

# **Oracle® Database Gateway for ODBC**

User's Guide

12c Release 2 (12.2)

**E49918-06**

January 2017

Oracle Database Gateway for ODBC User's Guide, 12c Release 2 (12.2)

E49918-06

Copyright © 2007, 2017, Oracle and/or its affiliates. All rights reserved.

Primary Author: Maitreyee Chaliha

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

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

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

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

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

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

---

---

# Contents

<b>Preface</b> .....	xi
Audience .....	xi
Documentation Accessibility .....	xi
Related Documents.....	xii
Conventions.....	xii
<b>1 Introduction to Oracle Database Gateway for ODBC</b>	
1.1 Overview of Oracle Database Gateways.....	1-2
1.2 About Heterogeneous Services Technology .....	1-2
1.3 About Oracle Database Gateway for ODBC.....	1-2
1.4 Oracle Database Gateway for ODBC Architecture.....	1-3
1.4.1 Oracle and Non-Oracle Systems on Separate Machines.....	1-3
1.4.2 Oracle and Non-Oracle Systems on the Same Machine .....	1-4
1.5 ODBC Connectivity Requirements .....	1-6
<b>2 Oracle Database Gateway for ODBC Features and Restrictions</b>	
2.1 Using the Pass-Through Feature .....	2-1
2.2 Using AUTO COMMIT.....	2-2
2.3 Known Restrictions .....	2-2
2.3.1 COMMIT or ROLLBACK in PL/SQL Cursor Loops Closes Open Cursors.....	2-3
2.3.2 SQL Syntax Restrictions .....	2-3
2.3.3 CALLBACK Links Support.....	2-4
2.4 Known Problems.....	2-4
2.4.1 Encrypted Format Login .....	2-4
2.4.2 Date Arithmetic .....	2-4
<b>A Data Type Conversion</b>	
<b>B Supported SQL Syntax and Functions</b>	
B.1 Supported SQL Statements .....	B-1
B.1.1 DELETE .....	B-2
B.1.2 INSERT .....	B-2

B.1.3	SELECT.....	B-2
B.1.4	UPDATE.....	B-2
B.2	Oracle Functions.....	B-2

## C Data Dictionary

C.1	Accessing the Non-Oracle Data Dictionary.....	C-1
C.2	Views and Tables Supported by Oracle Database Gateway for ODBC.....	C-2
C.2.1	Data Dictionary Mapping.....	C-3
C.2.2	ALL_CATALOG.....	C-4
C.2.3	ALL_COL_COMMENTS.....	C-4
C.2.4	ALL_CONS_COLUMNS.....	C-5
C.2.5	ALL_CONSTRAINTS.....	C-5
C.2.6	ALL_IND_COLUMNS.....	C-6
C.2.7	ALL_INDEXES.....	C-6
C.2.8	ALL_OBJECTS.....	C-8
C.2.9	ALL_TAB_COLUMNS.....	C-9
C.2.10	ALL_TAB_COMMENTS.....	C-10
C.2.11	ALL_TABLES.....	C-11
C.2.12	ALL_USERS.....	C-12
C.2.13	ALL_VIEWS.....	C-13
C.2.14	DICTIONARY.....	C-13
C.2.15	DICT_COLUMNS.....	C-14
C.2.16	USER_CATALOG.....	C-14
C.2.17	USER_COL_COMMENTS.....	C-14
C.2.18	USER_CONS_COLUMNS.....	C-15
C.2.19	USER_CONSTRAINTS.....	C-15
C.2.20	USER_IND_COLUMNS.....	C-16
C.2.21	USER_INDEXES.....	C-16
C.2.22	USER_OBJECTS.....	C-18
C.2.23	USER_TAB_COLUMNS.....	C-19
C.2.24	USER_TAB_COMMENTS.....	C-20
C.2.25	USER_TABLES.....	C-20
C.2.26	USER_USERS.....	C-22
C.2.27	USER_VIEWS.....	C-23

## D Initialization Parameters

D.1	Initialization Parameter File Syntax.....	D-1
D.2	Oracle Database Gateway for ODBC Initialization Parameters.....	D-2
D.2.1	HS_DB_DOMAIN.....	D-4
D.2.2	HS_DB_INTERNAL_NAME.....	D-4
D.2.3	HS_DB_NAME.....	D-5
D.2.4	HS_DESCRIBE_CACHE_HWM.....	D-5
D.2.5	HS_LANGUAGE.....	D-5

D.2.6	HS_LONG_PIECE_TRANSFER_SIZE .....	D-6
D.2.7	HS_OPEN_CURSORS .....	D-7
D.2.8	HS_RPC_FETCH_REBLOCKING .....	D-7
D.2.9	HS_RPC_FETCH_SIZE .....	D-8
D.2.10	HS_TIME_ZONE .....	D-8
D.2.11	HS_TRANSACTION_MODEL.....	D-8
D.2.12	IFILE .....	D-9
D.2.13	HS_FDS_TIMESTAMP_MAPPING.....	D-9
D.2.14	HS_FDS_DATE_MAPPING .....	D-9
D.2.15	HS_FDS_CONNECT_INFO.....	D-10
D.2.16	HS_FDS_DEFAULT_OWNER .....	D-10
D.2.17	HS_FDS_TRACE_LEVEL.....	D-10
D.2.18	HS_FDS_SHAREABLE_NAME .....	D-11
D.2.19	HS_FDS_FETCH_ROWS.....	D-11
D.2.20	HS_FDS_REMOTE_DB_CHARSET.....	D-11
D.2.21	HS_FDS_SQLLEN_INTERPRETATION .....	D-12
D.2.22	HS_FDS_REPORT_REAL_AS_DOUBLE.....	D-12

## Index



## List of Figures

1-1	Oracle and Non-Oracle Systems on Separate Machines.....	1-4
1-2	Oracle and Non-Oracle Systems on the Same Machine.....	1-5





