DROP PROCEDURE GetDept
CREATE PROCEDURE GetDept (@dno INTEGER, @dname VARCHAR(14) OUTPUT)
AS SELECT @dname=DNAME FROM GTW DEPT WHERE DEPTNO=@dno
go
```

Demonstration Table Definitions

This topic discusses demonstration table definitions for Oracle Database Gateway for Sybase.

Demonstration Table Contents

This topic discusses demonstration table contents for Oracle Database Gateway for Sybase.

3.4.1 Demonstration Table Definitions

This topic discusses demonstration table definitions for Oracle Database Gateway for Sybase.

The table definitions are listed in the following table using information retrieved by the SQL*PLUS DESCRIBE command:

GTW_EMP

Name N		1?	Type	
EMPNO	NOT	NULL	NUMBER(5)	
ENAME			VARCHAR2(10)	
JOB			VARCHAR2(9)	
MGR			NUMBER (5)	
HIREDATE			DATE	
SAL			NUMBER(7,2)	
COMM			NUMBER(7,2)	
DEPTNO			NUMBER (5)	

GTW_DEPT

me Null?		Type	
DEPTNO	NOT NULL	NUMBER (5)	
DNAME		VARCHAR2 (14)	
LOC		VARCHAR2(13)	

GTW_SALGRADE

Name	Null?	Type
GRADE		NUMBER(19,4)
LOSAL		NUMBER(9,4)
HISAL		NUMBER (9,4)

3.4.2 Demonstration Table Contents

This topic discusses demonstration table contents for Oracle Database Gateway for Sybase.

The contents of the Sybase tables are:

GTW_EMP

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESPERSON	7698	20-FEB-81	1600	300	30
7521	WARD	SALESPERSON	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESPERSON	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	09-DEC-82	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESPERSON	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	12-JAN-83	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

GTW_DEPT

DEPTNO DNAME		LOC	
10	ACCOUNTING	NEW	YORK
20	RESEARCH	DALLAS	



30	SALES	CHICAGO
40	OPERATIONS	BOSTON

GTW_SALGRADE

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

3.5 Case 1: Simple Queries

Demonstration using simple queries.

Case 1 demonstrates the following:

- A simple query
- A simple query retrieving full date information

The first query retrieves all the data from <code>GTW_DEPT</code> and confirms that the gateway is working correctly. The second query retrieves all the data from <code>GTW_EMP</code> including the time portion of the hire date because the default date format was set to <code>DD-MON-YY HH24:MM:SS</code> for the session by an <code>ALTER SESSION</code> command.

3.6 Case 2: A More Complex Query

Demonstration using a more complex query.

Case 2 demonstrates the following:

- The functions SUM(expression) and NVL(expr1, expr2) in the SELECT list
- The GROUP BY and HAVING clauses

This query retrieves the departments from GTW_EMP whose total monthly expenses are higher than \$10,000.

3.7 Case 3: Joining Sybase Tables

Demonstration joining Sybase tables.

Case 3 demonstrates the following:

- Joins between Sybase tables
- Subselects

The query retrieves information from three Sybase tables and relates the employees to their department name and salary grade, but only for those employees earning more than the average salary.

3.8 Case 4: Write Capabilities

Demonstration using write capabilities and the DELETE, UPDATE, and INSERT statements.

Case 4 is split into three cases and demonstrates the following:

DELETE Statement

Demonstration using write capabilities and the DELETE statement.

UPDATE Statement

Demonstration using write capabilities and the UPDATE statement.

INSERT Statement

Demonstration using write capabilities and the INSERT statement.

3.8.1 DELETE Statement

Demonstration using write capabilities and the DELETE statement.

Case 4a demonstrates bind values and subselect. All employees in department 20 and one employee, WARD, in department 30 are deleted.

3.8.2 UPDATE Statement

Demonstration using write capabilities and the UPDATE statement.

Case 4b provides an example of a simple $\tt UPDATE$ statement. In this example, employees are given a \$100 a month salary increase.

3.8.3 INSERT Statement

Demonstration using write capabilities and the INSERT statement.

Case 4c is an example of a simple insert statement that does not provide information for all columns.

3.9 Case 5: Data Dictionary Query

Demonstration using a data dictionary query.

Case 5 demonstrates data dictionary mapping. It retrieves all the tables and views that exist in the Sybase database that begin with "GTW".

3.10 Case 6: The Pass-Through Feature

Demonstration using the pass-through feature.

Case 6 demonstrates the gateway pass-through feature that allows an application to send commands or statements to Sybase.

This case demonstrates:

- A pass-through UPDATE statement using bind variables
- A pass-through SELECT statement
- UPDATE Statement

Demonstration using the pass-through feature and the UPDATE statement.



SELECT Statement

Demonstration using the pass-through feature and the SELECT statement.

3.10.1 UPDATE Statement

Demonstration using the pass-through feature and the UPDATE statement.

Case 6a provides an example of a pass-through UPDATE statement with bind variables. In this example, the salary for EMPNO 7934 is set to 4000.

3.10.2 SELECT Statement

Demonstration using the pass-through feature and the SELECT statement.

Case 6b provides an example of a pass-through SELECT statement. The data that is returned from the SELECT statement is inserted into a local table at the Oracle database.

3.11 Case 7: Executing Stored Procedures

Demonstration executing stored procedures.

Case 7 demonstrates the gateway executing a stored procedure in the Sybase database.

Before running the script, make sure that chained mode is set on, by issuing the following command in Sybase:

set chained on





Data Type Conversion

The gateway converts Sybase data types to Oracle data types.

For example:

Table A-1 Data Type Mapping and Restrictions

Sybase	Oracle	Comment	If Oracle uses large varchar (32k)
BIGINT	NUMBER (19)	-	-
BINARY	RAW	-	-
BIT	NUMBER(3)	-	-
CHAR	CHAR	-	-
DATE	DATE	-	-
DATETIME	DATE	-	-
DECIMAL	NUMBER(p[,s])	-	-
FLOAT	FLOAT (53)	-	-
IMAGE	LONG RAW	-	-
INT	NUMBER(10)	NUMBER range is -2,147,483,647 to 2,147,483,647	-
MONEY	NUMBER(p[,s])	-	-
NUMERIC	NUMBER(p[,s])	-	-
REAL	FLOAT(24)	-	-
SMALLDATETIME	DATE	-	-
SMALLINT	NUMBER(5)	NUMBER range is -32,767 to 32,767	-
SMALLMONEY	NUMBER(p[,s])	-	-
SYSNAME	VARCHAR2	-	-
TEXT	LONG	-	-
TIME	CHAR(15)	-	-
TIMESTAMP	RAW	-	-
TINYINT	NUMBER(3)	-	-
UNICHAR	NCHAR	-	-
UNSIGNED BIGINT	NUMBER (20)	-	-
UNSIGNED INT	NUMBER (10)	-	-
UNSIGNED SMALLINT	NUMBER(5)	-	-



Table A-1 (Cont.) Data Type Mapping and Restrictions

Sybase	Oracle	Comment	If Oracle uses large varchar (32k)
UNITEXT	LONG	if Oracle DB Character Set = Unicode. Otherwise, it is not supported	· -
UNIVARCHAR	NVARCHAR	-	-
VARBINARY	RAW	-	-
VARCHAR	VARCHAR2	N < = 4000	N <= 32767



B

Supported SQL Syntax and Functions

The following topics describe supported SQL syntax and functions.

Supported SQL Statements

With a few exceptions, the gateway provides full support for Oracle DELETE, INSERT, SELECT, and UPDATE statements.

Oracle Functions

All functions are evaluated by the Sybase database after the gateway has converted them to Sybase SQL.

B.1 Supported SQL Statements

With a few exceptions, the gateway provides full support for Oracle Delete, Insert, Select, and UPDATE statements.

The gateway does not support Oracle data definition language (DDL) statements. No form of the Oracle ALTER, CREATE, DROP, GRANT, or TRUNCATE statements can be used. Instead, for ALTER, CREATE, DROP, and GRANT statements, use the pass-through feature of the gateway if you need to use DDL statements against the Sybase database.



TRUNCATE cannot be used in a pass-through statement.

DELETE

Oracle Database Gateway for Sybase fully supports the DELETE statement.

INSERT

Oracle Database Gateway for Sybase fully supports the INSERT statement.

SELECT

Oracle Database Gateway for Sybase supports the SELECT statement with a few exceptions.

UPDATE

Oracle Database Gateway for Sybase fully supports the UPDATE statement.

See Also:

Oracle Database SQL Language Reference for detailed descriptions of keywords, parameters, and options and the Sybase documentation for details of executing SQL statements in a transaction.

B.1.1 DELETE

Oracle Database Gateway for Sybase fully supports the DELETE statement.

However, only Oracle functions supported by Sybase can be used.



Functions Supported by Sybase for a list of supported functions.

B.1.2 INSERT

Oracle Database Gateway for Sybase fully supports the INSERT statement.

However, only Oracle functions supported by Sybase can be used.



Functions Supported by Sybase for a list of supported functions.

B.1.3 SELECT

Oracle Database Gateway for Sybase supports the ${\tt SELECT}$ statement with a few exceptions.

The SELECT statement is supported, with these exceptions:

- CONNECT BY condition
- NOWAIT
- START WITH condition
- WHERE CURRENT OF

B.1.4 UPDATE

Oracle Database Gateway for Sybase fully supports the UPDATE statement.

However, only Oracle functions supported by Sybase can be used. Also, you cannot have SQL statements in the subquery that refer to the same table name in the outer query. Subqueries are not supported in the ${\tt SET}$ clause.



Functions Supported by Sybase for a list of supported functions.



B.2 Oracle Functions

All functions are evaluated by the Sybase database after the gateway has converted them to Sybase SQL.

· Functions Not Supported by Sybase

Oracle SQL functions with no equivalent function in Sybase are not supported in DELETE, INSERT, or UPDATE statements, but are evaluated by the Oracle database if the statement is a SELECT statement.

Functions Supported by Sybase

The gateway translates the following Oracle database functions in SQL statements to their equivalent Sybase functions.

B.2.1 Functions Not Supported by Sybase

Oracle SQL functions with no equivalent function in Sybase are not supported in DELETE, INSERT, or UPDATE statements, but are evaluated by the Oracle database if the statement is a SELECT statement.

That is, the Oracle database performs post-processing of SELECT statements sent to the gateway.

If an unsupported function is used in a DELETE, INSERT, or UPDATE, statement, the following Oracle error occurs:

ORA-02070: database db link name does not support function in this context

B.2.2 Functions Supported by Sybase

The gateway translates the following Oracle database functions in SQL statements to their equivalent Sybase functions.

Arithmetic Operators

Arithmetic operators suppoted by Oracle Database Gateway for Sybase.

Comparison Operators

Comparison operators supported by Oracle Database Gateway for Sybase.

Group Functions

Group functions supported by Oracle Database Gateway for Sybase.

String Functions

String functions supported by Oracle Database Gateway for Sybase.

Other Functions

Other functions supported by Oracle Database Gateway for Sybase.

B.2.2.1 Arithmetic Operators

Arithmetic operators suppoted by Oracle Database Gateway for Sybase.

