Oracle® Database Gateway

for ODBC User's Guide

11*g* Release 1 (11.1)

E10311-01

April 2007

Beta Draft

Oracle Database Gateway for ODBC User's Guide, 11*g* Release 1 (11.1)

E10311-01

Contents

B Supported SQL Syntax and Functions

C Data Dictionary

D Initialization Parameters

Index

List of Tables

Preface

This manual describes the Oracle Database Gateway for ODBC, which enables Oracle client applications to access non-Oracle systems 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 ODBC
- Diagnosing gateway errors
- Using the gateway to access non-Oracle system 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

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

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

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

Related Documents

For more information, see these Oracle resources:

- *Oracle Database New Features Guide*
- *Oracle Call Interface Programmer's Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Database Advanced Application Developer's 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 Administrator'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

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italics	Italic typeface indicates book titles or emphasis.	*Oracle Database Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.
`UPPERCASE monospace (fixed-width) font`	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, data types, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a `NUMBER` column. You can back up the database by using the `BACKUP` command. Query the `TABLE_NAME` column in the `USER_TABLES` data dictionary view. Use the `DBMS_STATS.GENERATE_STATS` procedure.
`lowercase monospace (fixed-width) font`	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter `sqlplus` to open SQL*Plus. The password is specified in the `orapwd` file. Back up the datafiles and control files in the `/disk1/oracle/dbs` directory. The `department_id`, `department_name`, and `location_id` columns are in the `hr.departments` table. Set the `QUERY_REWRITE_ENABLED` initialization parameter to `true`. Connect as `oe` user. The `JRepUtil` class implements these methods.
`lowercase italic monospace (fixed-width) font`	Lowercase italic monospace font represents placeholders or variables.	You can specify the `parallel_clause`. Run `Uold_release.SQL` where `old_release` refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
`[ ]`	Brackets enclose one or more optional items. Do not enter the brackets.	`DECIMAL (digits [ , precision ])`
`{ }`	Braces enclose two or more items, one of which is required. Do not enter the braces.	`{ENABLE	DISABLE}`

Convention	Meaning	Example
|	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	`{ENABLE | DISABLE}` `[COMPRESS | NOCOMPRESS]`

...	Horizontal ellipsis points indicate either:
	
	■ That we have omitted parts of the code that are not directly related to the example
	
	■ That you can repeat a portion of the code
		`CREATE TABLE ... AS subquery;`
		
		`SELECT col1, col2, ... , coln FROM employees;`

.
.
.
	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	`SQL> SELECT NAME FROM V$DATAFILE;`
		`NAME`
		`------------------------------------`
		`/fsl/dbs/tbs_01.dbf`
		`/fs1/dbs/tbs_02.dbf`
		`.`
		`.`
		`.`
		`/fsl/dbs/tbs_09.dbf`
		`9 rows selected.`

Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	`acctbal NUMBER(11,2);` `acct    CONSTANT NUMBER(4) := 3;`
Italics	Italicized text indicates placeholders or variables for which you must supply particular values.	`CONNECT SYSTEM/system_password` `DB_NAME = database_name`

UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	`SELECT last_name, employee_id FROM employees;`
		`SELECT * FROM USER_TABLES;`
		`DROP TABLE hr.employees;`

lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files.
	
	Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.
		`SELECT last_name, employee_id FROM employees;`
		`sqlplus hr/hr`
		`CREATE USER mjones IDENTIFIED BY ty3MU9;`

1

Introduction

This chapter introduces the challenge faced by organizations when running several different database systems. It briefly covers Heterogeneous Services, the technology that the Oracle Database Gateway for ODBC is based on.

To get a good understanding of generic gateway technology, Heterogeneous Services, and how Oracle Database Gateways fit in the picture, reading the *Oracle Database Heterogeneous Connectivity Administrator's Guide* first is highly recommended.

This chapter contains the following sections:

- Overview
- Heterogeneous Services Technology
- Oracle Database Gateway for ODBC
- Oracle Database Gateway for ODBC Architecture
- ODBC Connectivity Requirements

Overview

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 non-Oracle systems.

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 Gateways, enable transparent access to non-Oracle systems from an Oracle environment.

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 which 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.

> **See Also:** *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about Heterogeneous Services.

Oracle Database Gateway for ODBC

Oracle Database Gateway for ODBC is intended for low-end data integration solutions requiring the ad hoc query capability to connect from an Oracle database to non-Oracle systems. Any data source compatible with the ODBC standards described in this chapter can be accessed using Oracle Database Gateway for ODBC.

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.

Oracle Database Gateway for ODBC Architecture

To access the non-Oracle data store using Oracle Database Gateway for ODBC, the gateway works with an ODBC driver. The driver that you use must be on the same machine as the gateway. The non-Oracle system can reside on the same machine as the Oracle database or on a different machine.

The gateway can be installed on the machine running the non-Oracle system, the machine running the Oracle 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.

> **Note:** The ODBC driver may require non-Oracle client libraries even if the non-Oracle database is located on the same machine. Refer to your ODBC driver documentation for information about the requirements for the ODBC driver.

Oracle and Non-Oracle Systems on Separate Machines

Figure 1–1 shows an example of a configuration in which an Oracle and non-Oracle database are on separate machines, communicating through Oracle Database Gateway for ODBC. The client connects to the non-Oracle system through a network.

Figure 1–1   Oracle and Non-Oracle Systems on a Separate Machines
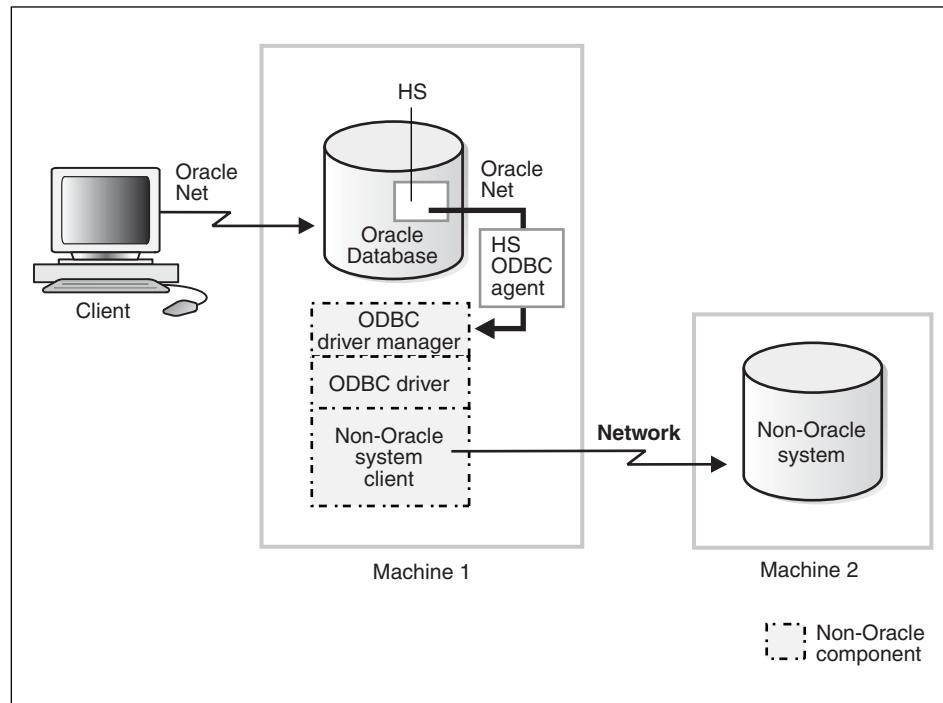
In this configuration:

1. A client connects to the Oracle database through Oracle Net.

2. The Heterogeneous Services component of the Oracle database connects through Oracle Net to the gateway.

3. The gateway communicates with the following non-Oracle components:

   ■ An ODBC driver manager

   ■ An ODBC driver

4. Each user session receives its own dedicated agent process spawned by the first use in that user session of the database link to the non-Oracle system. The agent process ends when the user session ends.