## List of Tables

A-1	Data Type Mapping and Restrictions .....	A-1
C-1	Oracle Database Gateway for ODBC Data Dictionary Mapping.....	C-3
C-2	ALL_CATALOG.....	C-4
C-3	ALL_COL_COMMENTS.....	C-4
C-4	ALL_CONS_COLUMNS.....	C-5
C-5	ALL_CONSTRAINTS.....	C-5
C-6	ALL_IND_COLUMNS.....	C-6
C-7	ALL_INDEXES.....	C-7
C-8	ALL_OBJECTS.....	C-9
C-9	ALL_TAB_COLUMNS.....	C-9
C-10	ALL_TAB_COMMENTS.....	C-10
C-11	ALL_TABLES.....	C-11
C-12	ALL_USERS.....	C-12
C-13	ALL_VIEWS.....	C-13
C-14	DICTIONARY.....	C-13
C-15	DICT_COLUMNS.....	C-14
C-16	USER_CATALOG.....	C-14
C-17	USER_COL_COMMENTS.....	C-14
C-18	USER_CONS_COLUMNS.....	C-15
C-19	USER_CONSTRAINTS.....	C-15
C-20	USER_IND_COLUMNS.....	C-16
C-21	USER_INDEXES.....	C-16
C-22	USER_OBJECTS.....	C-18
C-23	USER_TAB_COLUMNS.....	C-19
C-24	USER_TAB_COMMENTS.....	C-20
C-25	USER_TABLES.....	C-21
C-26	USER_USERS.....	C-22
C-27	USER_VIEWS.....	C-23



---

---

# 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.

[Audience](#) (page xi)

[Documentation Accessibility](#) (page xi)

[Related Documents](#) (page xii)

[Conventions](#) (page xii)

## 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

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

---

# Introduction to Oracle Database Gateway for ODBC

Oracle Database Gateways provide the ability to transparently access data residing in a non-Oracle system from an Oracle environment. The following sections briefly cover 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, refer to the *Oracle Database Heterogeneous Connectivity User's Guide* first.

Topics:

## [Overview of Oracle Database Gateways](#) (page 1-2)

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.

## [About Heterogeneous Services Technology](#) (page 1-2)

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 ODBC](#) (page 1-2)

Oracle Database Gateway for ODBC is intended for low-end data integration solutions requiring the dynamic query capability to connect from an Oracle database to non-Oracle systems.

## [Oracle Database Gateway for ODBC Architecture](#) (page 1-3)

The gateway works with an ODBC driver to access the non-Oracle data store using Oracle Database Gateway for ODBC. 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.

## [ODBC Connectivity Requirements](#) (page 1-6)

To use Oracle Database Gateway for ODBC, you must have an ODBC driver installed on the same machine as the gateway.

## 1.1 Overview of Oracle Database Gateways

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 data access is a problem that affects a lot of companies. Many companies run several different database systems. Each of these systems stores data and has a set of applications that run against it. Consolidating 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 migration to one consolidated database system is 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.

## 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.

---

---

**See Also:**

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

---

---

## 1.3 About Oracle Database Gateway for ODBC

Oracle Database Gateway for ODBC is intended for low-end data integration solutions requiring the dynamic 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.

## 1.4 Oracle Database Gateway for ODBC Architecture

The gateway works with an ODBC driver to access the non-Oracle data store using Oracle Database Gateway for ODBC. 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 considerations 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](#) (page 1-3)

The figure is 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.

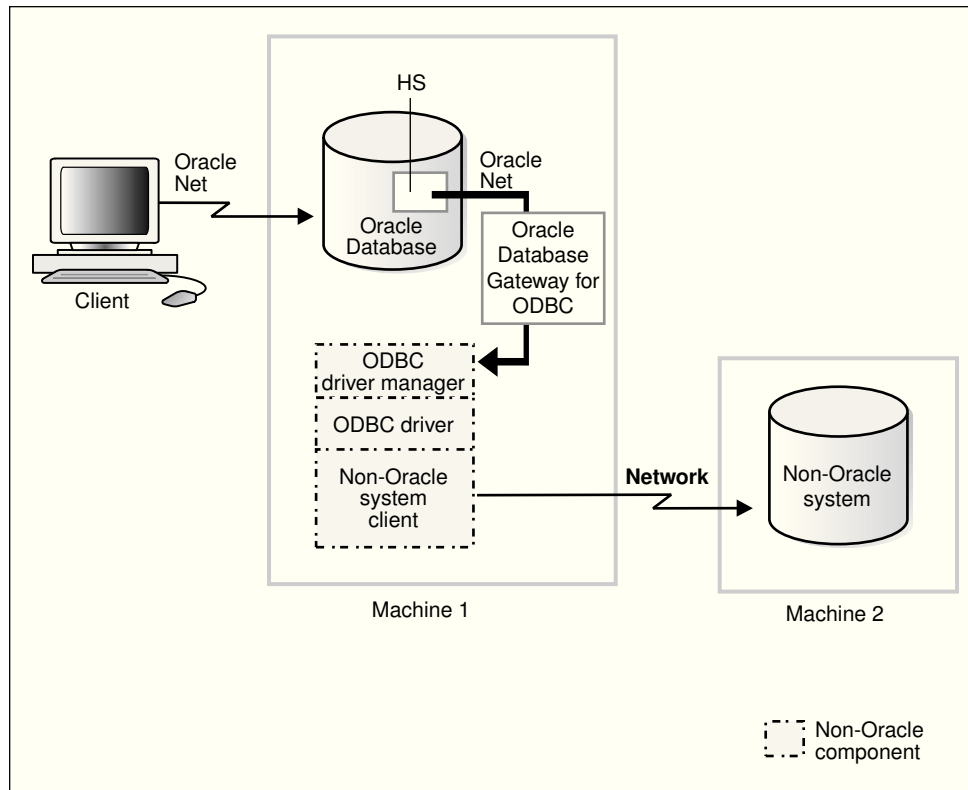
### [Oracle and Non-Oracle Systems on the Same Machine](#) (page 1-4)

The figure is 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.

### 1.4.1 Oracle and Non-Oracle Systems on Separate Machines

The figure is 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](#) (page 1-4) shows a non-Oracle system on a computer that is separate from the Oracle system.

**Figure 1-1 Oracle and Non-Oracle Systems on Separate Machines**

In this configuration:

- A client connects to the Oracle database through Oracle Net.
- The Heterogeneous Services component of the Oracle database connects through Oracle Net to the gateway.
- The gateway communicates with the following non-Oracle components:
  - An ODBC driver manager
  - An ODBC driver
- 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.

