Oracle7 Driver

For All Users

The following topics discuss the Oracle7 driver and how to install it for use by an application.

Overview

Driver Conformance Levels

Hardware and Software Requirements

Setting Up the Oracle Driver

Adding, Modifying, and Deleting Oracle7 Data Sources

Connecting to an Oracle7 Data Source

Troubleshooting

For Advanced Users

The following topics discuss how to use the Oracle driver directly.

Connection Strings (Advanced)

SQL Statements (Advanced)

Data Types (Advanced)

Error Messages (Advanced)

Initialization and Configuration Files (Advanced)

For Programmers

The following topics discuss how to use the Oracle driver programmatically. They are intended for application programmers and require knowledge of the Open Database Connectivity (ODBC) application programming interface (API).

SQLGetInfo Return Values (Programming)

ODBC API Functions (Programming)

Implementation Issues (Programming)

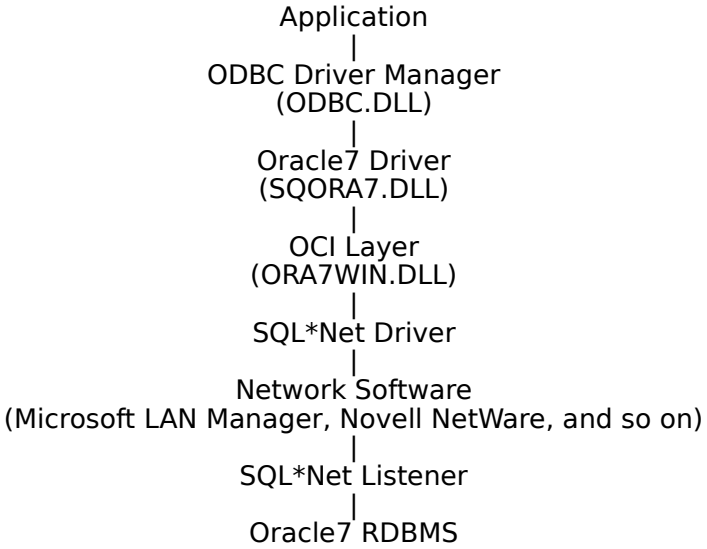
Overview

See Also

Oracle RDBMS is a multiuser relational database management system (DBMS) that runs on a variety of workstation and minicomputer operating systems. Structured Query Language (SQL) is used to access data in Oracle RDBMS. IBM PCs and compatibles running Microsoft Windows communicate with Oracle7 database servers across a network such as Microsoft LAN Manager, Novell NetWare, Banyan VINES, DECnet, or any TCP/IP network.

The Oracle7 driver enables applications to access data in an Oracle7 database through the Open Database Connectivity (ODBC) interface. It communicates with the network through SQL*Net.

The application/driver architecture is:



See Also

For All Users

- Adding, Modifying, and Deleting Oracle7 Data Sources
- Connecting to an Oracle7 Data Source
- Driver Conformance Levels
- Hardware and Software Requirements
- Setting Up the Oracle7 Driver

Driver Conformance Levels

See Also

The Oracle7 driver has the following conformance levels:

- API Conformance Level: Level 1
- SQL Conformance Level: Minimum

Note The Oracle7 driver also supports some of the functions in the level 2 API conformance level and almost all of the grammar in the core and extended SQL conformance levels.

To determine the minimum conformance levels (if any) drivers must have to be used with your application, see your application's documentation.

See Also

For Advanced Users

- Data Types (Advanced)
- SQL Statements (Advanced)

For Programmers

- ODBC API Functions (Programming)
- SQLGetInfo Return Values (Programming)

Hardware and Software Requirements

See Also

To access Oracle7 data, you must have:

- The Oracle7 driver.
- An Oracle7 RDBMS database.
- A network connecting the computers on which these reside and a SQL*Net connection across that network.

The following paragraphs describe the hardware and software required by each of these components.

Oracle7 Driver

The Oracle7 driver requires the following hardware:

- An Industry Standard Architecture (ISA) computer, such as the IBM PC/AT or compatible, or
- A Micro Channel Architecture (MCA) computer, such as an IBM PS/2 or compatible, or
- An Extended Industry Standard Architecture (EISA) computer with an 80386 or higher microprocessor.
- At least 2.5 megabytes of random-access memory (RAM); 4 MB of RAM are recommended.
- A hard disk drive and approximately 350 kilobytes of hard disk space for the Oracle7 driver if ODBC Setup is already installed, or 1 megabyte of hard disk space for a full installation.

The Oracle7 driver requires the following software:

- MS-DOS version 5.0 (or higher) or DR-DOS version 6 (or higher)
- Microsoft Windows version 3.1 or later
- ODBC Driver Manager version 1.0 (ODBC.DLL)
- Oracle7 Call Interface (OCI) dynamic-link library (ORA7WIN.DLL). Note that this file comes with ORACLE SQL*Net for windows.

Note The Oracle7 driver requires ORA7WIN.DLL, regardless of whether a later version of OCI has been installed. ORA7WIN.DLL is shipped with the SQL*Net for Windows.

Oracle7 RDBMS

To access data in an Oracle7 database with the Oracle7 driver, you must have Oracle RDBMS version 7 or later. For information about the hardware and software required by Oracle7 RDBMS, see the Oracle documentation.