Oracle	Sybase
+	+



Oracle	Sybase
-	-
*	*
/	/

B.2.2.2 Comparison Operators

Comparison operators supported by Oracle Database Gateway for Sybase.

Oracle	Sybase
=	=
>	>
<	<
>=	>=
<=	<=
<>, !=, ^=	<>, !=
IS NOT NULL	IS NOT NULL
IS NULL	IS NULL

B.2.2.3 Group Functions

Group functions supported by Oracle Database Gateway for Sybase.

Oracle	Sybase
AVG	AVG
COUNT	COUNT
MAX	MAX
MIN	MIN
SUM	SUM

B.2.2.4 String Functions

String functions supported by Oracle Database Gateway for Sybase.

Oracle	Sybase
11	+ (expression1 + expression2)
ASCII	ASCII
CHR	CHAR
INSTR	CHARINDEX
LENGTH	CHAR_LENGTH
LOWER	LOWER



Oracle	Sybase
SUBSTR (second argument cannot be a negative number)	SUBSTRING
UPPER	UPPER

B.2.2.5 Other Functions

Other functions supported by Oracle Database Gateway for Sybase.

Oracle	Sybase
ABS	ABS
CEIL	CEILING
COS	COS
EXP	EXP
FLOOR	FLOOR
LN	LOG
LOG	LOG10
MOD	8
NOT NVL	IS NOT NULL
NVL	IS NULL
POWER	POWER
ROUND	ROUND
SIN	SIN
SQRT	SQRT
TAN	TAN



C

Data Dictionary

The Oracle Database Gateway for Sybase translates a query that refers to an Oracle database data dictionary table into a query that retrieves the data from Sybase system tables.

You perform queries on data dictionary tables over the database link in the same way you query data dictionary tables in the Oracle database. The gateway data dictionary is similar to the Oracle database data dictionary in appearance and use.

Sybase System Tables

Sybase data dictionary information is stored in the Sybase database as Sybase system tables.

Accessing the Gateway Data Dictionary

Accessing a gateway data dictionary table or view is identical to accessing a data dictionary in an Oracle database.

Direct Queries to Sybase Tables

Queries issued directly to individual Sybase system tables are allowed but they return different results because the Sybase system table column names differ from those of the data dictionary view.

Supported Views and Tables

The Oracle Database Gateway for Sybase supports views and tables.

Data Dictionary Mapping

The tables in this section list Oracle data dictionary view names and the equivalent Sybase system tables.

Gateway Data Dictionary Descriptions

This topic provides descriptions of the Oracle Database Gateway for Sybase data dictionaries.

ALL_CATALOG

ALL_CATALOG data dictionary for Oracle Database Gateway for Sybase.

ALL COL COMMENTS

ALL_COL_COMMENTS data dictionary for Oracle Database Gateway for Sybase.

ALL_CONS_COLUMNS

 ${\tt ALL_CONS_COLUMNS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

ALL CONSTRAINTS

 ${\tt ALL_CONSTRAINTS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

ALL IND COLUMNS

ALL IND COLUMNS data dictionary for Oracle Database Gateway for Sybase.

ALL INDEXES

ALL INDEXES data dictionary for Oracle Database Gateway for Sybase.

ALL OBJECTS

ALL OBJECTS data dictionary for Oracle Database Gateway for Sybase.

ALL TAB COLUMNS

ALL TAB COLUMNS data dictionary for Oracle Database Gateway for Sybase.

ALL_TAB_COMMENTS

ALL TAB COMMENTS data dictionary for Oracle Database Gateway for Sybase.

ALL_TABLES

ALL TABLES data dictionary for Oracle Database Gateway for Sybase.

ALL USERS

ALL USERS data dictionary for Oracle Database Gateway for Sybase.

ALL VIEWS

ALL VIEWS data dictionary for Oracle Database Gateway for Sybase.

DBA CATALOG

DBA CATALOG data dictionary for Oracle Database Gateway for Sybase.

DBA_COL_COMMENTS

DBA COL COMMENTS data dictionary for Oracle Database Gateway for Sybase.

DBA_OBJECTS

DBA_OBJECTS data dictionary for Oracle Database Gateway for Sybase.

DBA TAB COLUMNS

DBA_TAB_COLUMNS data dictionary for Oracle Database Gateway for Sybase.

DBA TAB COMMENTS

DBA TAB COMMENTS data dictionary for Oracle Database Gateway for Sybase.

DBA TABLES

DBA TABLES data dictionary for Oracle Database Gateway for Sybase.

DICT_COLUMNS

DICT COLUMNS data dictionary for Oracle Database Gateway for Sybase.

DICTIONARY

DICTIONARY data dictionary for Oracle Database Gateway for Sybase.

DUAL

DUAL data dictionary for Oracle Database Gateway for Sybase.

TABLE PRIVILEGES

TABLE PRIVILEGES data dictionary for Oracle Database Gateway for Sybase.

USER CATALOG

USER CATALOG data dictionary for Oracle Database Gateway for Sybase.

USER COL COMMENTS

USER COL COMMENTS data dictionary for Oracle Database Gateway for Sybase.

USER CONS COLUMNS

USER CONS COLUMNS data dictionary for Oracle Database Gateway for Sybase.

USER CONSTRAINTS

USER CONSTRAINTS data dictionary for Oracle Database Gateway for Sybase.

USER IND COLUMNS

USER IND COLUMNS data dictionary for Oracle Database Gateway for Sybase.

USER INDEXES

USER INDEXES data dictionary for Oracle Database Gateway for Sybase.

USER OBJECTS

USER OBJECTS data dictionary for Oracle Database Gateway for Sybase.



USER_TAB_COLUMNS

USER TAB COLUMNS data dictionary for Oracle Database Gateway for Sybase.

USER TAB COMMENTS

USER TAB COMMENTS data dictionary for Oracle Database Gateway for Sybase.

USER_TABLES

USER TABLES data dictionary for Oracle Database Gateway for Sybase.

USER USERS

USER USERS data dictionary for Oracle Database Gateway for Sybase.

USER VIEWS

USER VIEWS data dictionary for Oracle Database Gateway for Sybase.

C.1 Sybase System Tables

Sybase data dictionary information is stored in the Sybase database as Sybase system tables.

All Sybase system tables have names prefixed with "sys". The Sybase system tables define the structure of a database. When you change data definitions, Sybase reads and modifies the Sybase system tables to add information about the user tables.

C.2 Accessing the Gateway Data Dictionary

Accessing a gateway data dictionary table or view is identical to accessing a data dictionary in an Oracle database.

You issue a SQL SELECT statement specifying a database link. The Oracle database data dictionary view and column names are used to access the gateway data dictionary in an Oracle database. Synonyms of supported views are also acceptable. For example, the following statement queries the data dictionary table ALL_CATALOG to retrieve all table names in the Sybase database:

```
SQL> SELECT * FROM "ALL CATALOG"@SYBS;
```

When a data dictionary access query is issued, the gateway:

- 1. Maps the requested table, view, or synonym to one or more Sybase system table names. The gateway translates all data dictionary column names to their corresponding Sybase column names within the query. If the mapping involves one Sybase system table, the gateway translates the requested table name to its corresponding Sybase system table name within the query. If the mapping involves multiple Sybase system tables, the gateway constructs a join in the query using the translated Sybase system table names.
- 2. Sends the translated query to Sybase.
- 3. Might convert the retrieved Sybase data to give it the appearance of the Oracle database data dictionary table.
- **4.** Passes the data dictionary information from the translated Sybase system table to the Oracle database.





The values returned when querying the gateway data dictionary might not be the same as the ones returned by the Oracle SQL*Plus DESCRIBE command.

C.3 Direct Queries to Sybase Tables

Queries issued directly to individual Sybase system tables are allowed but they return different results because the Sybase system table column names differ from those of the data dictionary view.

Also, certain columns in an Sybase system table cannot be used in data dictionary processing.

C.4 Supported Views and Tables

The Oracle Database Gateway for Sybase supports views and tables.

For example:

Supported Views and Tables	Supported Views and Tables
ALL_CATALOG	ALL_COL_COMMENTS
ALL_CONS_COLUMNS	ALL_CONSTRAINTS
ALL_IND_COLUMNS	ALL_INDEXES
ALL_OBJECTS	ALL_TAB_COLUMNS
ALL_TAB_COMMENTS	ALL_TABLES
ALL_USERS	ALL_VIEWS
DBA_CATALOG	DBA_COL_COMMENTS
DBA_OBJECTS	DBA_TAB_COLUMNS
DBA_TAB_COMMENTS	DBA_TABLES
DICT_COLUMNS	DICTIONARY
DUAL	TABLE_PRIVILEGES
USER_CATALOG	USER_COL_COMMENTS
USER_CONS_COLUMNS	USER_CONSTRAINTS
USER_IND_COLUMNS	USER_INDEXES
USER_OBJECTS	USER_TAB_COLUMNS
USER_TAB_COMMENTS	USER_TABLES
USER_USERS	USER_VIEWS

No other Oracle database data dictionary tables or views are supported. If you use a view not on the list, you receive the Oracle database error code for no more rows available.



Queries through the gateway of any data dictionary table or view beginning with ALL_can returns rows from the Sybase database even when access privileges for those Sybase objects have not been granted. When querying an Oracle database with the Oracle data dictionary, rows are returned only for those objects you are permitted to access.

C.5 Data Dictionary Mapping

The tables in this section list Oracle data dictionary view names and the equivalent Sybase system tables.

A plus sign (+) indicates that a join operation is involved.

Table C-1 Oracle Data Dictionary View Names and Sybase Equivalents

View Name	Sybase System Table Name
ALL_CATALOG	sysusers + sysobjects
ALL_COL_COMMENTS	sysusers + sysobjects + syscolumns
ALL_CONS_COLUMNS	sp_pkeys + sp_fkeys
ALL_CONSTRAINTS	sysconstraints + sysusers + sysobjects
ALL_IND_COLUMNS	sysusers + sysindexes + syscolumns
ALL_INDEXES	sysusers + sysindexes + sysobjects
ALL_OBJECTS	sysusers + sysobjects + sysindexes
ALL_TAB_COLUMNS	sysusers + sysobjects + syscolumns
ALL_TAB_COMMENTS	sysusers + sysobjects
ALL_TABLES	sysusers + sysobjects
ALL_USERS	sysusers
ALL_VIEWS	sysusers + sysobjects + syscomments
DBA_CATALOG	sysusers + sysobjects
DBA_COL_COMMENTS	sysusers + sysobjects + syscolumns
DBA_OBJECTS	sysusers + sysobjects + sysindexes
DBA_TABLES	sysusers + sysobjects
DBA_TAB_COLUMNS	sysusers + sysobjects + syscolumns
DBA_TAB_COMMENTS	sysusers + sysobjects
DICT_COLUMNS	sysobjects + syscolumns
DICTIONARY	sysobjects
DUAL	(Defined in the Gateway)
TABLE_PRIVILEGES	sysprotects + sysusers + sysobjects
JSER_CATALOG	sysusers + sysobjects
JSER_COL_COMMENTS	sysusers + sysobjects + syscolumns
JSER_CONS_COLUMNS	sp_pkeys + sp_fkeys
JSER_CONSTRAINTS	sysconstraints + sysusers + sysobjects
USER_IND_COLUMNS	sysusers + sysindexes + syscolumns
JSER_INDEXES	sysusers + sysindexes + sysobjects



Table C-1 (Cont.) Oracle Data Dictionary View Names and Sybase Equivalents

View Name	Sybase System Table Name
USER_OBJECTS	sysusers + sysobjects + sysindexes
USER_TAB_COLUMNS	sysusers + sysobjects + syscolumns
USER_TAB_COMMENTS	sysusers + sysobjects
USER_TABLES	sysusers + sysobjects
USER_USERS	sysusers
USER_VIEWS	sysusers + sysobjects + syscomments

Default Column Values

There is a minor difference between the gateway data dictionary and a typical Oracle database data dictionary.