---



---

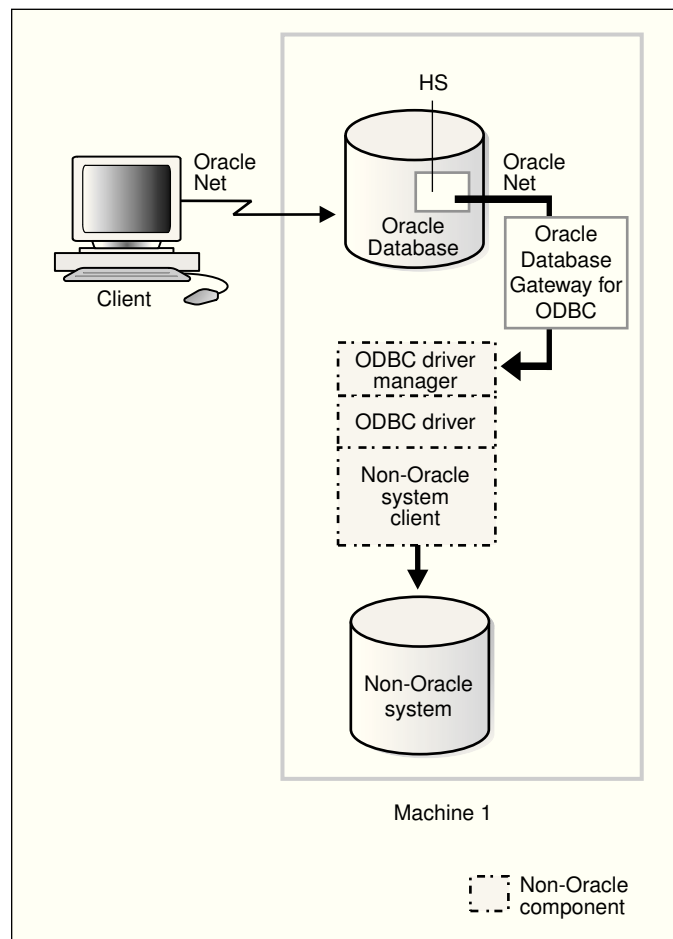
## 1.4.2 Oracle and Non-Oracle Systems on the Same Machine

The figure is 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 (page 1-5) shows a client accessing non-Oracle databases that reside on the same computer as the Oracle databases using Heterogeneous Services.

**Figure 1-2 Oracle and Non-Oracle Systems on the Same Machine**



In this configuration:

- A client connects to the Oracle database through Oracle Net.
- The Heterogeneous Services component of the Oracle database connects through Oracle Net to the gateway.
- 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.

- 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.

---

## 1.5 ODBC Connectivity Requirements

To use Oracle Database Gateway for ODBC, you must have an ODBC driver installed on the same machine as the gateway.

The ODBC driver manager and driver must meet the following requirements:

- The following ODBC catalog functions must work inside a transaction:
  - 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

---

# 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.

Topics:

[Using the Pass-Through Feature](#) (page 2-1)

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

[Using AUTO COMMIT](#) (page 2-2)

Some non-Oracle databases operate without logging. If read-only capability is desired under such environment, you need to set the `HS_TRANSACTION_MODEL` gateway parameter.

[Known Restrictions](#) (page 2-2)

If you encounter incompatibility problems not listed in this section or in "[Known Problems](#) (page 2-4)", contact Oracle Support Services.

[Known Problems](#) (page 2-4)

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.

## 2.1 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.

---

---

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 Database Heterogeneous Connectivity User's Guide* for more information about the DBMS\_HS\_PASSTHROUGH package.

---

---

## 2.2 Using AUTO COMMIT

Some non-Oracle databases operate without logging. If read-only capability is desired under such environment, you need to set the HS\_TRANSACTION\_MODEL gateway parameter.

The HS\_TRANSACTION\_MODEL parameter can be set as follows:

```
HS_TRANSACTION_MODEL=READ_ONLY_AUTOCOMMIT
```

However, if you still need to have update capability, then set HS\_TRANSACTION\_MODEL=SINGLE\_SITE\_AUTOCOMMIT in the gateway initialization parameter file. Any update is committed immediately. Commit-confirm is not allowed for the targets operating without logging.

## 2.3 Known Restrictions

If you encounter incompatibility problems not listed in this section or in "[Known Problems](#) (page 2-4)", contact Oracle Support Services.

The following are the known restrictions:

- Pass-through queries cannot read BLOB and CLOB data
- Updates or deletes that include unsupported functions within a WHERE clause are not allowed
- Oracle Database Gateway for ODBC does not support stored procedures
- Cannot participate in distributed transactions; only single-site transactions supported
- 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](#)  
(page 2-3)

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

[SQL Syntax Restrictions](#) (page 2-3)

Oracle Database Gateway for ODBC has the following restrictions on SQL syntax.

[CALLBACK Links Support](#) (page 2-4)

Oracle Database Gateway for ODBC does not support CALLBACK links.

## 2.3.1 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.

The following error can occur:

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

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

## 2.3.2 SQL Syntax Restrictions

Oracle Database Gateway for ODBC has the following restrictions on SQL syntax.

[WHERE CURRENT OF Clause](#) (page 2-3)

[CONNECT BY Clause](#) (page 2-3)

[ROWID](#) (page 2-3)

[EXPLAIN PLAN Statement](#) (page 2-3)

### 2.3.2.1 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 WHERE clause must be used.

### 2.3.2.2 CONNECT BY Clause

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

### 2.3.2.3 ROWID

The Oracle ROWID implementation is not supported.

### 2.3.2.4 EXPLAIN PLAN Statement

The EXPLAIN PLAN statement is not supported.

- SQL\*Plus COPY Command with Lowercase Table Names

Wrap lower case table name in double quotes.

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:**

[Supported SQL Syntax and Functions](#) (page B-1) for more information about restrictions on SQL syntax.

---

---

### 2.3.3 CALLBACK Links Support

Oracle Database Gateway for ODBC does not support CALLBACK links.