> **Note:** The ODBC driver may require non-Oracle client libraries even if the non-Oracle database is located on the same machine. Refer to your ODBC driver documentation for information about the requirements for the ODBC driver.

Oracle and Non-Oracle Systems on the Same Machine

Figure 1–2 shows an example of a configuration in which an Oracle and non-Oracle database are on the same machine, again communicating through Oracle Database Gateway for ODBC.

Figure 1–2   Oracle and Non-Oracle Systems on the Same Machine
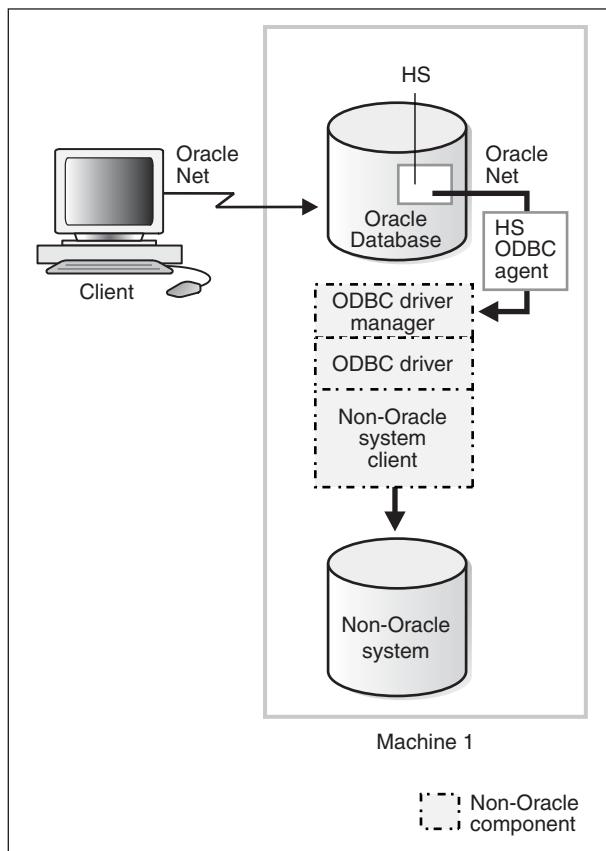
In this configuration:

1. A client connects to the Oracle database through Oracle Net.

2. The Heterogeneous Services component of the Oracle database server connects through Oracle Net to the gateway

3. The agent communicates with the following non-Oracle components:

    ■ An ODBC driver manager

    ■ An ODBC driver

    The driver then allows access to the non-Oracle data store.

4. Each user session receives its own dedicated agent process spawned by the first use in that user session of the database link to the non-Oracle system. The agent process ends when the user session ends.

> **Note:** The ODBC driver may require non-Oracle client libraries even if the non-Oracle database is located on the same machine. Refer to your ODBC driver documentation for information about the requirements for the ODBC driver.

ODBC Connectivity Requirements

To use Oracle Database Gateway for ODBC, you must have an ODBC driver installed on the same machine as the Oracle database. The ODBC driver manager and driver must meet the following requirements:

- The ODBC driver (and the non-Oracle system) must support a minimum transaction isolation level of **read committed**

- The following ODBC catalog functions must work inside a transaction for data dictionary translations to be supported:
  - `SQLColumns`
  - `SQLForeignKeys`
  - `SQLGetFunctions`
  - `SQLGetInfo`
  - `SQLGetTypeInfo`
  - `SQLPrimaryKeys`
  - `SQLProcedureColumns`
  - `SQLProcedures`
  - `SQLStatistics`
  - `SQLTables`

- On Windows:
  - The ODBC driver must have compliance level to ODBC standard 3.0. For multi-byte support, the driver needs to meet ODBC standard 3.5.
  - The ODBC driver and driver manager must conform to ODBC application program interface (API) conformance Level 1 or higher. If the ODBC driver or driver manager does not support multiple active ODBC cursors, the complexity of SQL statements that you can execute using Oracle Database Gateway for ODBC is restricted.

- On UNIX:
  - The ODBC driver manager must be installed on the same machine.
  - The ODBC driver must have compliance level to ODBC Standard 3.0 and have a conformance level 1 or higher. If the ODBC driver works with an ODBC driver manager, the ODBC driver manager must be compliant with ODBC Standard 3.0 or higher. The ODBC driver must have compliance level to ODBC standard 3.0. For multi-byte support, the driver needs to meet ODBC standard 3.5.

    See Also: Your ODBC driver documentation for dependencies on an ODBC driver manager, and *Oracle Database Concepts* for more information on transaction isolation levels.

- The ODBC driver you use must support all of the core SQL ODBC data types and must support SQL grammar level `SQL_92`. The ODBC driver should also expose the following ODBC APIs:
  - `SQLAllocHandle`
  - `SQLBindCol`

- SQLBindParameter
- SQLCancel
- SQLColAttribute
- SQLColumns
- SQLConnect
- SQLDescribeCol
- SQLDisconnect
- SQLDriverConnect
- SQLEndTran
- SQLExecDirect
- SQLExecute
- SQLFetch
- SQLForeignKeys
- SQLFreeHandle
- SQLFreeStmt
- SQLGetConnectAttr
- SQLGetData
- SQLGetDiagField
- SQLGetDiagRec
- SQLGetEnvAttr
- SQLGetFunctions
- SQLGetInfo
- SQLGetStmtAttr
- SQLGetTypeInfo
- SQLMoreResults
- SQLNumResultCols
- SQLParamData
- SQLPrepare
- SQLPrimaryKeys
- SQLProcedureColumns
- SQLProcedures
- SQLPutData
- SQLRowCount
- SQLSetConnectAttr
- SQLSetEnvAttr
- SQLSetDescField
- SQLSetDescRec

- – `SQLSetStmtAttr`
- – `SQLStatistics` - If statistics are to be supported
- – `SQLTables`

2

Oracle Database Gateway for ODBC Features and Restrictions

After the gateway is installed and configured, you can use the gateway to access data in non-oracle systems, pass native commands from applications to the non-oracle system, perform distributed queries, and copy data.