C.5.1 Default Column Values

There is a minor difference between the gateway data dictionary and a typical Oracle database data dictionary.

The Oracle database columns that are missing in an Sybase system table are filled with zeros, spaces, null values, not-applicable values (N.A.), or default values, depending on the column type.

C.6 Gateway Data Dictionary Descriptions

This topic provides descriptions of the Oracle Database Gateway for Sybase data dictionaries.

The gateway data dictionary tables and views provide the following information:

- Name, data type, and width of each column
- The contents of columns with fixed values

They are described here with information retrieved by an Oracle SQL*Plus DESCRIBE command. The values in the Null? column might differ from the Oracle database data dictionary tables and views. Any default value is shown to the right of an item, but this is not information returned by DESCRIBE.

C.7 ALL CATALOG

ALL CATALOG data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(5)	"TABLE" or "VIEW"



C.8 ALL_COL_COMMENTS

 ${\tt ALL_COL_COMMENTS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
COMMENTS	CHAR(1)	" "

C.9 ALL_CONS_COLUMNS

ALL CONS COLUMNS data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OWNER	VARCHAR2(30)	-
CONSTRAINT_NAME	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(8192)	-
POSITION	FLOAT(49)	-

C.10 ALL_CONSTRAINTS

ALL CONSTRAINTS data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OWNER	VARCHAR2(30)	-
CONSTRAINT_NAME	VARCHAR2(30)	-
CONSTRAINT_TYPE	VARCHAR2(1)	"R" or "P"
TABLE_NAME	VARCHAR2(30)	-
SEARCH_CONDITION	VARCHAR2(1)	NULL
R_OWNER	VARCHAR2(30)	-
R_CONSTRAINT_NAME	VARCHAR2(30)	-
DELETE_RULE	VARCHAR2(9)	"NO ACTION" or " "
STATUS	VARCHAR2(7)	"ENABLED"
DEFERRABLE	VARCHAR2(1)	NULL
DEFERRED	VARCHAR2(1)	NULL
VALIDATED	VARCHAR2(1)	NULL
GENERATED	VARCHAR2(1)	NULL
BAD	VARCHAR2(1)	NULL



Name	Туре	Value
RELY	VARCHAR2(1)	NULL
LAST_CHANGE	DATE	-

C.11 ALL_IND_COLUMNS

 ${\tt ALL_IND_COLUMNS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
INDEX_OWNER	VARCHAR2(30)	-
INDEX_NAME	VARCHAR2(30)	-
TABLE_OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(8192)	-
COLUMN_POSITION	FLOAT(49)	-
COLUMN_LENGTH	FLOAT(49)	-
DESCEND	VARCHAR(4)	"DESC" or "ASC"

C.12 ALL_INDEXES

 ${\tt ALL} \ \ {\tt INDEXES} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
INDEX_NAME	VARCHAR2(30)	-
INDEX_TYPE	VARCHAR2(1)	NULL
TABLE_OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(7)	"TABLE" or "CLUSTER"
UNIQUENESS	VARCHAR2(1)	NULL
COMPRESSION	VARCHAR2(1)	NULL
PREFIX_LENGTH	NUMBER	0
TABLESPACE_NAME	VARCHAR2(1)	NULL
INI_TRANS	NUMBER	0
MAX_TRANS	NUMBER	0
INITIAL_EXTENT	NUMBER	0
NEXT_EXTENT	NUMBER	0
MIN_EXTENTS	NUMBER	0
MAX_EXTENTS	NUMBER	0



Name	Туре	Value
PCT_INCREASE	NUMBER	0
PCT_THRESHOLD	NUMBER	0
INCLUDE_COLUMN	NUMBER	0
FREELISTS	NUMBER	0
FREELIST_GROUPS	NUMBER	0
PCT_FREE	NUMBER	0
LOGGING	VARCHAR2(1)	NULL
BLEVEL	NUMBER	0
LEAF_BLOCKS	NUMBER	0
DISTINCT_KEYS	NUMBER	0
AVG_LEAF_BLOCKS_PER_KEY	NUMBER	0
AVG_DATA_BLOCKS_PER_KEY	NUMBER	0
CLUSTERING_FACTOR	NUMBER	0
STATUS	VARCHAR2(1)	NULL
NUM_ROWS	NUMBER	0
SAMPLE_SIZE	NUMBER	0
LAST_ANALYZED	DATE	NULL
DEGREE	VARCHAR2(1)	NULL
INSTANCES	VARCHAR2(1)	NULL
PARTITIONED	VARCHAR2(1)	NULL
TEMPORARY	VARCHAR2(1)	NULL
GENERATED	VARCHAR2(1)	NULL
SECONDARY	VARCHAR2(1)	NULL
BUFFER_POOL	VARCHAR2(1)	NULL
USER_STATS	VARCHAR2(1)	NULL
DURATION	VARCHAR2(1)	NULL
PCT_DIRECT_ACCESS	NUMBER	0
ITYP_OWNER	VARCHAR2(1)	NULL
ITYP_NAME	VARCHAR2(1)	NULL
PARAMETERS	VARCHAR2(1)	NULL
GLOBAL_STATS	VARCHAR2(1)	NULL
DOMIDX_STATUS	VARCHAR2(1)	NULL
DOMIDX_OPSTATUS	VARCHAR2(1)	NULL
FUNCIDX_STATUS	VARCHAR2(1)	NULL

C.13 ALL_OBJECTS

 ${\tt ALL_OBJECTS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$



Name	Туре	Value
OWNER	VARCHAR2(30)	-
OBJECT_NAME	VARCHAR2(30)	-
SUBOBJECT_NAME	VARCHAR2(1)	NULL
OBJECT_ID	NUMBER	-
DATA_OBJECT_ID	NUMBER	0
OBJECT_TYPE	VARCHAR2(9)	"TABLE" or "VIEW" or "PROCEDURE"
CREATED	DATE	-
LAST_DDL_TIME	DATE	-
TIMESTAMP	VARCHAR2(1)	NULL
STATUS	VARCHAR2(5)	"VALID"
TEMPORARY	VARCHAR2(1)	NULL
GENERATED	VARCHAR2(1)	NULL
SECONDARY	VARCHAR2(1)	NULL

C.14 ALL_TAB_COLUMNS

 ${\tt ALL_TAB_COLUMNS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
DATA_TYPE	VARCHAR2(8)	-
DATA_TYPE_MOD	VARCHAR2(1)	NULL
DATA_TYPE_OWNER	VARCHAR2(1)	NULL
DATA_LENGTH	NUMBER	-
DATA_PRECISION	NUMBER	-
DATA_SCALE	NUMBER	-
NULLABLE	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	NUMBER	-
DEFAULT_LENGTH	NUMBER	0
DATA_DEFAULT	VARCHAR2(1)	NULL
NUM_DISTINCT	NUMBER	0
LOW_VALUE	NUMBER	0
HIGH_VALUE	NUMBER	0
DENSITY	NUMBER	0
NUM_NULLS	NUMBER	0



Name	Туре	Value
NUM_BUCKETS	NUMBER	0
LAST_ANALYZED	DATE	NULL
SAMPLE_SIZE	NUMBER	0
CHARACTER_SET_NAME	VARCHAR2(1)	NULL
CHAR_COL_DECL_LENGTH	NUMBER	0
GLOBAL_STATS	VARCHAR2(1)	NULL
USER_STATS	VARCHAR2(1)	NULL
AVG_COL_LEN	NUMBER	0

C.15 ALL_TAB_COMMENTS

 ${\tt ALL_TAB_COMMENTS} \ \ \textbf{data} \ \ \textbf{dictionary} \ \ \textbf{for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(5)	"TABLE" or "VIEW"
COMMENTS	VARCHAR2(1)	NULL

C.16 ALL_TABLES

 ${\tt ALL_TABLES} \ \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
OWNER	VARCHAR2 (30)	-
TABLE NAME	VARCHAR2 (30)	-
- TABLESPACE NAME	CHAR(1)	и и
- CLUSTER NAME	CHAR(1)	и и
OT NAME	CHAR(1)	" "
- PCT FREE	NUMBER(10)	0
- PCT_USED	NUMBER(10)	0
 INI_TRANS	NUMBER(10)	0
MAX_TRANS	NUMBER(10)	0
INITIAL_EXTENT	NUMBER(10)	0
NEXT_EXTENT	NUMBER(10)	0
MIN_EXTENTS	NUMBER(10)	0
MAX_EXTENTS	NUMBER(10)	0
PCT_INCREASE	NUMBER(10)	0



Name	Туре	Value
FREELISTS	NUMBER(10)	0
FREELIST_GROUPS	NUMBER (10)	0
LOGGING	CHAR(1)	11 11
BACKED_UP	CHAR(1)	11 11
NUM_ROWS	NUMBER(10)	0
BLOCKS	NUMBER (10)	0
EMPTY_BLOCKS	NUMBER (10)	0
AVG_SPACE	NUMBER (10)	0
CHAIN_CNT	NUMBER (10)	0
AVG_ROW_LEN	NUMBER (10)	0
AVG_SPACE_FREELIST_BLOCKS	NUMBER (10)	0
NUM_FREELIST_BLOCKS	NUMBER (10)	0
DEGREE	CHAR(1)	11 11
INSTANCES	CHAR(1)	11 11
CACHE	CHAR(1)	11 11
TABLE_LOCK	CHAR(1)	11 11
SAMPLE_SIZE	NUMBER (10)	0
LAST_ANALYZED	DATE	NULL
PARTITIONED	CHAR(1)	11 11
IOT_TYPE	CHAR(1)	11 11
TEMPORARY	CHAR(1)	11 11
SECONDARY	CHAR(1)	11 11
NESTED	CHAR(1)	11 11
BUFFER_POOL	CHAR(1)	11 11
ROW_MOVEMENT	CHAR(1)	11 11
GLOBAL_STATS	CHAR(1)	н н
USER_STATS	CHAR(1)	н н
DURATION	CHAR(1)	н н
SKIP_CORRUPT	CHAR(1)	н н
MONITORING	CHAR(1)	11 11

C.17 ALL_USERS

 ${\tt ALL_USERS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
USERNAME	VARCHAR2(30)	-
USER_ID	NUMBER (10)	-



Name	Туре	Value
CREATED	DATE	SYSDATE

C.18 ALL_VIEWS

 ${\tt ALL} \ \ {\tt VIEWS} \ \, {\tt data} \ \, {\tt dictionary} \ \, {\tt for} \ \, {\tt Oracle} \ \, {\tt Database} \ \, {\tt Gateway} \ \, {\tt for} \ \, {\tt Sybase}.$

Name	Type	Value
		value
OWNER	CHAR(30)	-
VIEW_NAME	CHAR (30)	-
TEXT_LENGTH	NUMBER (10)	0
TEXT	VARCHAR2 (32512)	-
TYPE_TEXT_LENGTH	NUMBER(10)	0
TYPE_TEXT	CHAR(1)	пп
OID_TEXT_LENGTH	NUMBER(10)	0
OID_TEXT	CHAR(1)	п п
VIEW_TYPE_OWNER	CHAR(1)	п п
VIEW_TYPE	CHAR(1)	и и

C.19 DBA_CATALOG

DBA CATALOG data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(5)	"TABLE" or "VIEW"

C.20 DBA_COL_COMMENTS

DBA COL COMMENTS data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
COMMENTS	CHAR (1)	н н



C.21 DBA_OBJECTS

 ${\tt DBA_OBJECTS} \ \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
OBJECT_NAME	VARCHAR2(30)	-
SUBOBJECT_NAME	VARCHAR2(1)	NULL
OBJECT_ID	NUMBER	-
DATA_OBJECT_ID	NUMBER	0
OBJECT_TYPE	VARCHAR2(9)	"TABLE" or "VIEW" or "PROCEDURE"
CREATED	DATE	-