Trying a CALLBACK link with the gateway will return the following error message:

```
ORA-02025: All tables in the SQL statement must be at the remote database
```

## 2.4 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.

[Encrypted Format Login](#) (page 2-4)

[Date Arithmetic](#) (page 2-4)

### 2.4.1 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.

### 2.4.2 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.



## Data Type Conversion

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

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_TYPE_TIMESTAMP` is 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](#) (page A-1) maps ODBC data types into Oracle data types.

**Table A-1 Data Type Mapping and Restrictions**

ODBC	Oracle	Criteria	If Oracle uses large varchar (32k)
<code>SQL_BIGINT</code>	<code>NUMBER(19,0)</code>	-	
<code>SQL_BINARY</code>	<code>RAW</code>	-	
<code>SQL_CHAR</code>	<code>CHAR</code>	-	
<code>SQL_DECIMAL(p,s)</code>	<code>NUMBER(p,s)</code>	-	
<code>SQL_DOUBLE</code>	<code>FLOAT(53)</code>	-	
<code>SQL_FLOAT</code>	<code>FLOAT(53)</code>	-	
<code>SQL_INTEGER</code>	<code>NUMBER(10)</code> Note: It is possible under some circumstance for the <code>INTEGER ANSI</code> data type to map to <code>Precision 38</code> , but it usually maps to <code>Precision 10</code> .	-	
<code>SQL_INTERVAL_YEAR</code>	<code>INTERVAL_YEAR_TO_MONTH</code>	-	
<code>SQL_INTERVAL_MONTH</code>	<code>INTERVAL_YEAR_TO_MONTH</code>	-	
<code>SQL_INTERVAL_YEAR_TO_MONTH</code>	<code>INTERVAL_YEAR_TO_MONTH</code>	-	

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

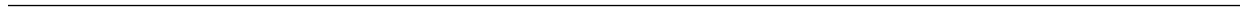
ODBC	Oracle	Criteria	If Oracle uses large varchar (32k)
SQL_INTERVAL_DAY	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_HOU R	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_MIN UTE	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_SEC OND	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_DAY _TO_HOUR	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_DAY _TO_MINUTE	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_DAY _TO_SECOND	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_HOU R_TO_MINUTE	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_HOU R_TO_SECOND	INTERVAL_DAY_TO_SECON D	-	
SQL_INTERVAL_MIN UTE_TO_SECOND	INTERVAL_DAY_TO_SECON D	-	
SQL_LONGVARBINAR Y	LONG RAW	-	
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.	4000 < N <= 32740	N <= 32767
SQL_NUMERIC(p[,s )	NUMBER(p[,s])	-	
SQL_REAL	FLOAT(24)	-	
SQL_SMALLINT	NUMBER(5)	-	
SQL_TYPE_TIME	CHAR(15)	-	
SQL_TINYINT	NUMBER(3)	-	
SQL_TYPE_DATE	DATE	-	

---

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

<b>ODBC</b>	<b>Oracle</b>	<b>Criteria</b>	<b>If Oracle uses large varchar (32k)</b>
SQL_TYPE_TIMESTAMP	DATE	-	
SQL_VARBINARY	RAW	-	
SQL_VARCHAR	VARCHAR2	N <= 4000	N <= 32767
SQL_WCHAR	NCHAR	-	
SQL_WVARCHAR	NVARCHAR	-	
SQL_WLONGVARCHAR	LONG	if Oracle DB Character Set = Unicode. Otherwise, it is not supported	
SQL_BIT	NUMBER (3)	-	

---



---

## Supported SQL Syntax and Functions

The following topics describe SQL syntax and functions supported by Oracle Database Gateway for ODBC.

Topics:

[Supported SQL Statements](#) (page B-1)

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:

[Oracle Functions](#) (page B-2)

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, Oracle Database Gateway for ODBC does not support many Oracle functions, possibly impacting performance.

### B.1 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.

---

---

[DELETE](#) (page B-2)

[INSERT](#) (page B-2)

[SELECT](#) (page B-2)

[UPDATE](#) (page B-2)

---

---

**See Also:**

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

---

---

## B.1.1 DELETE

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

## B.1.2 INSERT

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

## B.1.3 SELECT

The `SELECT` statement is fully supported, with these exceptions:

- `CONNECT BY condition`
- `NOWAIT`
- `START WITH condition`
- `WHERE CURRENT OF`
- `FOR UPDATE`

## B.1.4 UPDATE

The `UPDATE` statement is fully supported. However, only Oracle functions supported by the 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.

## B.2 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, Oracle Database Gateway for ODBC does not support many Oracle functions, 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 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 ODBC driver provider that is being used supports the following minimum set of SQL functions:

- `AVG(exp)`
- `LIKE(exp)`
- `COUNT(*)`
- `MAX(exp)`
- `MIN(exp)`
- `NOT`





---

## Data Dictionary

Data dictionary information is stored in the non-Oracle system as system tables and is accessed through ODBC application programming interfaces (APIs).

The following topics explain how to access non-Oracle data dictionaries, use supported views and tables, and data dictionary mapping:

[Accessing the Non-Oracle Data Dictionary](#) (page C-1)

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.

[Views and Tables Supported by Oracle Database Gateway for ODBC](#) (page C-2)

Oracle Database Gateway for ODBC supports only the views and tables shown in the table, Oracle Database Gateway for ODBC Data Dictionary Mapping.

### C.1 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 [Data Dictionary Mapping](#) (page C-3)). 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.

---

---

## C.2 Views and Tables Supported by Oracle Database Gateway for ODBC

Oracle Database Gateway for ODBC supports only the views and tables shown in the table, Oracle Database Gateway for ODBC Data Dictionary Mapping.

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](#) (page C-3)

The tables in this section list Oracle data dictionary view names and the equivalent ODBC APIs used.