This chapter contains the following sections:

- Using the Pass-Through Feature
- Known Restrictions
- Known Problems

Using the Pass-Through Feature

The gateway can pass native commands or statements from the application to the non-oracle system using the DBMS_HS_PASSTHROUGH package.

Use the DBMS_HS_PASSTHROUGH package in a PL/SQL block to specify the statement to be passed to the non-oracle system, as follows:

```
DECLARE
    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

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

> **Note:** TRUNCATE cannot be used in a pass-through statement.

> **Note:** As a general rule it is recommended that you COMMIT after each DDL statement in the pass-through especially when going to a Sybase database.

> **See Also:** *Oracle Database PL/SQL Packages and Types Reference* and Chapter 3 of Oracle Heterogeneous Connectivity Administrator's Guide for more information about the DBMS_HS_PASSTHROUGH package.

Known Restrictions

If you encounter incompatibility problems not listed in this section or in "Known Problems" on page 2-3, please contact Oracle Support Services. The following section describes the known restrictions:

- BLOB and CLOB data cannot be read by passthrough queries
- Updates or deletes that include unsupported functions within a WHERE clause are not allowed
- Does not support stored procedures
- Cannot participate in distributed transactions; they support single-site transactions only
- Does not support multithreaded agents
- Does not support updating LONG columns with bind variables
- Does not support rowids
- COMMIT or ROLLBACK in PL/SQL Cursor Loops Closes Open Cursors
- SQL Syntax

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 the following error:

```
ORA-1002:  fetch out of sequence
```

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

SQL Syntax

This section lists restrictions on the following SQL syntax:

- WHERE CURRENT OF Clause
- CONNECT BY Clause
- ROWID
- Subqueries in INSERT Statement
- EXPLAIN PLAN Statement

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 `WHERE` clause must be used.

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. For example, the following statement is not supported:

```
SQL> INSERT INTO "emp_target"@SYBS
        SELECT a."empno" FROM "emp_source"@SYBS a,
           "emp_source"@SYBS b WHERE b."empno"=9999
```

EXPLAIN PLAN Statement

The EXPLAIN PLAN statement is not supported.

- SQL*Plus COPY Command with Lowercase Table Names

  You need to use double quotes to wrap around lower case table names.

  For example:

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

> **See Also:** Appendix B, "Supported SQL Syntax and Functions"
> for more information about restrictions on SQL syntax.

Known Problems

This section describes known problems and includes suggestions for correcting them when possible. If you have any questions or concerns about the problems, contact Oracle Support Services.

The following known problems are described in this section:

- Encrypted Format Login
- Date Arithmetic

Encrypted Format Login

Oracle database no longer supports the initialization parameter `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.

Date Arithmetic

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

```
date + number
number + date
date - number
date1 - date2
```

Statements with the preceding expressions are sent to the non-oracle system without any translation. If the non-Oracle system does not support these date arithmetic functions, then the statements return an error.

A

Data Type Conversion

Oracle maps the ANSI data types through ODBC interfaces to supported Oracle data types. When the results of a query are returned, Oracle converts the ODBC data types to Oracle data types.

The tables in this appendix show how Oracle maps ANSI data types through ODBC interfaces to supported Oracle data types when it is retrieving data from a non-Oracle system.

This appendix contains the following table:

- Table A–1, " Mapping ANSI Data Types to Oracle Data Types Through an ODBC Interface"

Mapping ANSI Data Types to Oracle Data Types Through an ODBC Interface

The Oracle Database Gateway for ODBC maps the data types used in ODBC-compliant data sources to supported Oracle data types. When the results of a query are returned, the Oracle database converts the ODBC data types to Oracle data types. For example, the ODBC data type `SQL_TIMESTAMP` are converted to Oracle's `DATE` data type.

If a table contains a column whose data type is not supported by Oracle Database Gateway for ODBC, the column information is not returned to the Oracle database.

Table A–1 maps ODBC data types into equivalent ANSI and Oracle data types. In some cases equivalence to ANSI data types is not guaranteed to be exact because the ANSI SQL standard delegates definition of numeric precision and maximum length of character data to individual implementations. This table reflects a probable mapping between ANSI and ODBC data types for a typical implementation of ANSI SQL .

Table A–1   Mapping ANSI Data Types to Oracle Data Types Through an ODBC Interface

ANSI	ODBC	Oracle
NUMERIC(19,0)	SQL_BIGINT	NUMBER(19,0)
N/A	SQL_BINARY	RAW
CHAR	SQL_CHAR	CHAR
DATE	SQL_DATE	DATE
DECIMAL(p,s)	SQL_DECIMAL(p,s)	NUMBER(p,s)
DOUBLE PRECISION	SQL_DOUBLE	FLOAT(53)
FLOAT	SQL_FLOAT	FLOAT(53)

Table A–1 (Cont.) Mapping ANSI Data Types to Oracle Data Types Through an ODBC

ANSI	ODBC	Oracle
INTEGER	SQL_INTEGER	NUMBER(10)
		Note: It is possible under some circumstance for the INTEGER ANSI data type to map to Precision 38, but it usually maps to Precision 10.
N/A	SQL_LONGVARBINARY	LONG RAW
N/A	SQL_LONGVARCHAR	LONG
		Note: If an ANSI SQL implementation defines a large value for the maximum length of VARCHAR data, it is possible that ANSI VARCHAR will map to SQL_LONGVARCHAR and Oracle LONG. The same is true for OLE DB DBTYPE_STRING (long attribute).
REAL	SQL_REAL	FLOAT(24)
SMALLINT	SQL_SMALLINT	NUMBER(5)
TIME	SQL_TYPE_TIME	CHAR(15)
TIMESTAMP	SQL_TYPE_TIMESTAMP	DATE
NUMERIC(3,0)	SQL_TINYINT	NUMBER(3)
VARCHAR	SQL_VARCHAR	VARCHAR2
BOOLEAN	SQL_BIT	NUMBER(3)

B

Supported SQL Syntax and Functions

This appendix contains the following sections:

- Supported SQL Statements
- Oracle Functions

Supported SQL Statements

Oracle Database Gateway for ODBC supports the following statements, but only if the ODBC driver and non-Oracle system can execute them *and* if the statements contain supported Oracle SQL functions:

- `DELETE`
- `INSERT`
- `SELECT`
- `UPDATE`

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 non-Oracle system database.

> **Note:** `TRUNCATE` cannot be used in a pass-through statement.

> **See Also:** *Oracle Database SQL Language Reference* for detailed descriptions of keywords, parameters, and options.

DELETE

The `DELETE` statement is fully supported. However, only Oracle functions supported by non-Oracle system can be used.

INSERT

The `INSERT` statement is fully supported. However, only Oracle functions supported by the non-Oracle system can be used.

SELECT

The SELECT statement is fully supported, with these exceptions:

- `CONNECT BY` *condition*

- `NOWAIT`

- `START WITH` *condition*

- `WHERE CURRENT` OF

UPDATE

The `UPDATE` statement is fully supported. However, only Oracle functions supported by non-Oracle system 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 SET clause.

Oracle Functions

All functions are evaluated by the non-Oracle system after the gateway has converted them to the native SQL. Only a limited set of functions are assumed to be supported by the non-Oracle system. Most Oracle functions have no equivalent function in this limited set. Consequently, although post-processing is performed by the Oracle database server, many Oracle functions are not supported by Oracle Database Gateway for ODBC, possibly impacting performance.

If an Oracle SQL function is not supported by Oracle Database Gateway for ODBC, this function is not supported in `DELETE`, `INSERT`, or `UPDATE` statements. In `SELECT` statements, these functions are evaluated by the Oracle database server and processed after they are returned from the non-Oracle system.

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