LAST_DDL_TIME	DATE	-
TIMESTAMP	VARCHAR2(1)	NULL
STATUS	VARCHAR2(5)	NULL
TEMPORARY	VARCHAR2(1)	NULL
GENERATED	VARCHAR2(1)	NULL
SECONDARY	VARCHAR2(1)	NULL

C.22 DBA_TAB_COLUMNS

 ${\tt DBA_TAB_COLUMNS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
DATA_TYPE	VARCHAR2(8)	-
DATA_TYPE_MOD	VARCHAR2(1)	NULL
DATA_TYPE_OWNER	VARCHAR2(1)	NULL
DATA_LENGTH	NUMBER	-
DATA_PRECISION	NUMBER	-
DATA_SCALE	NUMBER	-
NULLABLE	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	NUMBER	-
DEFAULT_LENGTH	NUMBER	0
DATA_DEFAULT	VARCHAR2(1)	NULL
NUM_DISTINCT	NUMBER	0
LOW VALUE	NUMBER	0



Name	Туре	Value
HIGH_VALUE	NUMBER	0
DENSITY	NUMBER	0
NUM_NULLS	NUMBER	0
NUM_BUCKETS	NUMBER	0
LAST_ANALYZED	DATE	NULL
SAMPLE_SIZE	NUMBER	0
CHARACTER_SET_NAME	VARCHAR2(1)	NULL
CHAR_COL_DEC_LENGTH	NUMBER	0
GLOBAL_STATS	VARCHAR2(1)	NULL
USER_STATS	VARCHAR2(1)	NULL
AVG_COL_LEN	NUMBER	0

C.23 DBA_TAB_COMMENTS

 ${\tt DBA_TAB_COMMENTS} \ \ \textbf{data} \ \ \textbf{dictionary} \ \ \textbf{for Oracle Database Gateway} \ \textbf{for Sybase}.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(5)	"TABLE" or "VIEW"
COMMENTS	VARCHAR2(1)	NULL

C.24 DBA_TABLES

 ${\tt DBA_TABLES} \ \ \textbf{data} \ \ \textbf{dictionary} \ \ \textbf{for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLESPACE_NAME	CHAR(1)	" "
CLUSTER_NAME	CHAR(1)	" "
IOT_NAME	CHAR(1)	" "
PCT_FREE	NUMBER(10)	0
PCT_USED	NUMBER(10)	0
INI_TRANS	NUMBER(10)	0
MAX_TRANS	NUMBER(10)	0
INITIAL_EXTENT	NUMBER(10)	0
NEXT_EXTENT	NUMBER(10)	0



Name	Туре	Value
MIN_EXTENTS	NUMBER(10)	0
MAX_EXTENTS	NUMBER(10)	0
PCT_INCREASE	NUMBER(10)	0
FREELISTS	NUMBER(10)	0
FREELIST_GROUPS	NUMBER(10)	0
LOGGING	CHAR(1)	11 11
BACKED_UP	CHAR(1)	" "
NUM_ROWS	NUMBER(10)	0
BLOCKS	NUMBER(10)	0
EMPTY_BLOCKS	NUMBER(10)	0
AVG_SPACE	NUMBER (10)	0
CHAIN_CNT	NUMBER(10)	0
AVG_ROW_LEN	NUMBER(10)	0
AVG_SPACE_FREELIST_BLOCKS	NUMBER(10)	0
NUM_FREELIST_BLOCKS	NUMBER(10)	0
DEGREE	CHAR(1)	11 11
INSTANCES	CHAR(1)	" "
CACHE	CHAR(1)	11 11
TABLE_LOCK	CHAR(1)	" "
SAMPLE_SIZE	NUMBER(10)	0
LAST_ANALYZED	DATE	NULL
PARTITIONED	CHAR(1)	" "
IOT_TYPE	CHAR(1)	" "
TEMPORARY	CHAR(1)	" "
SECONDARY	CHAR(1)	" "
NESTED	CHAR(1)	" "
BUFFER_POOL	CHAR(1)	" "
ROW_MOVEMENT	CHAR(1)	" "
GLOBAL_STATS	CHAR(1)	н
USER_STATS	CHAR(1)	11 11
DURATION	CHAR(1)	и и
SKIP_CORRUPT	CHAR(1)	11 11
MONITORING	CHAR(1)	11 11

C.25 DICT_COLUMNS

 ${\tt DICT_COLUMNS} \ \ \textbf{data \ dictionary for Oracle Database Gateway for Sybase}.$



Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
COMMENTS	CHAR (1)	н н

C.26 DICTIONARY

DICTIONARY data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
COMMENTS	CHAR(1)	и и

C.27 DUAL

DUAL data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
DUMMY	VARCHAR2(1)	"X"

C.28 TABLE_PRIVILEGES

TABLE PRIVILEGES data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
GRANTEE	VARCHAR2(30)	-
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
GRANTOR	VARCHAR2(30)	-
SELECT_PRIV	VARCHAR2(1)	"Y"
INSERT_PRIV	VARCHAR2(1)	"A"
DELETE_PRIV	VARCHAR2(1)	"Y"
UPDATE_PRIV	VARCHAR2(1)	"A"
REFERENCES_PRIV	VARCHAR2(1)	"A"
ALTER_PRIV	VARCHAR2(1)	"Y"
INDEX_PRIV	VARCHAR2(1)	"Y"
CREATED	DATE	-



C.29 USER_CATALOG

USER CATALOG data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(5)	"TABLE" or "VIEW"

C.30 USER_COL_COMMENTS

USER COL COMMENTS data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
COMMENTS	VARCHAR2(1)	NULL

C.31 USER_CONS_COLUMNS

USER CONS COLUMNS data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OWNER	VARCHAR2(30)	-
CONSTRAINT_NAME	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(4000)	-
POSITION	NUMBER	-

C.32 USER_CONSTRAINTS

 ${\tt USER_CONSTRAINTS} \ \ \textbf{data} \ \ \textbf{dictionary} \ \ \textbf{for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
OWNER	VARCHAR2(30)	-
CONSTRAINT_NAME	VARCHAR2(30)	-
CONSTRAINT_TYPE	VARCHAR2(1)	"R" or "P"
TABLE_NAME	VARCHAR2(30)	-
SEARCH_CONDITION	VARCHAR2(1)	NULL
R_OWNER	VARCHAR2(30)	-
R_CONSTRAINT_NAME	VARCHAR2(30)	-



Name	Туре	Value
DELETE_RULE	VARCHAR2(9)	"NOACTION" or " "
STATUS	VARCHAR2(7)	"ENABLED"
DEFERRABLE	VARCHAR2(1)	NULL
DEFERRED	VARCHAR2(1)	NULL
VALIDATED	VARCHAR2(1)	NULL
GENERATED	VARCHAR2(1)	NULL
BAD	VARCHAR2(1)	NULL
RELY	VARCHAR2(1)	NULL
LAST_CHANGE	DATE	-

C.33 USER_IND_COLUMNS

 ${\tt USER_IND_COLUMNS} \ \ \textbf{data dictionary for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
INDEX_NAME	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(8192)	-
COLUMN_POSITION	FLOAT(49)	-
COLUMN_LENGTH	FLOAT(49)	-
DESCEND	VARCHAR(4)	"DESC" or "ASC"

C.34 USER_INDEXES

 ${\tt USER_INDEXES} \ \ \textbf{data} \ \ \textbf{dictionary} \ \ \textbf{for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
INDEX_NAME	VARCHAR2(30)	-
INDEX_TYPE	VARCHAR2(1)	NULL
TABLE_OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(7)	"TABLE" or "CLUSTER"
UNIQUENESS	VARCHAR2(1)	NULL
COMPRESSION	VARCHAR2(1)	NULL
PREFIX_LENGTH	NUMBER	0
TABLESPACE_NAME	VARCHAR2(1)	NULL
INI_TRANS	NUMBER	0
MAX_TRANS	NUMBER	0



INITIAL_EXTENT NUMBER 0	
NEXT_EXTENT NUMBER 0	
MIN_EXTENTS NUMBER 0	
MAX_EXTENTS NUMBER 0	
PCT_INCREASE NUMBER 0	
PCT_THRESHOLD NUMBER 0	
INCLUDE_COLUMN NUMBER 0	
FREELISTS NUMBER 0	
FREELIST_GROUPS NUMBER 0	
PCT_FREE NUMBER 0	
LOGGING VARCHAR2(1) NULL	
BLEVEL NUMBER 0	
LEAF_BLOCKS NUMBER 0	
DISTINCT_KEYS NUMBER 0	
AVG_LEAF_BLOCKS_PER_KEY NUMBER 0	
AVG_DATA_BLOCKS_PER_KEY NUMBER 0	
CLUSTERING_FACTOR NUMBER 0	
STATUS VARCHAR2(1) NULL	
NUM_ROWS NUMBER 0	
SAMPLE_SIZE NUMBER 0	
LAST_ANALYZED DATE NULL	
DEGREE VARCHAR2(1) NULL	
INSTANCES VARCHAR2(1) NULL	
PARTITIONED VARCHAR2(1) NULL	
TEMPORARY VARCHAR2(1) NULL	
GENERATED VARCHAR2(1) NULL	
SECONDARY VARCHAR2(1) NULL	
BUFFER_POOL VARCHAR2(1) NULL	
USER_STATS VARCHAR2(1) NULL	
DURATION VARHCAR2(1) NULL	
PCT_DIRECT_ACCESS NUMBER 0	
ITYP_OWNER VARCHAR2(1) NULL	
ITYP_NAME VARCHAR2(1) NULL	
PARAMETERS VARCHAR2(1) NULL	
GLOBAL_STATS VARCHAR2(1) NULL	
DOMIDX_STATUS VARCHAR2(1) NULL	
DOMIDX_OPSTATUS VARCHAR2(1) NULL	



Name	Туре	Value
FUNCIDX_STATUS	VARCHAR2(1)	NULL

C.35 USER_OBJECTS

USER OBJECTS data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
OBJECT_NAME	VARCHAR2(30)	-
SUBOBJECT_NAME	VARCHAR2(1)	NULL
OBJECT_ID	NUMBER	-
DATA_OBJECT_ID	NUMBER	0
OBJECT_TYPE	VARCHAR2(9)	"TABLE" or "VIEW" or "PROCEDURE"
CREATED	DATE	-
LAST_DDL_TIME	DATE	-
TIMESTAMP	VARCHAR2(1)	NULL
STATUS	VARCHAR2(5)	"VALID"
TEMPORARY	VARCHAR2(1)	NULL
GENERATED	VARCHAR2(1)	NULL
SECONDARY	VARCHAR2(1)	NULL

C.36 USER_TAB_COLUMNS

 ${\tt USER_TAB_COLUMNS} \ \ \textbf{data dictionary for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
DATA_TYPE	VARCHAR2(8)	-
DATA_TYPE_MOD	VARCHAR2(1)	NULL
DATA_TYPE_OWNER	VARCHAR2(1)	NULL
DATA_LENGTH	NUMBER	-
DATA_PRECISION	NUMBER	-
DATA_SCALE	NUMBER	-
NULLABLE	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	NUMBER	-
DEFAULT_LENGTH	NUMBER	0
DATA_DEFAULT	VARCHAR2(1)	NULL
NUM_DISTINCT	NUMBER	0



Name	Туре	Value
LOW_VALUE	NUMBER	0
HIGH_VALUE	NUMBER	0
DENSITY	NUMBER	0
NUM_NULLS	NUMBER	0
NUM_BUCKETS	NUMBER	0
LAST_ANALYZED	DATE	NULL
SAMPLE_SIZE	NUMBER	0
CHARACTER_SET_NAME	VARCHAR2(1)	NULL
CHAR_COL_DECL_LENGTH	NUMBER	0
GLOBAL_STATS	VARCHAR2(1)	NULL
USER_STATS	VARCHAR2(1)	NULL
AVG_COL_LEN	NUMBER	0

C.37 USER_TAB_COMMENTS

 ${\tt USER_TAB_COMMENTS} \ data \ dictionary \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(5)	"TABLE" or "VIEW"
COMMENTS	VARCHAR2(1)	NULL

C.38 USER_TABLES

USER TABLES data dictionary for Oracle Database Gateway for Sybase.