[ALL\\_CATALOG](#) (page C-4)

[ALL\\_COL\\_COMMENTS](#) (page C-4)

[ALL\\_CONS\\_COLUMNS](#) (page C-5)

[ALL\\_CONSTRAINTS](#) (page C-5)

[ALL\\_IND\\_COLUMNS](#) (page C-6)

[ALL\\_INDEXES](#) (page C-6)

[ALL\\_OBJECTS](#) (page C-8)

[ALL\\_TAB\\_COLUMNS](#) (page C-9)

[ALL\\_TAB\\_COMMENTS](#) (page C-10)

[ALL\\_TABLES](#) (page C-11)

[ALL\\_USERS](#) (page C-12)

[ALL\\_VIEWS](#) (page C-13)

[DICTIONARY](#) (page C-13)

[DICT\\_COLUMNS](#) (page C-14)

[USER\\_CATALOG](#) (page C-14)

[USER\\_COL\\_COMMENTS](#) (page C-14)

[USER\\_CONS\\_COLUMNS](#) (page C-15)[USER\\_CONSTRAINTS](#) (page C-15)[USER\\_IND\\_COLUMNS](#) (page C-16)[USER\\_INDEXES](#) (page C-16)[USER\\_OBJECTS](#) (page C-18)[USER\\_TABCOLUMNS](#) (page C-19)[USER\\_TAB\\_COMMENTS](#) (page C-20)[USER\\_TABLES](#) (page C-20)[USER\\_USERS](#) (page C-22)[USER\\_VIEWS](#) (page C-23)

## C.2.1 Data Dictionary Mapping

The tables in this section list Oracle data dictionary view names and the equivalent ODBC APIs used.

[Table C-1](#) (page C-3) 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
<a href="#">Table C-2</a> (page C-4)	SQLTables
<a href="#">Table C-3</a> (page C-4)	SQLColumns
<a href="#">Table C-4</a> (page C-5)	SQLPrimaryKeys, SQLForeignKeys
<a href="#">Table C-5</a> (page C-5)	SQLPrimaryKeys, SQLForeignKeys
<a href="#">Table C-6</a> (page C-6)	SQLStatistics
<a href="#">Table C-7</a> (page C-7)	SQLStatistics
<a href="#">Table C-8</a> (page C-9)	SQLTables, SQLProcedures, SQLStatistics
<a href="#">Table C-9</a> (page C-9)	SQLColumns
<a href="#">Table C-10</a> (page C-10)	SQLTables
<a href="#">Table C-11</a> (page C-11)	SQLStatistics
<a href="#">Table C-12</a> (page C-12)	SQLTables
<a href="#">Table C-13</a> (page C-13)	SQLTables
<a href="#">Table C-14</a> (page C-13)	SQLTables
<a href="#">Table C-15</a> (page C-14)	SQLTables
<a href="#">Table C-17</a> (page C-14)	SQLColumns
<a href="#">Table C-18</a> (page C-15)	SQLPrimaryKeys, SQLForeignKeys

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

View	ODBC API
<a href="#">Table C-19</a> (page C-15)	SQLPrimaryKeys, SQLForeignKeys
<a href="#">Table C-20</a> (page C-16)	SQLStatistics
<a href="#">Table C-21</a> (page C-16)	SQLStatistics
<a href="#">Table C-22</a> (page C-18)	SQLTables, SQLProcedures, SQLStatistics
<a href="#">Table C-23</a> (page C-19)	SQLColumns
<a href="#">Table C-24</a> (page C-20)	SQLTables
<a href="#">Table C-25</a> (page C-21)	SQLStatistics
<a href="#">Table C-26</a> (page C-22)	SQLTables
<a href="#">Table C-27</a> (page C-23)	SQLTables

## C.2.2 ALL\_CATALOG

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

**Table C-2 ALL\_CATALOG**

Name	Type	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(11)	"TABLE" or "VIEW" or "SYNONYM"

## C.2.3 ALL\_COL\_COMMENTS

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

**Table C-3 ALL\_COL\_COMMENTS**

Name	Type	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-

**Table C-3 (Cont.) ALL\_COL\_COMMENTS**

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

## C.2.4 ALL\_CONS\_COLUMNS

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

**Table C-4 ALL\_CONS\_COLUMNS**

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

## C.2.5 ALL\_CONSTRAINTS

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

**Table C-5 ALL\_CONSTRAINTS**

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

**Table C-5 (Cont.) ALL\_CONSTRAINTS**

Name	Type	Value
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

## C.2.6 ALL\_IND\_COLUMNS

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

**Table C-6 ALL\_IND\_COLUMNS**

Name	Type	Value
INDEX_OWNER	VARCHAR2(30)	-
INDEX_NAME	VARCHAR2(30)	-
TABLE_OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(4000)	-
COLUMN_POSITION	NUMBER	-
COLUMN_LENGTH	NUMBER	-
DESCEND	VARCHAR2(4)	"DESC" or "ASC"

## C.2.7 ALL\_INDEXES

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

**Table C-7 ALL\_INDEXES**

Name	Type	Value
OWNER	VARCHAR2 (30)	-
INDEX_NAME	VARCHAR2 (30)	-
INDEX_TYPE	VARCHAR2 (27)	NULL
TABLE_OWNER	VARCHAR2 (30)	-
TABLE_NAME	VARCHAR2 (30)	-
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

**Table C-7 (Cont.) ALL\_INDEXES**

<b>Name</b>	<b>Type</b>	<b>Value</b>
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
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

## C.2.8 ALL\_OBJECTS

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



**Table C-8 ALL\_OBJECTS**

Name	Type	Value
OWNER	VARCHAR2(30)	-
OBJECT_NAME	VARCHAR2(30)	-
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

### C.2.9 ALL\_TAB\_COLUMNS

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

**Table C-9 ALL\_TAB\_COLUMNS**

Name	Type	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
DATA_TYPE	VARCHAR2(106)	-
DATA_TYPE_MOD	VARCHAR2(3)	NULL
DATA_TYPE_OWNER	VARCHAR2(30)	NULL
DATA_LENGTH	NUMBER	-
DATA_PRECISION	NUMBER	-

**Table C-9 (Cont.) ALL\_TAB\_COLUMNS**

Name	Type	Value
DATA_SCALE	NUMBER	-
NULLABLE	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	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

## C.2.10 ALL\_TAB\_COMMENTS

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

**Table C-10 ALL\_TAB\_COMMENTS**

Name	Type	Value
OWNER	VARCHAR2(30)	-
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(11)	"TABLE" or "VIEW"
COMMENTS	VARCHAR2(4000)	NULL

## C.2.11 ALL\_TABLES

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

**Table C-11 ALL\_TABLES**

Name	Type	Value
OWNER	VARCHAR2 ( 30 )	-
TABLE_NAME	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
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