```
ORA-02070: database db_link_name does not support function in this context
```

Oracle Database Gateway for ODBC assumes that the following minimum set of SQL functions is supported by the ODBC driver provider that is being used:

- `AVG(`*exp*`)`

- `LIKE(`*exp*`)`

- `COUNT(*)`

- `MAX(`*exp*`)`

- `MIN(`*exp*`)`

- `NOT`

C

Data Dictionary

Data dictionary information is stored in the non-Oracle system as system tables and is accessed through ODBC application programming interfaces (APIs). This appendix documents data dictionary translation support. It explains how to access non-Oracle data dictionaries, describes how to use supported views and tables, and explains data dictionary mapping.

This appendix contains the following topics:

- Accessing the Non-Oracle Data Dictionary
- Views and Tables Supported by Oracle Database Gateway for ODBC

Accessing the Non-Oracle Data Dictionary

Accessing a non-Oracle data dictionary table or view is identical to accessing a data dictionary in an Oracle database. You issue a SELECT statement specifying a database link. The Oracle data dictionary view and column names are used to access the non-Oracle data dictionary. Synonyms of supported views are also acceptable.

For example, the following statement queries the data dictionary table ALL_USERS to retrieve all users in the non-Oracle system:

```
SQL SELECT * FROM all_users@sid1;
```

When you issue a data dictionary access query, the ODBC agent:

1. Maps the requested table, view, or synonym to one or more ODBC APIs (see Section , "Data Dictionary Mapping"). The agent translates all data dictionary column names to their corresponding non-Oracle column names within the query.

2. Sends the sequence of APIs to the non-Oracle system.

3. Possibly converts the retrieved non-Oracle data to give it the appearance of the Oracle data dictionary table.

4. Passes the data dictionary information from the non-Oracle system table to Oracle.

> **Note:** The values returned when querying the Oracle Database Gateway for ODBC data dictionary may not be the same as those returned by the Oracle SQL*Plus DESCRIBE command.

Views and Tables Supported by Oracle Database Gateway for ODBC

Oracle Database Gateway for ODBC supports only the views and tables shown in Table C–1.

If you use an unsupported view, you receive an Oracle error message stating no rows were selected.

If you want to query data dictionary views using `SELECT... FROM DBA_*`, first connect as Oracle user `SYSTEM` or `SYS`. Otherwise, you receive the following error message:

```
ORA-28506: Parse error in data dictionary translation for %s stored in %s
```

Using Oracle Database Gateway for ODBC, queries of the supported data dictionary tables and views beginning with the characters `ALL_` may return rows from the non-Oracle system when you do not have access privileges for those non-Oracle objects. When querying an Oracle database with the Oracle data dictionary, rows are returned only for those objects you are permitted to access.

Data Dictionary Mapping

The tables in this section list Oracle data dictionary view names and the equivalent ODBC APIs used. Table C–1 shows a list of all Oracle data dictionary view names supported by Oracle Database Gateway for ODBC.

Table C–1   Oracle Database Gateway for ODBC Data Dictionary Mapping

View	ODBC API
ALL_CATALOG	SQLTables
ALL_COL_COMMENTS	SQLColumns
ALL_CONS_COLUMNS	SQLPrimaryKeys, SQLForeignKeys
ALL_CONSTRAINTS	SQLPrimaryKeys, SQLForeignKeys
ALL_IND_COLUMNS	SQLStatistics
ALL_INDEXES	SQLStatistics
ALL_OBJECTS	SQLTables, SQLProcedures, SQLStatistics
ALL_TAB_COLUMNS	SQLColumns
ALL_TAB_COMMENTS	SQLTables
ALL_TABLES	SQLStatistics
ALL_USERS	SQLTables
ALL_VIEWS	SQLTables
DICTIONARY	SQLTables
USER_CATALOG	SQLTables
USER_COL_COMMENTS	SQLColumns
USER_CONS_COLUMNS	SQLPrimaryKeys, SQLForeignKeys
USER_CONSTRAINTS	SQLPrimaryKeys, SQLForeignKeys
USER_IND_COLUMNS	SQLStatistics
USER_INDEXES	SQLStatistics
USER_OBJECTS	SQLTables, SQLProcedures, SQLStatistics
USER_TAB_COLUMNS	SQLColumns
USER_TAB_COMMENTS	SQLTables

Table C–1   (Cont.) Oracle Database Gateway for ODBC Data Dictionary Mapping

View	ODBC API
USER_TABLES	SQLStatistics
USER_USERS	SQLTables
USER_VIEWS	SQLTables

Oracle Database Gateway for ODBC Data Dictionary Descriptions

The Oracle Database Gateway for ODBC data dictionary tables and views provide the following information:

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

In the descriptions that follow, the values in the Null? column may differ from the Oracle data dictionary tables and views. Any default value is shown to the right of an item.

ALL_CATALOG

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_TYPE	-	VARCHAR2(11)	"TABLE" or "VIEW" or "SYNONYM"

ALL_COL_COMMENTS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	NOT NULL	VARCHAR2(30)	-
COMMENTS	-	VARCHAR2(4000)	NULL

ALL_CONS_COLUMNS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
CONSTRAINT_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	-	VARCHAR2(4000)	-
POSITION	-	NUMBER	-

ALL_CONSTRAINTS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
CONSTRAINT_NAME	NOT NULL	VARCHAR2(30)	-
CONSTRAINT_TYPE	-	VARCHAR2(1)	"R" or "P"
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
SEARCH_CONDITION	-	LONG	NULL
R_OWNER	-	VARCHAR2(30)	-
R_CONSTRAINT_NAME	-	VARCHAR2(30)	-
DELETE_RULE	-	VARCHAR2(9)	"CASCADE" or "NO ACTION" or "SET NULL"
STATUS	-	VARCHAR2(8)	NULL
DEFERRABLE	-	VARCHAR2(14)	NULL
DEFERRED	-	VARCHAR2(9)	NULL
VALIDATED	-	VARCHAR2(13)	NULL
GENERATED	-	VARCHAR2(14)	NULL
BAD	-	VARCHAR2(3)	NULL
RELY	-	VARCHAR2(4)	NULL
LAST_CHANGE	-	DATE	NULL

ALL_IND_COLUMNS

Name	Null?	Type	Value
INDEX_OWNER	NOT NULL	VARCHAR2(30)	-
INDEX_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	-	VARCHAR2(4000)	-
COLUMN_POSITION	NOT NULL	NUMBER	-
COLUMN_LENGTH	NOT NULL	NUMBER	-
DESCEND	-	VARCHAR2(4)	"DESC" or "ASC"

ALL_INDEXES

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
INDEX_NAME	NOT NULL	VARCHAR2(30)	-
INDEX_TYPE	-	VARCHAR2(27)	NULL
TABLE_OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-

Name	Null?	Type	Value
TABLE_TYPE	-	CHAR(5)	"TABLE"
UNIQUENESS	-	VARCHAR2(9)	"UNIQUE" or "NONUNIQUE"
COMPRESSION	-	VARCHAR2(8)	NULL
PREFIX_LENGTH	-	NUMBER	0
TABLESPACE_NAME	-	VARCHAR2(30)	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_COLUMNS	-	NUMBER	0
FREELISTS	-	NUMBER	0
FREELIST_GROUPS	-	NUMBER	0
PCT_FREE	-	NUMBER	0
LOGGING	-	VARCHAR2(3)	NULL
BLEVEL	-	NUMBER	0
LEAF_BLOCKS	-	NUMBER	0
DISTINCT_KEYS	-	NUMBER	
AVG_LEAF_BLOCKS_PER_KEY	-	NUMBER	0
AVG_DATA_BLOCKS_PER_KEY	-	NUMBER	0
CLUSTERING_FACTOR	-	NUMBER	0
STATUS	-	VARCHAR2(8)	NULL
NUM_ROWS	-	NUMBER	0
SAMPLE_SIZE	-	NUMBER	0
LAST_ANALYZED	-	DATE	NULL
DEGREE	-	VARCHAR2(40)	NULL
INSTANCES	-	VARCHAR2(40)	NULL
PARTITIONED	-	VARCHAR2(3)	NULL
TEMPORARY	-	VARCHAR2(1)	NULL
GENERATED	-	VARCHAR2(1)	NULL
SECONDARY	-	VARCHAR2(1)	NULL
BUFFER_POOL	-	VARCHAR2(7)	NULL
USER_STATS	-	VARCHAR2(3)	NULL

Name	Null?	Type	Value
DURATION	-	VARCHAR2(15)	NULL
PCT_DIRECT_ ACCESS	-	NUMBER	0
ITYP_OWNER	-	VARCHAR2(30)	NULL
ITYP_NAME	-	VARCHAR2(30)	NULL
PARAMETERS	-	VARCHAR2(1000)	NULL
GLOBAL_STATS	-	VARCHAR2(3)	NULL
DOMIDX_STATUS	-	VARCHAR2(12)	NULL
DOMIDX_OPSTATUS	-	VARCHAR2(6)	NULL
FUNCIDX_STATUS	-	VARCHAR2(8)	NULL

ALL_OBJECTS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
OBJECT_NAME	NOT NULL	VARCHAR2(30)	-
SUBOBJECT_NAME	–	VARCHAR2(30)	NULL
OBJECT_ID	NOT NULL	NUMBER	0
DATA_OBJECT_ID	–	NUMBER	0
OBJECT_TYPE	-	VARCHAR2(18)	"TABLE" or "VIEW" or "SYNONYM" or "INDEX" or "PROCEDURE"
CREATED	NOT NULL	DATE	NULL
LAST_DDL_TIME	NOT NULL	DATE	NULL
TIMESTAMP	-	VARCHAR2(19)	NULL
STATUS	-	VARCHAR2(7)	NULL
TEMPORARY	-	VARCHAR2(1)	NULL
GENERATED	-	VARCHAR2(1)	NULL
SECONDARY	-	VARCHAR2(1)	NULL

ALL_TAB_COLUMNS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	NOT NULL	VARCHAR2(30)	-
DATA_TYPE	-	VARCHAR2(106)	-
DATA_TYPE_MOD	-	VARCHAR2(3)	NULL
DATA_TYPE_OWNER	-	VARCHAR2(30)	NULL
DATA_LENGTH	NOT NULL	NUMBER	-

Name	Null?	Type	Value