Name	Туре	Value
TABLE_NAME	VARCHAR2(30)	-
TABLESPACE_NAME	VARCHAR2(1)	NULL
CLUSTER_NAME	VARCHAR2(1)	NULL
IOT_NAME	VARCHAR2(1)	NULL
PCT_FREE	NUMBER	0
PCT_USED	NUMBER	0
INI_TRANS	NUMBER	0
MAX_TRANS	NUMBER	0
INITIAL_EXTENT	NUMBER	0
NEXT_EXTENT	NUMBER	0
MIN_EXTENTS	NUMBER	0



Name	Туре	Value	
MAX_EXTENTS	NUMBER	0	
PCT_INCREASE	NUMBER	0	
FREELISTS	NUMBER	0	
FREELIST_GROUPS	NUMBER	0	
LOGGING	VARCHAR2(1)	NULL	
BACKED_UP	VARCHAR2(1)	NULL	
NUM_ROWS	NUMBER	0	
BLOCKS	NUMBER	0	
EMPTY_BLOCKS	NUMBER	0	
AVG_SPACE	NUMBER	0	
CHAIN_CNT	NUMBER	0	
AVG_ROW_LEN	NUMBER	0	
AVG_SPACE_FREELIST_BLOCKS	NUMBER	0	
NUM_FREELIST_BLOCKS	NUMBER	0	
DEGREE	VARCHAR2(1)	NULL	
INSTANCES	VARCHAR2(1)	NULL	
CACHE	VARCHAR2(1)	NULL	
TABLE_LOCK	VARCHAR2(1)	NULL	
SAMPLE_SIZE	NUMBER	0	
LAST_ANALYZED	DATE	NULL	
PARTITIONED	VARCHAR2(1)	NULL	
IOT_TYPE	VARCHAR2(1)	NULL	
TEMPORARY	VARHCAR2(1)	NULL	
SECONDARY	VARCHAR2(1)	NULL	
NESTED	VARCHAR2(1)	NULL	
BUFFER_POOL	VARCHAR2(1)	NULL	
ROW_MOVEMENT	VARCHAR2(1)	NULL	
GLOBAL_STATS	VARCHAR2(1)	NULL	
USER_STATS	VARCHAR2(1)	NULL	
DURATION	VARCHAR2(1)	NULL	
SKIP_CORRUPT	VARCHAR2(1)	NULL	
MONITORING	VARCHAR2(1)	NULL	

C.39 USER_USERS

USER_USERS data dictionary for Oracle Database Gateway for Sybase.



Name	Туре	Value
USERNAME	VARCHAR2(30)	-
USER_ID	NUMBER (5)	-
ACCOUNT_STATUS	VARCHAR2(4)	"OPEN"
LOCK_DATE	DATE	NULL
EXPIRY_DATE	DATE	NULL
DEFAULT_TABLESPACE	VARCHAR2(1)	NULL
TEMPORARY_TABLESPACE	VARCHAR2(1)	NULL
CREATED	DATE	SYSDATE
INITIAL_RSRC_CONSUMER_GROUP	VARCHAR2(1)	NULL
EXTERNAL_NAME	VARCHAR2(1)	NULL

C.40 USER_VIEWS

 ${\tt USER_VIEWS} \ \ \textbf{data} \ \ \textbf{dictionary} \ \ \textbf{for Oracle Database Gateway for Sybase}.$

Name	Туре	Value
VIEW_NAME	VARCHAR2(30)	-
TEXT_LENGTH	NUMBER	0
TEXT	VARCHAR2 (255)	-
TYPE_TEXT_LENGTH	NUMBER	0
TYPE_TEXT	VARCHAR2(1)	NULL
OID_TEXT_LENGTH	NUMBER	0
OID_TEXT	VARCHAR2(1)	NULL
VIEW_TYPE_OWNER	VARCHAR2(1)	NULL
VIEW_TYPE	VARCHAR2(1)	NULL



D

Initialization Parameters

The Oracle database initialization parameters in the init.ora file are distinct from the gateway initialization parameters.

Set the gateway parameters in the initialization parameter file using an agent-specific mechanism, or set them in the Oracle data dictionary using the <code>DBMS_HS</code> package. The gateway initialization parameter file must be available when the gateway is started.

The following topics contain a list of the gateway initialization parameters that can be set for each gateway and their description. The topics also describe the initialization parameter file syntax.

Initialization Parameter File Syntax

This topic describes using the initialization parameter files for Oracle Database Gateway for Sybase.

HS CALL NAME

HS CALL NAME initialization parameter for Oracle Database Gateway for Sybase.

HS_DB_DOMAIN

 $\verb|HS_DB_DOMAIN| initialization| parameter for Oracle Database Gateway for Sybase.$

HS_DB_INTERNAL_NAME

HS DB INTERNAL NAME initialization parameter for Oracle Database Gateway for Sybase.

HS DB NAME

HS DB NAME initialization parameter for Oracle Database Gateway for Sybase.

HS DESCRIBE CACHE HWM

 ${\tt HS_DESCRIBE_CACHE_HWM}$ initialization parameter for Oracle Database Gateway for Sybase.

HS FDS ARRAY EXEC

HS FDS ARRAY EXEC initialization parameter for Oracle Database Gateway for Sybase.

HS FDS AUTHENTICATE METHOD

HS_FDS_AUTHENTICATE_METHOD initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_CONNECT_INFO

HS FDS CONNECT INFO initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_DATE_MAPPING

HS FDS DATE MAPPING initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_ENCRYPT_SESSION

HS_FDS_ENCRYPT_SESSION initialization parameter for Oracle Database Gateway for Sybase.

HS FDS FETCH ROWS

HS_FDS_FETCH_ROWS initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_PROC_IS_FUNC

HS_FDS_PROC_IS_FUNC initialization parameter for Oracle Database Gateway for Sybase.

HS FDS QUOTE IDENTIFIER

HS_FDS_QUOTE_IDENTIFIER initialization parameter for Oracle Database Gateway for Sybase.

HS FDS RECOVERY ACCOUNT

HS_FDS_RECOVERY_ACCOUNT initialization parameter for Oracle Database Gateway for Sybase.

HS FDS RECOVERY PWD

HS_FDS_RECOVERY_PWD initialization parameter for Oracle Database Gateway for Sybase.

HS FDS REMOTE DB CHARSET

HS_FDS_REMOTE_DB_CHARSET initialization parameter for Oracle Database Gateway for Sybase.

HS FDS REPORT REAL AS DOUBLE

HS_FDS_REPORT_REAL_AS_DOUBLE initialization parameter for Oracle Database Gateway for Sybase.

HS FDS RESULTSET SUPPORT

HS_FDS_RESULTSET_SUPPORT initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_RSET_RETURN_ROWCOUNT

 ${\tt HS_FDS_RSET_RETURN_ROWCOUNT\ initialization\ parameter\ for\ Oracle\ Database\ Gateway\ for\ Sybase.}$

HS FDS SQLLEN INTERPRETATION

 ${\tt HS_FDS_SQLLEN_INTERPRETATION}$ initialization parameter for Oracle Database Gateway for Sybase.

HS FDS SUPPORT STATISTICS

HS_FDS_SUPPORT_STATISTICS initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_TIMESTAMP_MAPPING

HS_FDS_TIMESTAMP_MAPPING initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_TRACE_LEVEL

 ${\tt HS_FDS_TRACE_LEVEL} \ \ initialization \ parameter \ for \ Oracle \ Database \ Gateway \ for \ Sybase.$

HS_FDS_TRANSACTION_ISOLATION

 ${\tt HS_FDS_TRANSACTION_ISOLATION} \ \ \textbf{initialization parameter for Oracle Database} \\ \textbf{Gateway for Sybase}.$

HS FDS TRANSACTION LOG

 ${\tt HS_FDS_TRANSACTION_LOG}$ initialization parameter for Oracle Database Gateway for Sybase.

HS FDS TRUSTSTORE FILE

 ${\tt HS_FDS_TRUSTSTORE_FILE}$ initialization parameter for Oracle Database Gateway for Sybase.



HS_FDS_TRUSTSTORE_PASSWORD

HS_FDS_TRUSTSTORE_PASSWORD initialization parameter for Oracle Database Gateway for Sybase.

HS_FDS_VALIDATE_SERVER_CERT

HS_FDS_VALIDATE_SERVER_CERT initialization parameter for Oracle Database Gateway for Sybase.

HS IDLE TIMEOUT

HS IDLE TIMEOUT initialization parameter for Oracle Database Gateway for Sybase.

• HS_KEEP_REMOTE_COLUMN_SIZE

HS_KEEP_REMOTE_COLUMN_SIZE initialization parameter for Oracle Database Gateway for Sybase.

HS_LANGUAGE

HS LANGUAGE initialization parameter for Oracle Database Gateway for Sybase.

HS LONG PIECE TRANSFER SIZE

HS_LONG_PIECE_TRANSFER_SIZE initialization parameter for Oracle Database Gateway for Sybase.

HS NLS LENGTH SEMANTICS

HS_NLS_LENGTH_SEMANTICS initialization parameter for Oracle Database Gateway for Sybase.

HS_OPEN_CURSORS

HS OPEN CURSORS initialization parameters for Oracle Database Gateway for Sybase.

HS RPC FETCH REBLOCKING

 ${\tt HS_RPC_FETCH_REBLOCKING}$ initialization parameter for Oracle Database Gateway for Sybase.

HS_RPC_FETCH_SIZE

HS RPC FETCH SIZE initialization parameter for Oracle Database Gateway for Sybase.

HS TIME ZONE

HS TIME ZONE initialization parameter for Oracle Database Gateway for Sybase.