**Table C-11 (Cont.) ALL\_TABLES**

Name	Type	Value
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	VARHCHAR2 ( 15 )	NULL
SKIP_CORRUPT	VARCHAR2 ( 8 )	NULL
MONITORING	VARCHAR2 ( 3 )	NULL

## C.2.12 ALL\_USERS

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

**Table C-12 ALL\_USERS**

Name	Type	Value
USERNAME	VARCHAR2 ( 30 )	-
USER_ID	NUMBER	0

**Table C-12 (Cont.) ALL\_USERS**

Name	Type	Value
CREATED	DATE	NULL

### C.2.13 ALL\_VIEWS

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

**Table C-13 ALL\_VIEWS**

Name	Type	Value
OWNER	VARCHAR2 ( 30 )	-
VIEW_NAME	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

### C.2.14 DICTIONARY

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

**Table C-14 DICTIONARY**

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

## C.2.15 DICT\_COLUMNS

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

**Table C-15 DICT\_COLUMNS**

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

## C.2.16 USER\_CATALOG

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

**Table C-16 USER\_CATALOG**

Name	Type	Value
TABLE_NAME	VARCHAR2(30)	-
TABLE_TYPE	VARCHAR2(11)	"TABLE" or "VIEW" or "SYNONYM"

## C.2.17 USER\_COL\_COMMENTS

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

**Table C-17 USER\_COL\_COMMENTS**

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

## C.2.18 USER\_CONS\_COLUMNS

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

**Table C-18 USER\_CONS\_COLUMNS**

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

## C.2.19 USER\_CONSTRAINTS

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

**Table C-19 USER\_CONSTRAINTS**

Name	Type	Value
OWNER	VARCHAR2 ( 30 )	-
CONSTRAINT_NAME	VARCHAR2 ( 30 )	-
CONSTRAINT_TYPE	VARCHAR2 ( 1 )	R or P
TABLE_NAME	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

**Table C-19 (Cont.) USER\_CONSTRAINTS**

Name	Type	Value
VALIDATED	VARCHAR2 ( 13 )	NULL
GENERATED	VARCHAR2 ( 14 )	NULL
BAD	VARCHAR2 ( 3 )	NULL
RELY	VARCHAR2 ( 4 )	NULL
LAST_CHANGE	DATE	NULL

## C.2.20 USER\_IND\_COLUMNS

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

**Table C-20 USER\_IND\_COLUMNS**

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

## C.2.21 USER\_INDEXES

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

**Table C-21 USER\_INDEXES**

Name	Type	Value
INDEX_NAME	VARCHAR2 ( 30 )	-
INDEX_TYPE	VARCHAR2 ( 27 )	NULL
TABLE_OWNER	VARCHAR2 ( 30 )	-



**Table C-21 (Cont.) USER\_INDEXES**

<b>Name</b>	<b>Type</b>	<b>Value</b>
TABLE_NAME	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
NUM_ROWS	NUMBER	0
SAMPLE_SIZE	NUMBER	0
LAST_ANALYZED	DATE	NULL

**Table C-21 (Cont.) USER\_INDEXES**

Name	Type	Value
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	VARHCHAR2 ( 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

## C.2.22 USER\_OBJECTS

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

**Table C-22 USER\_OBJECTS**

Name	Type	Value
OBJECT_NAME	VARCHAR2 ( 128 )	-
SUBOBJECT_NAME	VARCHAR2 ( 30 )	NULL
OBJECT_ID	NUMBER	0
DATA_OBJECT_ID	NUMBER	0

**Table C-22 (Cont.) USER\_OBJECTS**

Name	Type	Value
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

### C.2.23 USER\_TABCOLUMNS

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

**Table C-23 USER\_TABCOLUMNS**

Name	Type	Value
TABLE_NAME	VARCHAR2(30)	-
COLUMN_NAME	VARCHAR2(30)	-
DATA_TYPE	VARCHAR2(106)	-
DATA_TYPE_MOD	VARCHAR2(3)	NULL
DATA_TYPE_OWNER	VARCHAR2(30)	NULL
DATA_LENGTH	NUMBER	-
DATA_PRECISION	NUMBER	-
DATA_SCALE	NUMBER	-
NULLABLE	VARCHAR2(1)	"Y" or "N"
COLUMN_ID	NUMBER	-
DEFAULT_LENGTH	NUMBER	NULL
DATA_DEFAULT	LONG	NULL
NUM_DISTINCT	NUMBER	NULL

**Table C-23 (Cont.) USER\_TABCOLUMNS**

Name	Type	Value
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

## C.2.24 USER\_TAB\_COMMENTS

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

**Table C-24 USER\_TAB\_COMMENTS**

Name	Type	Value
TABLE_NAME	VARCHAR2( 30 )	-
TABLE_TYPE	VARCHAR2( 11 )	"TABLE" or "VIEW"
COMMENTS	VARCHAR2( 4000 )	NULL

## C.2.25 USER\_TABLES

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

**Table C-25 USER\_TABLES**

Name	Type	Value
TABLE_NAME	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
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

**Table C-25 (Cont.) USER\_TABLES**

<b>Name</b>	<b>Type</b>	<b>Value</b>
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
USER_STATS	VARCHAR2 ( 3 )	NULL
DURATION	VARCHAR2 ( 15 )	NULL
SKIP_CORRUPT	VARCHAR2 ( 8 )	NULL
MONITORING	VARCHAR2 ( 3 )	NULL

## C.2.26 USER\_USERS

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

**Table C-26 USER\_USERS**

<b>Name</b>	<b>Type</b>	<b>Value</b>
USERNAME	VARCHAR2 ( 30 )	-
USER_ID	NUMBER	0
ACCOUNT_STATUS	VARCHAR2 ( 32 )	OPEN
LOCK_DATE	DATE	NULL
EXPIRY_DATE	DATE	NULL
DEFAULT_TABLESPACE	VARCHAR2 ( 30 )	NULL
TEMPORARY_TABLESPACE	VARCHAR2 ( 30 )	NULL
CREATED	DATE	NULL

**Table C-26 (Cont.) USER\_USERS**

Name	Type	Value
INITIAL_RSRC_CONSUMER_G ROUP	VARCHAR2(30)	NULL
EXTERNAL_NAME	VARCHAR2(4000)	NULL

## C.2.27 USER\_VIEWS

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

**Table C-27 USER\_VIEWS**

Name	Type	Value
VIEW_NAME	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





---

# 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 `DBMS_HS` 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](#) (page D-1)

This topic explains the syntax for the initialization parameter file.