DATA_PRECISION	-	NUMBER	-
DATA_SCALE	-	NUMBER	-
NULLABLE	-	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	NOT NULL	NUMBER	-
DEFAULT_LENGTH	–	NUMBER	0
DATA_DEFAULT	-	LONG	NULL
NUM_DISTINCT	-	NUMBER	0
LOW_VALUE	-	RAW(32)	NULL
HIGH_VALUE	-	RAW(32)	NULL
DENSITY	-	NUMBER	0
NUM_NULLS	-	NUMBER	0
NUM_BUCKETS	-	NUMBER	0
LAST_ANALYZED	-	DATE	NULL
SAMPLE_SIZE	-	NUMBER	0
CHARACTER_SET_NAME	-	VARCHAR2(44)	NULL
CHAR_COL_DEC_LENGTH	-	NUMBER	0
GLOBAL_STATS	-	VARCHAR2(3)	NULL
USER_STATS	-	VARCHAR2(3)	NULL
AVG_COL_LEN	-	NUMBER	0

ALL_TAB_COMMENTS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_TYPE	-	VARCHAR2(11)	"TABLE" or "VIEW"
COMMENTS	-	VARCHAR2(4000)	NULL

ALL_TABLES

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
TABLESPACE_NAME	-	VARCHAR2(30)	NULL
CLUSTER_NAME	-	VARCHAR2(30)	NULL
IOT_NAME	-	VARCHAR2(30)	NULL
PCT_FREE	-	NUMBER	0
PCT_USED	-	NUMBER	0
INI_TRANS	-	NUMBER	0

Name	Null?	Type	Value
MAX_TRANS	-	NUMBER	0
INITIAL_EXTENT	-	NUMBER	0
NEXT_EXTENT	-	NUMBER	0
MIN_EXTENTS	-	NUMBER	0
MAX_EXTENTS	-	NUMBER	0
PCT_INCREASE	-	NUMBER	0
FREELISTS	-	NUMBER	0
FREELIST_GROUPS	-	NUMBER	0
LOGGING	-	VARCHAR2(3)	NULL
BACKED_UP	-	VARCHAR2(1)	NULL
NUM_ROWS	-	NUMBER	-
BLOCKS	-	NUMBER	-
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(10)	NULL
INSTANCES	-	VARCHAR2(10)	NULL
CACHE	-	VARCHAR2(5)	NULL
TABLE_LOCK	-	VARCHAR2(8)	NULL
SAMPLE_SIZE	-	NUMBER	0
LAST_ANALYZED	-	DATE	NULL
PARTITIONED	-	VARCHAR2(3)	NULL
IOT_TYPE	-	VARCHAR2(12)	NULL
TEMPORARY	-	VARCHAR2(1)	NULL
SECONDARY	-	VARCHAR2(1)	NULL
NESTED	-	VARCHAR2(3)	NULL
BUFFER_POOL	-	VARCHAR2(7)	NULL
ROW_MOVEMENT	-	VARCHAR2(8)	NULL
GLOBAL_STATS	-	VARCHAR2(3)	NULL
USER_STATS	-	VARCHAR2(3)	NULL
DURATION	-	VARHCAR2(15)	NULL
SKIP_CORRUPT	-	VARCHAR2(8)	NULL
MONITORING	-	VARCHAR2(3)	NULL

ALL_USERS

Name	Null?	Type	Value
USERNAME	NOT NULL	VARCHAR2(30)	-
USER_ID	NOT NULL	NUMBER	0
CREATED	NOT NULL	DATE	NULL

ALL_VIEWS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
VIEW_NAME	NOT NULL	VARCHAR2(30)	-
TEXT_LENGTH	-	NUMBER	0
TEXT	NOT NULL	LONG	NULL
TYPE_TEXT_LENGTH	-	NUMBER	0
TYPE_TEXT	-	VARCHAR2(4000)	NULL
OID_TEXT_LENGTH	-	NUMBER	0
OID_TEXT	-	VARCHAR2(4000)	NULL
VIEW_TYPE_OWNER	-	VARCHAR2(30)	NULL
VIEW_TYPE	-	VARCHAR2(30)	NULL

DICTIONARY

Name	Null?	Type	Value
TABLE_NAME	-	VARCHAR2(30)	-
COMMENTS	-	VARCHAR2(4000)	NULL

DICT_COLUMNS

Name	Null?	Type	Value
TABLE_NAME	-	VARCHAR2(30)	-
COLUMN_NAME	-	VARCHAR2(30)	-
COMMENTS	-	VARCHAR2(4000)	NULL

USER_CATALOG

Name	Null?	Type	Value
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_TYPE	-	VARCHAR2(11)	"TABLE" or, "VIEW" or "SYNONYM"

USER_COL_COMMENTS

Name	Null?	Type	Value
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	NOT NULL	VARCHAR2(30)	-
COMMENTS	-	VARCHAR2(4000)	NULL

USER_CONS_COLUMNS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
CONSTRAINT_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	-	VARCHAR2(4000)	-
POSITION	-	NUMBER	-

USER_CONSTRAINTS

Name	Null?	Type	Value
OWNER	NOT NULL	VARCHAR2(30)	-
CONSTRAINT_NAME	NOT NULL	VARCHAR2(30)	-
CONSTRAINT_TYPE	-	VARCHAR2(1)	R or P
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
SEARCH_CONDITION	-	LONG	NULL
R_OWNER	-	VARCHAR2(30)	-
R_CONSTRAINT_NAME	-	VARCHAR2(30)	-
DELETE_RULE	-	VARCHAR2(9)	"CASCADE" or "NO ACTION" or "SET NULL"
STATUS	-	VARCHAR2(8)	NULL
DEFERRABLE	-	VARCHAR2(14)	NULL
DEFERRED	-	VARCHAR2(9)	NULL
VALIDATED	-	VARCHAR2(13)	NULL
GENERATED	-	VARCHAR2(14)	NULL
BAD	-	VARCHAR2(3)	NULL
RELY	-	VARCHAR2(4)	NULL
LAST_CHANGE	-	DATE	NULL

USER_IND_COLUMNS

Name	Null?	Type	Value
INDEX_NAME	-	VARCHAR2(30)	-

Name	Null?	Type	Value
TABLE_NAME	-	VARCHAR2(30)	-
COLUMN_NAME	-	VARCHAR2(4000)	–
COLUMN_POSITION	-	NUMBER	–
COLUMN_LENGTH	-	NUMBER	–
DESCEND	-	VARCHAR2(4)	"DESC" or "ASC"

USER_INDEXES

Name	Null?	Type	Value
INDEX_NAME	NOT NULL	VARCHAR2(30)	–
INDEX_TYPE	-	VARCHAR2(27)	NULL
TABLE_OWNER	NOT NULL	VARCHAR2(30)	-
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_TYPE	-	VARCHAR2(11)	"TABLE"
UNIQUENESS	-	VARCHAR2(9)	"UNIQUE" or "NONUNIQUE"
COMPRESSION	-	VARCHAR2(8)	NULL
PREFIX_LENGTH	-	NUMBER	0
TABLESPACE_NAME	-	VARCHAR2(30)	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_COLUMNS	-	NUMBER	0
FREELISTS	-	NUMBER	0
FREELIST_GROUPS	-	NUMBER	0
PCT_FREE	-	NUMBER	0
LOGGING	-	VARCHAR2(3)	NULL
BLEVEL	-	NUMBER	0
LEAF_BLOCKS	-	NUMBER	0
DISTINCT_KEYS	-	NUMBER	-
AVG_LEAF_BLOCKS_PER_KEY	-	NUMBER	0
AVG_DATA_BLOCKS_PER_KEY	-	NUMBER	0
CLUSTERING_FACTOR	-	NUMBER	0
STATUS	-	VARCHAR2(8)	NULL

Name	Null?	Type	Value
NUM_ROWS	-	NUMBER	0
SAMPLE_SIZE	-	NUMBER	0
LAST_ANALYZED	-	DATE	NULL
DEGREE	-	VARCHAR2(40)	NULL
INSTANCES	-	VARCHAR2(40)	NULL
PARTITIONED	-	VARCHAR2(3)	NULL
TEMPORARY	-	VARCHAR2(1)	NULL
GENERATED	-	VARCHAR2(1)	NULL
SECONDARY	-	VARCHAR2(1)	NULL
BUFFER_POOL	-	VARCHAR2(7)	NULL
USER_STATS	-	VARCHAR2(3)	NULL
DURATION	-	VARHCAR2(15)	NULL
PCT_DIRECT_ACCESS	-	NUMBER	0
ITYP_OWNER	-	VARCHAR2(30)	NULL
ITYP_NAME	-	VARCHAR2(30)	NULL
PARAMETERS	-	VARCHAR2(1000)	NULL
GLOBAL_STATS	-	VARCHAR2(3)	NULL
DOMIDX_STATUS	-	VARCHAR2(12)	NULL
DOMIDX_OPSTATUS	-	VARCHAR2(6)	NULL
FUNCIDX_STATUS	-	VARCHAR2(8)	NULL

USER_OBJECTS

Name	Null?	Type	Value
OBJECT_NAME	-	VARCHAR2(128)	-
SUBOBJECT_NAME	-	VARCHAR2(30)	NULL
OBJECT_ID	-	NUMBER	0
DATA_OBJECT_ID	-	NUMBER	0
OBJECT_TYPE	-	VARCHAR2(18)	"TABLE" or "VIEW" or "SYNONYM" or "INDEX" or "PROCEDURE"
CREATED	-	DATE	NULL
LAST_DDL_TIME	-	DATE	NULL
TIMESTAMP	-	VARCHAR2(19)	NULL
STATUS	-	VARCHAR2(7)	NULL
TEMPORARY	-	VARCHAR2(1)	NULL