HS TRANSACTION MODEL

HS TRANSACTION MODEL initialization parameter for Oracle Database Gateway for Sybase.

IFILE

IFILE initializaation parameter.

D.1 Initialization Parameter File Syntax

This topic describes using the initialization parameter files for Oracle Database Gateway for Sybase.

The syntax for the initialization parameter file is as follows:

- The file is a sequence of commands.
- Each command should start on a separate line.
- End of line is considered a command terminator (unless escaped with a backslash).
- If there is a syntax error in an initialization parameter file, none of the settings take effect.
- Set the parameter values as follows:

[SET][PRIVATE] parameter=value



Where:

parameter is an initialization parameter name. It is a string of characters starting with a letter and consisting of letters, digits and underscores. Initialization parameter names are case sensitive.

value is the initialization parameter value. It is case sensitive. An initialization parameter value is either:

- A string of characters that does not contain any backslashes, white space or double quotation marks (")
- A quoted string beginning with a double quotation mark and ending with a double quotation mark. The following can be used inside a quoted string:
 - backslash (\) is the escape character
 - * \n inserts a new line
 - * \t inserts a tab
 - * \" inserts a double quotation mark
 - * \\ inserts a backslash

A backslash at the end of the line continues the string on the next line. If a backslash precedes any other character then the backslash is ignored.

For example, to enable tracing for an agent, set the <code>HS_FDS_TRACE_LEVEL</code> initialization parameter as follows:

```
HS FDS TRACE LEVEL=ON
```

SET and PRIVATE are optional keywords. You cannot use either as an initialization parameter name. Most parameters are needed only as initialization parameters, so you usually do not need to use the SET or PRIVATE keywords. If you do not specify either SET or PRIVATE, the parameter is used only as an initialization parameter for the agent.

SET specifies that, in addition to being used as an initialization parameter, the parameter value is set as an environment variable for the agent process. Use SET for parameter values that the drivers or non-Oracle system need as environment variables.

PRIVATE specifies that the initialization parameter should be private to the agent and should not be uploaded to the Oracle database. Most initialization parameters should not be private. If, however, you are storing sensitive information like a password in the initialization parameter file, then you may not want it uploaded to the server because the initialization parameters and values are not encrypted when uploaded. Making the initialization parameters private prevents the upload from happening and they do not appear in dynamic performance views. Use PRIVATE for the initialization parameters only if the parameter value includes sensitive information such as a user name or password.

SET PRIVATE specifies that the parameter value is set as an environment variable for the agent process and is also private (not transferred to the Oracle database, not appearing in dynamic performance views or graphical user interfaces).

D.2 HS_CALL_NAME

HS CALL NAME initialization parameter for Oracle Database Gateway for Sybase.



Property	Description
Default value	None
Range of values	Not applicable

Specifies the remote functions that can be referenced in SQL statements. The value is a list of remote functions and their owners, separated by semicolons, in the following format:

owner name.function name

For example:

owner1.A1; owner2.A2; owner3.A3

If an owner name is not specified for a remote function, the default owner name becomes the user name used to connect to the remote database (specified when the Heterogeneous Services database link is created or taken from user session if not specified in the DB link).

The entries for the owner names and the function names are case sensitive.

D.3 HS_DB_DOMAIN

HS DB DOMAIN initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	WORLD
Range of values	1 to 199 characters

Specifies a unique network sub-address for a non-Oracle system. The ${\tt HS_DB_DOMAIN}$ initialization parameter is similar to the ${\tt DB_DOMAIN}$ initialization parameter, described in the Oracle Database Reference. The ${\tt HS_DB_DOMAIN}$ initialization parameter is required if you use the Oracle Names server. The ${\tt HS_DB_NAME}$ and ${\tt HS_DB_DOMAIN}$ initialization parameters define the global name of the non-Oracle system.



The <code>HS_DB_NAME</code> and <code>HS_DB_DOMAIN</code> initialization parameters must combine to form a unique address in a cooperative server environment.

D.4 HS_DB_INTERNAL_NAME

HS DB INTERNAL NAME initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	01010101
Range of values	1 to 16 hexadecimal characters



Specifies a unique hexadecimal number identifying the instance to which the Heterogeneous Services agent is connected. This parameter's value is used as part of a transaction ID when global name services are activated. Specifying a nonunique number can cause problems when two-phase commit recovery actions are necessary for a transaction.

D.5 HS DB NAME

HS DB NAME initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	но
Range of values	1 to 8 characters

Specifies a unique alphanumeric name for the data store given to the non-Oracle system. This name identifies the non-Oracle system within the cooperative server environment. The <code>HS_DB_NAME</code> and <code>HS_DB_DOMAIN</code> initialization parameters define the global name of the non-Oracle system.

D.6 HS DESCRIBE CACHE HWM

HS_DESCRIBE_CACHE_HWM initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	100
Range of values	1 to 4000

Specifies the maximum number of entries in the describe cache used by Heterogeneous Services. This limit is known as the describe cache high water mark. The cache contains descriptions of the mapped tables that Heterogeneous Services reuses so that it does not have to re-access the non-Oracle data store.

If you are accessing many mapped tables, increase the high water mark to improve performance. Increasing the high water mark improves performance at the cost of memory usage.

D.7 HS_FDS_ARRAY_EXEC

HS FDS ARRAY EXEC initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	TRUE
Range of values	{TRUE FALSE}
Syntax	HS_FDS_ARRAY_EXEC= {TRUE FALSE}



If set to $\ TRUE$, the gateway will use array operations for insert, update, delete statements containing binds against the remote data source. The array size is determined by the value of the $\ HS\ FDS\ FETCH\ ROWS\ init$ parameter.

If set to FALSE, the gateway will not use array operations for insert, update, and delete statements. Instead, a single statement will be issued for every value.

D.8 HS_FDS_AUTHENTICATE_METHOD

HS_FDS_AUTHENTICATE_METHOD initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	CLEARTEXT
Range of values	{CLEARTEXT ENCRYPT ENCRYPT_BOTH CLIENT}
Syntax	HS_FDS_AUTHENTICATE_METHOD= {CLEARTEXT ENCRYPT ENCRYPT_BOTH CLIENT}

Specifies the way in which user ID and password are sent to the remote DB2 server and authenticated. Valid values are:

- CLEARTEXT: user ID and password are sent in clear text to server (default).
- ENCRYPT: password is sent encrypted to server.
- ENCRYPT BOTH: user ID and password are sent encrypted to server.
- CLIENT: user ID is validated on the client side instead of by the server.

D.9 HS_FDS_CONNECT_INFO

HS FDS CONNECT INFO initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	None
Range of Values	Not applicable

 ${\tt HS_FDS_CONNECT_INFO}$ that describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for ${\tt HS}$ FDS CONNECT INFO for the gateway is as follows:

```
HS_FDS_CONNECT_INFO=host_name:port_number/database_name
```

where, <code>host_name</code> is the host name or IP address of the machine hosting the Sybase database, <code>port_number</code> is the port number of the Sybase database server, and <code>database name</code> is the Sybase database name.

This release supports IPv6 format, so you can enter IPv6 format in place of hostname, but you need to wrap square brackets around the IPv6 specification.

For example,



HS FDS CONNECT INFO=[2001:0db8:20c:f1ff:fec6:38af]:port number/...

D.10 HS_FDS_DATE_MAPPING

 ${\tt HS_FDS_DATE_MAPPING}$ initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	DATE
Range of Values	DATE CHAR
Syntax	HS_FDS_DATE_MAPPING={DATE CHAR}

If set to CHAR, then non-oracle target date would be mapped to CHAR (10). If set to DATE, then non-Oracle target date would be mapped to Oracle date.

D.11 HS_FDS_ENCRYPT_SESSION

HS_FDS_ENCRYPT_SESSION initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	NONE
Range of values	{NONE SSL NOTRUST_SSL}
Syntax	HS_FDS_ENCRYPT_SESSION = {NONE SSL NOTRUST_SSL}

Specifies the way the session to Sybase is encrypted. Valid values are:

- NONE: data transmitted between the gateway and Sybase is not encrypted. (default).
- SSL: data transmitted between the gateway and Sybase is encrypted using SSL.
- NOTRUST_SSL: This option is equivalent to the SSL setting, with initialization parameter HS FDS VALIDATE SERVER CERT = DISABLED

D.12 HS FDS FETCH ROWS

HS_FDS_FETCH_ROWS initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	100
Range of Values	Any integer between 1 and 1000
Syntax	HS_FDS_FETCH_ROWS=num

HS_FDS_FETCH_ROWS specifies the fetch array size. This is the number of rows to be fetched from the non-Oracle database and to return to Oracle database at one time.



This parameter will be affected by the $\mbox{HS_RPC_FETCH_SIZE}$ and $\mbox{HS_RPC_FETCH_REBLOCKING}$ parameters.

D.13 HS_FDS_PROC_IS_FUNC

HS_FDS_PROC_IS_FUNC initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

Enables return values from functions. By default, all stored procedures and functions do not return a return value to the user.



If you set this initialization parameter, you must change the syntax of the procedure execute statement for all existing stored procedures to handle return values.

D.14 HS FDS QUOTE IDENTIFIER

HS_FDS_QUOTE_IDENTIFIER initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	TRUE for Sybase
Range of Values	TRUE FALSE
Syntax	<pre>HS_FDS_QUOTE_IDENTIFIER = { TRUE FALSE }</pre>

HS_FDS_QUOTE_IDENTIFIER overrides the target's ability to support quote identifier depending on the value provided. The default value is TRUE for Sybase.

D.15 HS_FDS_RECOVERY_ACCOUNT

HS_FDS_RECOVERY_ACCOUNT initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	RECOVER
Range of values	Any valid user ID

Specifies the name of the recovery account used for the commit-confirm transaction model. An account with user name and password must be set up at the non-Oracle system. For more information about the commit-confirm model, see the <code>HS_TRANSACTION_MODEL</code> parameter.



The name of the recovery account is case sensitive.

D.16 HS_FDS_RECOVERY_PWD

HS_FDS_RECOVERY_PWD initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	RECOVER
Range of values	Any valid password

Specifies the password of the recovery account used for the commit-confirm transaction model set up at the non-Oracle system. For more information about the commit-confirm model, see the HS TRANSACTION MODEL parameter.

The name of the password of the recovery account is case sensitive.

D.17 HS_FDS_REMOTE_DB_CHARSET

HS_FDS_REMOTE_DB_CHARSET initialization parameter for Oracle Database Gateway for Sybase.

Property	Description	
Default Value	None	
Range of values	Not applicable	
Syntax	HS_FDS_REMOTE_DB_CHARSET	

This parameter is valid only when <code>HS_LANGUAGE</code> is set to <code>AL32UTF8</code> and the gateway runs on Windows. As more Oracle databases and non-Oracle databases use Unicode as database character sets, it is preferable to also run the gateway in Unicode character set. To do so, you must set <code>HS_LANGUAGE=AL32UTF8</code>. However, when the gateway runs on Windows, the Microsoft ODBC Driver Manager interface can exchange data only in the double-byte character set, UCS2. This results in extra ratio expansion of described buffer and column sizes. To compensate, the gateway can readjust the column size if <code>HS_FDS_REMOTE_DB_CHARSET</code> is set to the corresponding non-Oracle database character set. For example,

HS FDS REMOTE DB CHARSET=K016KSC5601.