[Oracle Database Gateway for ODBC Initialization Parameters](#) (page D-2)

This topic lists the initialization file parameters that can be set for the Oracle Database Gateway for ODBC.

## D.1 Initialization Parameter File Syntax

This topic explains the syntax for the initialization parameter file.

- 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 `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 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 Oracle Database Gateway for ODBC Initialization Parameters

This topic lists the initialization file parameters that can be set for the Oracle Database Gateway for ODBC.

[HS\\_DB\\_DOMAIN](#) (page D-4)

Specifies a unique network sub-address for a non-Oracle system.

[HS\\_DB\\_INTERNAL\\_NAME](#) (page D-4)

Specifies a unique hexadecimal number identifying the instance to which the Heterogeneous Services agent is connected.

[HS\\_DB\\_NAME](#) (page D-5)

Specifies a unique alphanumeric name for the data store given to the non-Oracle system.

**HS\_DESCRIBE\_CACHE\_HWM** (page D-5)

Specifies the maximum number of entries in the describe cache used by Heterogeneous Services.

**HS\_LANGUAGE** (page D-5)

Provides Heterogeneous Services with character set, language, and territory information of the non-Oracle data source.

**HS\_LONG\_PIECE\_TRANSFER\_SIZE** (page D-6)

Sets the size of the piece of LONG data being transferred.

**HS\_OPEN\_CURSORS** (page D-7)

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

**HS\_RPC\_FETCH\_REBLOCKING** (page D-7)

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.

**HS\_RPC\_FETCH\_SIZE** (page D-8)

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

**HS\_TIME\_ZONE** (page D-8)

Specifies the default local time zone displacement for the current SQL session.

**HS\_TRANSACTION\_MODEL** (page D-8)

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

**IFILE** (page D-9)

Use the IFILE initialization parameter to embed another initialization file within the current initialization file.

**HS\_FDS\_TIMESTAMP\_MAPPING** (page D-9)

Maps non-Oracle timestamps to Oracle timestamps.

**HS\_FDS\_DATE\_MAPPING** (page D-9)

Maps non-Oracle target dates to Oracle target dates.

**HS\_FDS\_CONNECT\_INFO** (page D-10)

HS\_FDS\_CONNECT\_INFO describes the connection to the non-Oracle system.

**HS\_FDS\_DEFAULT\_OWNER** (page D-10)

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.

**HS\_FDS\_TRACE\_LEVEL** (page D-10)

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

**HS\_FDS\_SHAREABLE\_NAME** (page D-11)

Specifies the full path name to the ODBC driver manager.

**HS\_FDS\_FETCH\_ROWS** (page D-11)

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.

**HS\_FDS\_REMOTE\_DB\_CHARSET** (page D-11)

This parameter is valid only when HS\_LANGUAGE is set to AL32UTF8 and the gateway runs on Windows.

**HS\_FDS\_SQLLEN\_INTERPRETATION** (page D-12)

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.

**HS\_FDS\_REPORT\_REAL\_AS\_DOUBLE** (page D-12)

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

## D.2.1 HS\_DB\_DOMAIN

Specifies a unique network sub-address for a non-Oracle system.

---

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

---

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.

---

---

## D.2.2 HS\_DB\_INTERNAL\_NAME

Specifies a unique hexadecimal number identifying the instance to which the Heterogeneous Services agent is connected.

---

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

---

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.2.3 HS\_DB\_NAME

Specifies a unique alphanumeric name for the data store given to the non-Oracle system.

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

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.

### D.2.4 HS\_DESCRIBE\_CACHE\_HWM

Specifies the maximum number of entries in the describe cache used by Heterogeneous Services.

Property	Description
Default value	100
Range of values	1 to 4000

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.2.5 HS\_LANGUAGE

Provides Heterogeneous Services with character set, language, and territory information of the non-Oracle data source.

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

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](#) (page D-6)

[Language](#) (page D-6)

[Territory](#) (page D-6)

### D.2.5.1 Character Sets

Ideally, the character sets of the Oracle database and the non-Oracle data source are the same. In almost all cases, `HS_LANGUAGE` 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.

---

---

**Note:**

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](#) (page D-11) for instruction on how to adjust to correct sizes.

### D.2.5.2 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.

### D.2.5.3 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.

## D.2.6 HS\_LONG\_PIECE\_TRANSFER\_SIZE

Sets the size of the piece of `LONG` data being transferred.

---

Property	Description
Default value	64 KB

---

Property	Description
Range of values	Any value up to 2 GB

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.2.7 HS\_OPEN\_CURSORS

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

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

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.

## D.2.8 HS\_RPC\_FETCH\_REBLOCKING

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.

Property	Description
Default value	ON
Range of values	OFF or ON

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.2.9 HS\_RPC\_FETCH\_SIZE

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

Property	Description
Default value	50000
Range of values	1 to 10000000

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.2.10 HS\_TIME\_ZONE

Specifies the default local time zone displacement for the current SQL session.

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

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.2.11 HS\_TRANSACTION\_MODEL

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

Property	Description
Default Value	None
Range of Values	READ_ONLY and SINGLE_SITE

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 and cannot be used with Oracle Streams to replicate data.



## D.2.12 IFILE

Use the IFILE initialization parameter to embed another initialization file within the current initialization file.

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

The value should be an absolute path and should not contain environment variables. The three levels of nesting limit do not apply.

---



---

**See Also:**

*Oracle Database Reference*

---



---

## D.2.13 HS\_FDS\_TIMESTAMP\_MAPPING

Maps non-Oracle timestamps to Oracle timestamps.

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

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

## D.2.14 HS\_FDS\_DATE\_MAPPING

Maps non-Oracle target dates to Oracle target dates.

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.2.15 HS\_FDS\_CONNECT\_INFO

HS\_FDS\_CONNECT\_INFO describes the connection to the non-Oracle system.