GENERATED	-	VARCHAR2(1)	NULL
SECONDARY	-	VARCHAR2(1)	NULL

USER_TAB_COLUMNS

Name	Null?	Type	Value
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
COLUMN_NAME	NOT NULL	VARCHAR2(30)	-
DATA_TYPE	-	VARCHAR2(106)	-
DATA_TYPE_MOD	-	VARCHAR2(3)	NULL
DATA_TYPE_OWNER	-	VARCHAR2(30)	NULL
DATA_LENGTH	NOT NULL	NUMBER	-
DATA_PRECISION	-	NUMBER	-
DATA_SCALE	-	NUMBER	-
NULLABLE	-	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	NOT NULL	NUMBER	-
DEFAULT_LENGTH	-	NUMBER	NULL
DATA_DEFAULT	-	LONG	NULL
NUM_DISTINCT	-	NUMBER	NULL
LOW_VALUE	-	RAW(32)	NULL
HIGH_VALUE	-	RAW(32)	NULL
DENSITY	-	NUMBER	0
NUM_NULLS	-	NUMBER	0
NUM_BUCKETS	-	NUMBER	0
LAST_ANALYZED	-	DATE	NULL
SAMPLE_SIZE	-	NUMBER	0
CHARACTER_SET_NAME	-	VARCHAR2(44)	NULL
CHAR_COL_DECL_LENGTH	-	NUMBER	0
GLOBAL_STATS	-	VARCHAR2(3)	NULL
USER_STATS	-	VARCHAR2(3)	NULL
AVG_COL_LEN	-	NUMBER	0

USER_TAB_COMMENTS

Name	Null?	Type	Value
TABLE_NAME	NOT NULL	VARCHAR2(30)	-
TABLE_TYPE	-	VARCHAR2(11)	"TABLE" or "VIEW"
COMMENTS	-	VARCHAR2(4000)	NULL

USER_TABLES

Name	Null?	Type	Value
TABLE_NAME	NOT NULL	VARCHAR2(30)	-

Name	Null?	Type	Value
TABLESPACE_NAME	-	VARCHAR2(30)	NULL
CLUSTER_NAME	-	VARCHAR2(30)	NULL
IOT_NAME	-	VARCHAR2(30)	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
MAX_EXTENTS	-	NUMBER	0
PCT_INCREASE	-	NUMBER	0
FREELISTS	-	NUMBER	0
FREELIST_GROUPS	-	NUMBER	0
LOGGING	-	VARCHAR2(3)	NULL
BACKED_UP	-	VARCHAR2(1)	NULL
NUM_ROWS	-	NUMBER	-
BLOCKS	-	NUMBER	-
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(10)	NULL
INSTANCES	-	VARCHAR2(10)	NULL
CACHE	-	VARCHAR2(5)	NULL
TABLE_LOCK	-	VARCHAR2(8)	NULL
SAMPLE_SIZE	-	NUMBER	0
LAST_ANALYZED	-	DATE	NULL
PARTITIONED	-	VARCHAR2(3)	NULL
IOT_TYPE	-	VARCHAR2(12)	NULL
TEMPORARY	-	VARHCAR2(1)	NULL
SECONDARY	-	VARCHAR2(1)	NULL
NESTED	-	VARCHAR2(3)	NULL
BUFFER_POOL	-	VARCHAR2(7)	NULL
ROW_MOVEMENT	-	VARCHAR2(8)	NULL
GLOBAL_STATS	-	VARCHAR2(3)	NULL

Name	Null?	Type	Value
USER_STATS	-	VARCHAR2(3)	NULL
DURATION	-	VARCHAR2(15)	NULL
SKIP_CORRUPT	-	VARCHAR2(8)	NULL
MONITORING	-	VARCHAR2(3)	NULL

USER_USERS

Name	Null?	Type	Value
USERNAME	NOT NULL	VARCHAR2(30)	-
USER_ID	NOT NULL	NUMBER	0
ACCOUNT_STATUS	NOT NULL	VARCHAR2(32)	OPEN
LOCK_DATE	-	DATE	NULL
EXPIRY_DATE	-	DATE	NULL
DEFAULT_TABLESPACE	NOT NULL	VARCHAR2(30)	NULL
TEMPORARY_TABLESPACE	NOT NULL	VARCHAR2(30)	NULL
CREATED	NOT NULL	DATE	NULL
INITIAL_RSRC_CONSUMER_ GROUP	-	VARCHAR2(30)	NULL
EXTERNAL_NAME	-	VARCHAR2(4000)	NULL

USER_VIEWS

Name	Null?	Type	Value
VIEW_NAME	NOT NULL	VARCHAR2(30)	-
TEXT_LENGTH	-	NUMBER	0
TEXT	-	LONG	NULL
TYPE_TEXT_LENGTH	-	NUMBER	0
TYPE_TEXT	-	VARCHAR2(4000)	NULL
OID_TEXT_LENGTH	-	NUMBER	0
OID_TEXT	-	VARCHAR2(4000)	NULL
VIEW_TYPE_OWNER	-	VARCHAR2(30)	NULL
VIEW_TYPE	-	VARCHAR2(30)	NULL

D

Initialization Parameters

The Oracle database initialization parameters in the `init.ora` file are distinct from 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 `DBMS_HS` package. The gateway initialization parameter file must be available when the gateway is started.

This appendix contains a list of the gateway initialization parameters that can be set for each gateway and their description. It also describes the initialization parameter file syntax. It includes the following sections:

- Initialization Parameter File Syntax
- Oracle Database Gateway for ODBC Initialization Parameters
- Initialization Parameter Descriptions

Initialization Parameter File Syntax

The syntax for the initialization parameter file is as follows:

1. The file is a sequence of commands.

2. Each command should start on a separate line.

3. End of line is considered a command terminator (unless escaped with a backslash).

4. If there is a syntax error in an initialization parameter file, none of the settings take effect.

5. 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. A string of characters that does not contain any backslashes, white space or double quotation marks (")

   b. 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 HS_FDS_TRACE_LEVEL 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 username 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).

Oracle Database Gateway for ODBC Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for ODBC. They are as follows:

- HS_DB_DOMAIN

- HS_DB_INTERNAL_NAME

- HS_DB_NAME

- HS_DESCRIBE_CACHE_HWM

- HS_LANGUAGE

- HS_LONG_PIECE_TRANSFER_SIZE

- HS_OPEN_CURSORS

- HS_RPC_FETCH_REBLOCKING

- HS_RPC_FETCH_SIZE

- HS_FDS_SHAREABLE_NAME

- HS_TIME_ZONE

- IFILE

- HS_FDS_CONNECT_INFO

- HS_FDS_DEFAULT_OWNER

- HS_FDS_TRACE_LEVEL

- HS_TRANSACTION_MODEL

- HS_FDS_FETCH_ROWS

Initialization Parameter Description

The following sections describe all the initialization file parameters that can be set for gateways.

HS_DB_DOMAIN

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

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

> **Note:** The HS_DB_NAME and HS_DB_DOMAIN initialization parameters must combine to form a unique address in a cooperative server environment.

HS_DB_INTERNAL_NAME

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.

HS_DB_NAME

Property	Description
Default value	HO
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 HS_DB_NAME and HS_DB_DOMAIN initialization parameters define the global name of the non-Oracle system.

HS_DESCRIBE_CACHE_HWM

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.

HS_LANGUAGE

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]
```