D.18 HS_FDS_REPORT_REAL_AS_DOUBLE

HS_FDS_REPORT_REAL_AS_DOUBLE initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE



Enables Oracle Database Gateway for Sybase treat SINGLE FLOAT PRECISION fields as DOUBLE FLOAT PRECISION fields.

D.19 HS_FDS_RESULTSET_SUPPORT

HS_FDS_RESULTSET_SUPPORT initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

Enables result sets to be returned from stored procedures. By default, all stored procedures do not return a result set to the user.



If you set this initialization parameter, you must do the following:

- Change the syntax of the procedure execute statement for all existing stored procedures, to handle result sets
- Work in the sequential mode of Heterogeneous Services

D.20 HS FDS RSET RETURN ROWCOUNT

 ${\tt HS_FDS_RSET_RETURN_ROWCOUNT}$ initialization parameter for Oracle Database Gateway for Sybase.

Property	Description	
Default Value	FALSE	
Range of values	{TRUE FALSE}	
Syntax	HS_FDS_RSET_RETURN_ROWCOUNT= {TRUE FALSE}	

When set to \mathtt{TRUE} , the gateway returns the row counts of DML statements that are executed inside a stored procedure. The row count is returned as a single row, single column result set of type signed integer.

When set to FALSE, the gateway skips the row counts of DML statements that are executed inside a stored procedure. This is the default behavior, and it is the behavior of 11.1 and older gateways.

D.21 HS_FDS_SQLLEN_INTERPRETATION

HS_FDS_SQLLEN_INTERPRETATION initialization parameter for Oracle Database Gateway for Sybase.



Property	Description	
Default Value	64	
Range of values	{64 32}	
Syntax	HS_FDS_SQLLEN_INTERPRETATION= {64 32}	

This parameter is only valid for 64 bit platforms. ODBC standard specifies SQLLEN (of internal ODBC construct) being 64 bit on 64 bit platforms, but some ODBC driver managers and drivers violate this convention, and implement it as 32 bit. In order for the gateway to compensate their behavior, you need to specify HS_FDS_SQLLEN_INTERPRETATION=32 if you use these types of driver managers and driver.

D.22 HS_FDS_SUPPORT_STATISTICS

HS_FDS_SUPPORT_STATISTICS initialization parameter for Oracle Database Gateway for Sybase.

Property	Description	
Default Value	TRUE	
Range of values	{TRUE FALSE }	
Syntax	HS_FDS_SUPPORT_STATISTICS= {TRUE FALSE}	

We gather statistics from the non-Oracle database by default. You can choose to disable the gathering of remote database statistics by setting the ${\tt HS_FDS_SUPPORT_STATISTICS}$ parameter to FALSE.

D.23 HS_FDS_TIMESTAMP_MAPPING

HS_FDS_TIMESTAMP_MAPPING initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	DATE
Range of Values	CHAR DATE TIMESTAMP
Syntax	HS_FDS_TIMESTAMP_MAPPING={CHAR DATE TIMESTAMP}

If set to \mbox{CHAR} , then non-Oracle target timestamp would be mapped to \mbox{CHAR} (26). If set to \mbox{DATE} (default), then non-Oracle target timestamp would be mapped to Oracle \mbox{DATE} . If set to $\mbox{TIMESTAMP}$, then non-Oracle target timestamp would be mapped to Oracle $\mbox{TIMESTAMP}$.

D.24 HS FDS TRACE LEVEL

HS FDS TRACE LEVEL initialization parameter for Oracle Database Gateway for Sybase.



Property	Description
Default Value	OFF
Range of values	OFF, ON, DEBUG

Specifies whether error tracing is turned on or off for gateway connectivity.

The following values are valid:

- OFF disables the tracing of error messages.
- ON enables the tracing of error messages that occur when you encounter problems. The
 results are written by default to a gateway log file in LOG directory where the gateway is
 installed.
- DEBUG enables the tracing of detailed error messages that can be used for debugging.

D.25 HS_FDS_TRANSACTION_ISOLATION

 ${\tt HS_FDS_TRANSACTION_ISOLATION}\ initialization\ parameter\ for\ Oracle\ Database\ Gateway\ for\ Sybase.$

Property	Description
Default Value	READ_COMMITTED
Range of Values	{READ_UNCOMMITTED READ_COMMITTED REPEATABLE_READ SERIALIZABLE}
Syntax	<pre>HS_FDS_ISOLATION_LEVEL={{READ_UNCOMMITTED READ_COMMITTED REPEATABLE_READ SERIALIZABLE}</pre>

HS_FDS_TRANSACTION_ISOLATION specifies the isolation level that is used for the transaction that the gateway opens on the non-Oracle database.

The isolation levels of READ_UNCOMMITTED, READ_COMMITTED, REPEATABLE_READ, and SERIALIZABLE are the four isolation levels defined in the SQL standard and adopted by both ANSI and ISO/IEC. For additional information regarding them, see *Oracle Database Concepts*.

Use caution when specifying an isolation level lower than the Oracle transaction isolation level being used, as the gateway transaction will have different Preventable Read Phenomena from what will occur in the Oracle database transaction.

D.26 HS_FDS_TRANSACTION_LOG

HS FDS TRANSACTION LOG initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	HS_TRANSACTION_LOG
Range of Values	Any valid table name



Specifies the name of the table created in the non-Oracle system for logging transactions. For more information about the transaction model, see the HS_TRANSACTION_MODEL parameter.

D.27 HS_FDS_TRUSTSTORE_FILE

HS_FDS_TRUSTSTORE_FILE initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	none
Range of values	path to truststore file
Syntax	<pre>HS_FDS_TRUSTSTORE_FILE = path to truststore file</pre>

Specifies the path that specifies the location of the truststore file. The truststore file contains a list of the valid Certificate Authorities (CAs) that are trusted by the client machine for SSL server authentication.

D.28 HS_FDS_TRUSTSTORE_PASSWORD

HS_FDS_TRUSTSTORE_PASSWORD initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	none
Range of values	password
Syntax	HS_FDS_TRUSTSTORE_PASSWORD= password

Specifies the password required to access the truststore.

D.29 HS_FDS_VALIDATE_SERVER_CERT

HS_FDS_VALIDATE_SERVER_CERT initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	ENABLED
Range of values	{ENABLED DISABLED}
Syntax	<pre>HS_FDS_VALIDATE_SERVER_CERT = {ENABLED DISABLED}</pre>

Specifies whether the driver validates the certificate that is sent by the database server when SSL encryption is enabled through <code>HS_FDS_ENCRYPT_SESSION</code>. When using SSL server authentication, any certificate sent by the server must be issued by a trusted Certificate Authority. Valid values are:

ENABLED: the gateway validates the certificate that is sent by the database server.
 Any certificate from the server must be issued by a trusted Certificate Authority in

the truststore file. The truststore information is specified using the HS FDS TRUSTSTORE FILE and HS FDS TRUSTSTORE PASSWORD initialization parameters.

 DISABLED: the gateway does not validate the certificate that is sent by the database server.

D.30 HS IDLE TIMEOUT

HS IDLE TIMEOUT initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	0 (no timeout)
Range of Values	0-9999 (minutes)
Syntax	HS_IDLE_TIMEOUT=num

This feature is only available for Oracle Net TCP protocol. When there is no activity for a connected gateway session for this specified time period, the gateway session would be terminated automatically with pending update (if any) rolled back.

D.31 HS_KEEP_REMOTE_COLUMN_SIZE

 ${\tt HS_KEEP_REMOTE_COLUMN_SIZE}$ initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	OFF
Range of Values	OFF LOCAL REMOTE ALL
Syntax	HS_KEEP_REMOTE_COLUMN_SIZE = OFF LOCAL REMOTE ALL
Parameter type	String

HS_KEEP_REMOTE_COLUMN_SIZE specifies whether to suppress ratio expansion when computing the length of (VAR) CHAR datatypes during data conversion from the non-Oracle database to the gateway, and then to the Oracle database. When it is set to REMOTE, the expansion is suppressed between the non-Oracle database and the gateway. When it is set to LOCAL, the expansion is suppressed between the gateway and the Oracle database. When it is set to ALL, the expansion is suppressed from the non-Oracle database to the Oracle database.

When the parameter is set, the expansion is suppressed when reporting the remote column size, calculating the implicit resulting buffer size, and instantiating in the local Oracle database. This has effect only for remote column size from non-Oracle database to Oracle database. If the gateway runs on Windows and HS_LANGUAGE=AL32UTF8, then you must not specify this parameter, as it would influence other ratio related parameter operation. It has no effect for calculating ratio for data moving from Oracle database to non-Oracle database through gateway during INSERT, UPDATE, or DELETE.



D.32 HS_LANGUAGE

HS LANGUAGE initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	System-specific
Range of values	Any valid language name (up to 255 characters)

Provides Heterogeneous Services with character set, language, and territory information of the non-Oracle data source. The value must use the following format:

language[territory.character set]



The globalization support initialization parameters affect error messages, the data for the SQL Service, and parameters in distributed external procedures.

- Character Sets
 HS_LANGUAGE initialization parameter character sets.
- Language
 HS_LANGUAGE initialization parameter language component.
- Territory
 HS_LANGUAGE initialization parameter territory clause.

D.32.1 Character Sets

HS LANGUAGE initialization parameter character sets.

Ideally, the character sets of the Oracle database and the non-Oracle data source are the same. In almost all cases, <code>HS_LANGUAGE</code> should be set exactly the same as Oracle database character set for optimal character set mapping and performance. If they are not the same, Heterogeneous Services attempts to translate the character set of the non-Oracle data source to the Oracle database character set, and back again. The translation can degrade performance. In some cases, Heterogeneous Services cannot translate a character from one character set to another.



The specified character set must be a superset of the operating system character set on the platform where the agent is installed.

As more Oracle databases and non-Oracle databases use Unicode as database character sets, it is preferable to also run the gateway in Unicode character set. To do so, you must set HS_LANGUAGE=AL32UTF8. However, when the gateway runs on Windows, the Microsoft ODBC Driver Manager interface can exchange data only in the double-byte character set, UCS2. This results in extra ratio expansion of described



buffer and column sizes. Refer to HS_FDS_REMOTE_DB_CHARSET for instruction on how to adjust to correct sizes.

D.32.2 Language

HS_LANGUAGE initialization parameter language component.

The language component of the HS LANGUAGE initialization parameter determines:

- Day and month names of dates
- AD, BC, PM, and AM symbols for date and time
- Default sorting mechanism

Note that Oracle does not determine the language for error messages for the generic Heterogeneous Services messages (ORA-25000 through ORA-28000). These are controlled by the session settings in the Oracle database.

D.32.3 Territory

HS LANGUAGE initialization parameter territory clause.

The territory clause specifies the conventions for day and week numbering, default date format, decimal character and group separator, and ISO and local currency symbols. Note that the level of globalization support between the Oracle database and the non-Oracle data source depends on how the gateway is implemented.

D.33 HS_LONG_PIECE_TRANSFER_SIZE

 ${\tt HS_LONG_PIECE_TRANSFER_SIZE}$ initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	64 KB
Range of values	Any value up to 2 GB

Sets the size of the piece of LONG data being transferred. A smaller piece size means less memory requirement, but more round-trips to fetch all the data. A larger piece size means fewer round-trips, but more of a memory requirement to store the intermediate pieces internally. Thus, the initialization parameter can be used to tune a system for the best performance, with the best trade-off between round-trips and memory requirements, and network latency or response time.