---

Property	Description
Default Value	None
Range of Values	Not applicable

---

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

The entry for *dsn\_value* is case sensitive.

## D.2.16 HS\_FDS\_DEFAULT\_OWNER

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.

---

Property	Description
Default Value	None
Range of Values	Not applicable

---

---

**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.

---

## D.2.17 HS\_FDS\_TRACE\_LEVEL

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

---

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

---

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.2.18 HS\_FDS\_SHAREABLE\_NAME

Specifies the full path name to the ODBC driver manager.

Property	Description
Default Value	None
Range of Values	Not applicable

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.

This parameter applies only to UNIX based platforms.

### D.2.19 HS\_FDS\_FETCH\_ROWS

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.

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

This parameter will be affected by the HS\_RPC\_FETCH\_SIZE and HS\_RPC\_FETCH\_REBLOCKING parameters.

### D.2.20 HS\_FDS\_REMOTE\_DB\_CHARSET

This parameter is valid only when HS\_LANGUAGE is set to AL32UTF8 and the gateway runs on Windows.

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

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. To compensate, the gateway can adjust to correct size if `HS_FDS_REMOTE_DB_CHARSET` is set to the corresponding non-Oracle database character set. For example, `HS_FDS_REMOTE_DB_CHARSET=KO16KSC5601`.

### D.2.21 HS\_FDS\_SQLLEN\_INTERPRETATION

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.

Property	Description
Default Value	64
Range of values	{64 32}
Syntax	<code>HS_FDS_SQLLEN_INTERPRETATION= {64 32}</code>

In order for Oracle Database Gateway for ODBC to compensate their behavior, you need to specify `HS_FDS_SQLLEN_INTERPRETATION=32` if you use these types of driver managers and driver.

### D.2.22 HS\_FDS\_REPORT\_REAL\_AS\_DOUBLE

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

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

The default value is `FALSE`.

---

---

# Index

## A

---

ALTER statement, [B-1](#)

## C

---

character sets

Heterogeneous Services, [D-6](#)

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-2](#)

mapping for Oracle Database Gateway for ODBC, [C-3](#)

Oracle database name/SQL Server name, [C-3](#)

translation support for Oracle Database Gateway for ODBC, [C-1](#)

data dictionary views

Oracle Database Gateway for ODBC, [C-2](#)

data type

VARBINARY, [2-4](#)

DELETE statement, [B-2](#)

describe cache high water mark

definition, [D-5](#)

drivers

ODBC, [1-6](#)

DROP statement, [B-1](#)

## E

---

Encrypted format login, [2-4](#)

Error messages

error tracing, [D-10](#)

## F

---

fetch array size, with HS\_FDS\_FETCH\_ROWS, [D-11](#)

## G

---

gateway

pass-through feature, [2-1](#)

supported functions, [B-1](#)

supported SQL syntax, [B-1](#)

globalization support

Heterogeneous Services, [D-5](#)

GRANT statement, [B-1](#)

## H

---

Heterogeneous Services

defining maximum number of open cursors, [D-7](#)

optimizing data transfer, [D-7](#)

Oracle Database Gateway for ODBC

architecture, [1-3](#)

definition, [1-2](#)

non-Oracle data dictionary access, [C-1](#)

ODBC connectivity requirements, [1-6](#)

supported functions, [B-3](#)

supported SQL syntax, [B-1](#)

supported tables, [C-2](#)

setting global name, [D-5](#)

specifying cache high water mark, [D-5](#)

tuning internal data buffering, [D-8](#)

tuning LONG data transfer, [D-6](#)

HS\_DB\_NAME initialization parameter, [D-5](#)

HS\_DESCRIBE\_CACHE\_HWM initialization parameter, [D-5](#)

HS\_FDS\_CONNECT\_INFO, [D-10](#)

HS\_FDS\_DEFAULT\_OWNER initialization parameter, [D-10](#)

HS\_FDS\_FETCH\_ROWS parameter, [D-11](#)

HS\_FDS\_RECOVERY\_PWD initialization parameter, [D-12](#)

HS\_FDS\_SHAREABLE\_NAME initialization parameter, [D-11](#)

HS\_FDS\_TRACE\_LEVEL initialization parameter enabling agent tracing, [D-2](#)

HS\_LANGUAGE initialization parameter, [D-5](#)

HS\_LONG\_PIECE\_TRANSFER\_SIZE initialization parameter, [D-6](#)

HS\_OPEN\_CURSORS initialization parameter, [D-7](#)  
HS\_RPC\_FETCH\_REBLOCKING initialization  
parameter, [D-7](#)  
HS\_RPC\_FETCH\_SIZE initialization parameter, [D-8](#)  
HS\_TIME\_ZONE initialization parameter, [D-8](#)

---

## I

IFILE initialization parameter, [D-9](#)  
Initialization parameter file  
customizing, [D-1](#)  
INSERT statement, [B-2](#)

---

## K

Known restrictions, [2-2](#)

---

## O

ODBC agents  
connectivity requirements, [1-6](#)  
functions, [1-7](#)  
ODBC connectivity  
data dictionary mapping, [C-3](#)  
ODBC driver, [1-6](#)  
requirements, [1-6](#)  
specifying path to library, [D-11](#)  
OLE DB connectivity  
data dictionary mapping, [C-3](#)  
Oracle Database Gateway for ODBC  
architecture  
Oracle and non-Oracle on same machine, [1-5](#)  
Oracle and non-Oracle on separate machines,  
[1-3](#)  
data dictionary  
translation support, [C-1](#)

Oracle Database Gateway for ODBC (*continued*)  
data dictionary (*continued*)  
definition, [1-2](#)  
non-Oracle data dictionary access, [C-1](#)  
ODBC connectivity requirements, [1-6](#)  
supported functions, [B-3](#)  
supported SQL syntax, [B-1](#)

---

## P

parameters  
gateway initialization file  
HS\_FDS\_FETCH\_ROWS, [D-11](#)

---

## R

ROWID, [2-3](#)

---

## S

SELECT statement, [B-2](#)

---

## T

TRUNCATE statement, [B-1](#)

---

## U

UPDATE statement, [B-2](#)

---

## V

VARBINARY data type, [2-4](#)

---

## W

WHERE CURRENT OF clause, [2-3](#)