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

Character Sets

Ideally, the character sets of the Oracle database and the non-Oracle data source are the same. 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.

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

Language

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.

> **Note:** Use the `HS_NLS_DATE_LANGUAGE` initialization parameter to set the day and month names, and the AD, BC, PM, and AM symbols for dates and time independently from the language.

Territory

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.

HS_LONG_PIECE_TRANSFER_SIZE

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.

HS_OPEN_CURSORS

Property	Description
Default value	50
Range of values	1 to the value of `OPEN_CURSORS` initialization parameter of Oracle database

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 OPEN_CURSORS initialization parameter in the Oracle database is recommended.

HS_RPC_FETCH_REBLOCKING

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.

HS_RPC_FETCH_SIZE

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 response time or *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.

HS_TIME_ZONE

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
```

HS_TRANSACTION_MODEL

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

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

The following values are possible:

- 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.

IFILE

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 does not apply.

> **See Also:** *Oracle Database Reference*

HS_FDS_CONNECT_INFO

Property	Description
Default Value	None
Range of Values	Not applicable

HS_FDS_CONNECT_INFO which describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for HS_FDS_CONNECT_INFO for the gateway is as follows:

```
HS_FDS_CONNECT_INFO=dsn_value
```

where, *dsn_value* on Windows, is the name of the system DSN defined in the Windows ODBC Data Source Administrator and on UNIX, it is data source name configured in the odbc.ini file.

The entry for *dsn_value* is case sensitive.

HS_FDS_DEFAULT_OWNER

Property	Description
Default Value	None
Range of Values	Not applicable

The name of the table owner that is used for the non-Oracle database tables if an owner is not specified in the SQL statements.

> **Note:** If this parameter is not specified and the owner is not explicitly specified in the SQL statement, then the user name of the Oracle user or the user name specified when creating the database link is used.

HS_FDS_TRACE_LEVEL

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.

HS_FDS_SHAREABLE_NAME

Property	Description
Default Value	None
Range of Values	Not applicable

Specifies the full path name to the ODBC driver manager.

This is a required parameter, whose format is:

HS_FDS_SHAREABLE_NAME=*odbc_installation_path*/lib/libodbc.sl

Where:

odbc_installation_path is the path where the ODBC driver is installed.

Thsi parameter applies only to UNIX based platforms.

HS_FDS_FETCH_ROWS

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 HS_RPC_FETCH_SIZE and HS_RPC_FETCH_ REBLOCKING parameters.

Index

A

ALTER statement, B-1

C

character sets
    Heterogeneous Services, D-4
CONNECT BY clause, 2-3
CREATE statement, B-1

D

Data definition language, B-1
data dictionary
    contents with Oracle Database Gateway for
        ODBC, C-1
    mapping for Oracle Database Gateway for
        ODBC, C-2
    Oracle server name/SQL Server name, C-2
    translation support for Oracle Database Gateway
        for ODBC, C-1
data dictionary views
    Oracle Database Gateway for ODBC, C-1
Data type
    VARBINARY, 2-3
DELETE statement, B-1
describe cache high water mark
    definition, D-4
drivers
    ODBC, 1-5
DROP statement, B-1

E

Encrypted format login, 2-3
Error messages
    error tracing, D-8

F

fetch array size, with HS_FDS_FETCH_ROWS, D-9

G

Gateway
    pass-through feature, 2-1

supported functions, B-1
supported SQL syntax, B-1
globalization support
    Heterogeneous Services, D-4
GRANT statement, B-1

H

Heterogeneous Services
    defining maximum number of open cursors, D-5
    optimizing data transfer, D-6
    Oracle Database Gateway for ODBC
        architecture, 1-2
        definition, 1-2
        non-Oracle data dictionary access, C-1
        ODBC connectivity requirements, 1-5
        supported functions, B-2
        supported SQL syntax, B-1
        supported tables, C-1
    setting global name, D-4
    specifying cache high water mark, D-4
    tuning internal data buffering, D-6
    tuning LONG data transfer, D-5
HS_DB_NAME initialization parameter, D-4
HS_DESCRIBE_CACHE_HWM initialization
    parameter, D-4
HS_FDS_CONNECT_INFO, D-7
HS_FDS_DEFAULT_OWNER initialization
    parameter, D-8
HS_FDS_FETCH_ROWS parameter, D-9
HS_FDS_SHAREABLE_NAME initialization
    parameter, D-8
HS_FDS_TRACE_LEVEL initialization
    parameter, D-8
    enabling agent tracing, D-2
HS_LANGUAGE initialization parameter, D-4
HS_LONG_PIECE_TRANSFER_SIZE initialization
    parameter, D-5
HS_OPEN_CURSORS initialization parameter, D-5
HS_RPC_FETCH_REBLOCKING initialization
    parameter, D-6
HS_RPC_FETCH_SIZE initialization parameter, D-6
HS_TIME_ZONE initialization parameter, D-6

I

IFILE initialization parameter, D-7
Initialization parameter file
 customizing, D-1
INSERT statement, 2-3, B-1

K

Known restrictions, 2-2

O

ODBC agents
 connectivity requirements, 1-5
 functions, 1-5
ODBC connectivity
 data dictionary mapping, C-2
 ODBC driver, 1-5
 requirements, 1-5
 specifying path to library, D-8
OLE DB connectivity
 data dictionary mapping, C-2
Oracle Database Gateway for ODBC
 architecture, 1-2
  Oracle and non-Oracle on same machine, 1-3
  Oracle and non-Oracle on separate
   machines, 1-3
 data dictionary
  translation support, C-1
 definition, 1-2
 DELETE statement, B-2
 INSERT statement, B-2
 non-Oracle data dictionary access, C-1
 ODBC connectivity requirements, 1-5
 supported functions, B-2
 supported SQL syntax, B-1
 UPDATE statement, B-2

P

parameters
 gateway initialization file
  HS_FDS_FETCH_ROWS, D-9

R

ROWID, 2-3

S

SELECT statement, B-2
 accessing non-Oracle system, C-1

T

TRUNCATE statement, B-1

U

unsupported functions
 Oracle Database Gateway for ODBC, B-2

UPDATE statement, B-2

V

VARBINARY data type, 2-3

W

WHERE CURRENT OF clause, 2-3

Beta Draft