D.34 HS_NLS_LENGTH_SEMANTICS

HS NLS LENGTH SEMANTICS initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	BYTE
Range of Values	BYTE CHAR



Property	Description
Syntax	HS_NLS_LENGTH_SEMANTICS = { BYTE CHAR }

This release of gateway has Character Semantics functionality equivalent to the Oracle Database Character Semantics, that is, NLS_LENGTH_SEMANTICS. When HS_NLS_LENGTH_SEMANTICS is set to CHAR, the (VAR) CHAR columns of Sybase database are to be interpreted as having CHAR semantics. The only situation the gateway does not honor the HS_NLS_LENGTH_SEMANTICS=CHAR setting is when both Oracle database and the gateway are on the same multi-byte character set.

D.35 HS_OPEN_CURSORS

HS OPEN CURSORS initialization parameters for Oracle Database Gateway for Sybase.

Property	Description
Default value	50
Range of values	1 to the value of Oracle's ${\tt OPEN_CURSORS}$ initialization parameter

Defines the maximum number of cursors that can be open on one connection to a non-Oracle system instance.

The value never exceeds the number of open cursors in the Oracle database. Therefore, setting the same value as the <code>OPEN_CURSORS</code> initialization parameter in the Oracle database is recommended.

D.36 HS RPC FETCH REBLOCKING

 ${\tt HS_RPC_FETCH_REBLOCKING}$ initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	ON
Range of values	OFF or ON

Controls whether Heterogeneous Services attempts to optimize performance of data transfer between the Oracle database and the Heterogeneous Services agent connected to the non-Oracle data store.

The following values are possible:

- OFF disables reblocking of fetched data so that data is immediately sent from agent to server.
- ON enables reblocking, which means that data fetched from the non-Oracle system
 is buffered in the agent and is not sent to the Oracle database until the amount of
 fetched data is equal or higher than the value of HS_RPC_FETCH_SIZE initialization
 parameter. However, any buffered data is returned immediately when a fetch
 indicates that no more data exists or when the non-Oracle system reports an error.



D.37 HS_RPC_FETCH_SIZE

HS RPC FETCH SIZE initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value	50000
Range of values	1 to 10000000

Tunes internal data buffering to optimize the data transfer rate between the server and the agent process.

Increasing the value can reduce the number of network round-trips needed to transfer a given amount of data, but also tends to increase data bandwidth and to reduce latency as measured between issuing a query and completion of all fetches for the query. Nevertheless, increasing the fetch size can increase latency for the initial fetch results of a query, because the first fetch results are not transmitted until additional data is available.

D.38 HS_TIME_ZONE

HS TIME ZONE initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default value for '[+ -]hh:mm'	Derived from the NLS_TERRITORY initialization parameter
Range of values for '[+ -]hh:mm'	Any valid datetime format mask

Specifies the default local time zone displacement for the current SQL session. The format mask, [+|-]hh:mm, is specified to indicate the hours and minutes before or after UTC (Coordinated Universal Time—formerly Greenwich Mean Time). For example:

 $HS_TIME_ZONE = [+ | -] hh:mm$

D.39 HS_TRANSACTION_MODEL

HS TRANSACTION MODEL initialization parameter for Oracle Database Gateway for Sybase.

Property	Description
Default Value	COMMIT_CONFIRM
Range of Values	COMMIT_CONFIRM, READ_ONLY, SINGLE_SITE, READ_ONLY_AUTOCOMMIT, SINGLE_SITE_AUTOCOMMIT

Specifies the type of transaction model that is used when the non-Oracle database is updated by a transaction.

The following values are possible:



- COMMIT_CONFIRM provides read and write access to the non-Oracle database and allows the gateway to be part of a distributed update. To use the commit-confirm model, the following items must be created in the non-Oracle database:
 - Transaction log table. The default table name is HS_TRANSACTION_LOG. A
 different name can be set using the HS_FDS_TRANSACTION_LOG parameter. The
 transaction log table must be granted SELECT, DELETE, and INSERT privileges
 set to public.
 - Recovery account. The account name is assigned with the HS FDS RECOVERY ACCOUNT parameter.
 - Recovery account password. The password is assigned with the HS FDS RECOVERY PWD parameter.
- READ ONLY provides read access to the non-Oracle database.
- SINGLE_SITE provides read and write access to the non-Oracle database. However, the gateway cannot participate in distributed updates.
- READ_ONLY_AUTOCOMMIT provides read only access to the non-Oracle database that does not use logging.
- SINGLE_SITE_AUTOCOMMIT provides read and write access to the non-Oracle database without logging. The gateway cannot participate in distributed updates. Moreover, any update to the non-Oracle database is committed immediately.

D.40 IFILE

IFILE initialization parameter.

Property	Description
Default value	None
Range of values	Valid parameter file names

Use the IFILE initialization parameter to embed another initialization file within the current initialization file. The value should be an absolute path and should not contain environment variables. The three levels of nesting limit do not apply.



Oracle Database Reference



Index

A	Data type (continued)
NTED 11 D 1	INTEGER, A-1
ALTER statement, <i>B-1</i>	LONG, A-1
Arithmetic operators, B-3	MONEY, A-1
	NUMERIC, A-1
В	RAW, <i>A-1</i>
	REAL, <i>A-1</i>
BINARY data type, A-1	SMALL DATETIME, A-1
BIT data type, A-1	SMALL MONEY, A-1
	SMALLINT, A-1
С	TEXT, <i>A-1</i>
	TINYINT, A-1
Case rules, 2-10	VARBINARY, <i>2-18</i> , <i>A-1</i>
Case studies, 3-1	VARCHAR, A-1
Chained mode, 2-9	VARCHAR2, <i>A-1</i>
CHAR data type, <i>A-1</i>	DATE data type, A-1
character sets	DATETIME data type, A-1
Heterogeneous Services, D-16	DDL statement, 2-15
COMMIT	DECIMAL data type, A-1
restrictions, 2-15	DELETE statement, 3-6, B-2, B-3
Commit point site, 2-15	demonstration build SQL script, 3-2
Comparison operators, <i>B-4</i>	Demonstration files, 3-2
CONCAT operator, 2-19	Demonstration tables, 3-2
CONNECT BY clause, 2-16	Demonstration tables build SQL script, 3-2
COPY command, 2-17	describe cache high water mark
CREATE statement, <i>B-1</i>	definition, D-6
Cursor loops	DROP statement, <i>B-1</i>
restrictions, 2-15	
	E
D	
	Encrypted format login, 2-18
Data definition language, B-1	Error messages
Data dictionary	error tracing, <i>D-12</i>
views, C-4	Errors
Data type	ORA-02070, <i>2-15</i>
BINARY, A-1	Executing Stored Procedures, 3-7
BIT, <i>A-1</i>	
CHAR, <i>A-1</i>	F
conversion, 2-11	•
DATE, <i>A-1</i>	fetch array size, with HS_FDS_FETCH_ROWS,
DATETIME, A-1	D-8
DECIMAL, A-1	FLOAT data type, A-1
FLOAT, A-1	Functions in SQL, 2-3
IMAGE, <i>A-1</i>	



G	parameter, <i>D-18</i>	
Gateway	HS_RPC_FETCH_SIZE initialization parameter,	
case studies, 3-1	D-19	
data dictionary tables, <i>C-1</i>	HS_TIME_ZONE initialization parameter, D-19	
pass-through feature, 2-2	,	
supported functions, B-1	1	
supported SQL syntax, B-1		
globalization support	IFILE initialization parameter, <i>D-20</i>	
Heterogeneous Services, <i>D-16</i>	IMAGE data type, A-1	
GRANT statement, <i>B-1</i>	Initialization parameter file	
Group functions, B-4	customizing, <i>D-1</i>	
	INSERT statement, 2-17, 3-6, B-2, B-3	
1.1	INTEGER data type, A-1	
Н	isolation level,	
Hotorogonous Sonicos		
Heterogeneous Services	HS_FDS_TRANSACTION_ISOLATION,	
defining maximum number of open cursors, D-18	D-13	
optimizing data transfer, <i>D-18</i>	I/	
setting global name, D-6	K	
specifying cache high water mark, D-6	Known restrictions, 2-13	
tuning internal data buffering, <i>D-19</i>	Tallowit restrictions, 2 10	
tuning LONG data transfer, D-17		
Hexadecimal notation, 2-11	L	
HS_CALL_NAME initialization parameter, D-4	Locking, database, <i>2-13</i>	
HS_DB_NAME initialization parameter, D-6	LONG data type, <i>A-1</i>	
HS_DESCRIBE_CACHE_HWM initialization	LONG data type, A-1	
parameter, D-6		
HS_FDS_CONNECT_INFO, D-7	M	
HS_FDS_FETCH_ROWS parameter, D-8	MONEY data to a A A	
HS_FDS_PROC_IS_FUNC initialization	MONEY data type, A-1	
parameter, D-9		
HS_FDS_RECOVERY_PWD initialization	N	
parameter, <i>D-10</i>		
HS_FDS_RESULTSET_SUPPORT initialization	NULL values, 2-9	
parameter, <i>D-11</i>	NUMERIC data type, A-1	
HS_FDS_TRACE_LEVEL initialization		
parameter, <i>D-12</i>	0	
enabling agent tracing, D-3		
HS_FDS_TRANSACTION_ISOLATIONparamete	Objects, naming rules, 2-10	
r, <i>D-13</i>	ORA-02070, 2-15	
HS_FDS_TRANSACTION_LOG initialization		
parameter, <i>D-13</i>	Р	
HS_IDLE_TIMEOUT initialization parameter,	'	
D-15	parameters	
HS_KEEP_REMOTE_COLUMN_SIZE	gateway initialization file	
initialization parameter, <i>D-15</i>	HS_FDS_FETCH_ROWS, <i>D-8</i>	
HS_LANGUAGE initialization parameter, <i>D-16</i>	HS_FDS_TRANSACTION_ISOLATION,	
HS_LONG_PIECE_TRANSFER_SIZE	D-13	
initialization parameter, D-17	Pass-through feature, 2-15	
HS_NLS_LENGTH_SEMANTICS initialization	Pass-Through Feature, 3-6	
parameter, <i>D-17</i>	Passing commands to database, 2-15	
HS_OPEN_CURSORS initialization parameter,	PL/SQL, 2-19	
11.79		



R	SUM function, 3-5
RAW data type, A-1 REAL data type, A-1	Т
remote HS_FDS_TRANSACTION_ISOLATION, D-13 remote functions referenced in SQL statements, D-4 ROLLBACK restrictions, 2-15 ROWID, 2-16	TEXT data type, A-1 TINYINT data type, A-1 Transaction modes, 2-9 transactional capability, 2-15 transactional integrity, 2-14 TRUNCATE statement, B-1 Two-phase commit, 2-15
S	U
	UPDATE statement, 3-6, 3-7, B-2, B-3
savepoint support, 2-15 SELECT statement, 3-7, B-2, C-3 SMALL DATETIME data type, A-1	V
SMALLINT data type, A-1 SQL statements, HS_FDS_TRANSACTION_ISOLATION, D-13	VARBINARY data type, 2-18, A-1 VARCHAR data type, A-1 VARCHAR2 data type, A-1
Stored procedures, 2-15, 2-19	W
running in chained mode, 2-9 Stored procedures in SQL Server, 2-3 String functions, B-4	WHERE CURRENT OF clause, 2-16