Network Software

A network is required to connect the platforms on which Oracle7 RDBMS and the Oracle7 driver reside. For information about the hardware and software required by each network, see that network's documentation.

Note If you are using Microsoft LAN Manager with the named pipes protocol, it must be version 2.0 or later. If you are using Microsoft LAN Manager with the TCP/IP protocol, it must be version 2.2 or later.

SQL*Net

The Oracle7 driver and Oracle7 RDBMS use SQL*Net as a common network interface; it should be possible to use the Oracle7 driver on any SQL*Net connection from a computer running Windows to an Oracle7 database server.

For complete information about SQL*Net, see the SQL*Net documentation.

Note SQL*Net components are available only from Oracle Corporation.

See Also

For All Users

Setting Up the Oracle7 Driver

SQL*Net

SQL*Net is a network interface package from Oracle Corporation that allows applications to access data in Oracle RDBMS across a variety of networks.

Setting Up the Oracle7 ODBC Driver

See Also

To set up the Oracle7 ODBC Driver

1. If you have not already done so, install the ORACLE SQL*Net client components on your computer. If you want to test your ORACLE SQL*Net connection, set up your computer as an Oracle7 client workstation and run SQL*Plus or NetTest. For information on how to do this, see the SQL*Net documentation.
2. Add a data source for each copy of Oracle7 RDBMS in which you want to access data.

To set up a new version of the Oracle7 driver

1. In the Main group in the Program Manager window, double-click the Control Panel icon. In the Control Panel window, double-click the ODBC icon.

   The Data Sources dialog box is displayed.
2. In the Data Sources dialog box, choose the Drivers button.

   The Drivers dialog box is displayed.
3. In the Drivers dialog box, choose the Add button.

   The Add Driver dialog box is displayed.
4. In the text box, type the name of the drive and directory containing the Oracle7 driver in the text box. Or choose the Browse button to select a drive and directory name.
5. In the Add Driver dialog box, choose the OK button.

   The Install Drivers dialog box is displayed.
6. In the Available ODBC Drivers list, select Oracle7.
7. Choose the OK button.

   The Oracle7 driver is installed.

To delete the Oracle7 driver

1. In the Main group in the Program Manager window, double-click the Control Panel icon. In the Control Panel window, double-click the ODBC icon.

   The Data Sources dialog box is displayed.
2. In the Data Sources dialog box, choose the Drivers button.

   The Drivers dialog is displayed.
3. In the Installed ODBC Drivers list, select Oracle7.
4. Choose the Delete button.

   A message asks you to confirm that you want to remove the driver and all of the data sources that use the driver.
5. Choose the Yes button.

See Also

For All Users

Adding, Modifying, and Deleting Oracle7 Data Sources

Hardware and Software Requirements

For Advanced Users

Initialization and Configuration Files (Advanced)

Data Source (Oracle)

A data source includes the data a user wants to access and the information needed to get to that data. For the Oracle7 driver, a data source is a specific copy of Oracle7 RDBMS, the computer on which it resides, the network used to access that computer, and the SQL*Net components used as an interface to that network.

Adding, Modifying, and Deleting Oracle7 Data Sources

See Also

Before you can access data with the Oracle7 driver, you must add a data source for each of your copies of Oracle7 RDBMS. The Oracle7 driver uses the information you enter when you add the data source to access the data. You can change or delete a data source at any time.

To add an Oracle7 data source

1. In the Main group in the Program Manager window, double-click the Control Panel icon. In the Control Panel window, double-click the ODBC icon.
2. In the Data Sources dialog box, choose the Add button.

   The Add Data Source dialog box is displayed.
3. In the Installed ODBC Drivers list, select Oracle7 and choose the OK button.

   The Oracle7 ODBC Driver dialog box is displayed.
4. In the Oracle7 ODBC Driver dialog box, set the option values as necessary and choose the OK button.

To modify an Oracle7 data source

1. In the Main group in the Program Manager window, double-click the Control Panel icon. In the Control Panel window, double-click the ODBC icon.
2. In the Data Sources dialog box, select the data source in the Data Sources list and choose the Setup button.

   The Oracle7 ODBC Driver dialog box is displayed.
3. In the Oracle7 ODBC Driver dialog box, set the option values as necessary and choose the OK button.

To delete an Oracle7 data source

1. In the Main group in the Program Manager window, double-click the Control Panel icon. In the Control Panel window, double-click the ODBC icon.
2. In the Data Sources dialog box, select the data source you want to delete in the Data Sources list.
3. Choose the Delete button, and then choose the Yes button to confirm the deletion.

See Also

For All Users

- Connecting to an Oracle7 Data Source
- Setting Up the Oracle7 Driver

For Advanced Users

- Initialization and Configuration Files (Advanced)

Connecting to an Oracle7 Data Source

See Also

To connect to a data source, the Oracle7 ODBC Driver requires that an ORACLE SQL*Net driver is installed on your computer and the corresponding ORACLE SQL*Net listener be running on the Oracle7 database server. ORACLE SQL*Net for windows is a Dynamic Linked Library (DLL) based application. For more information about ORACLE SQL*Net, see the ORACLE SQL*Net documentation.

As part of the connection process, an application can prompt you for information. If an application prompts you for information about an Oracle data source, do the following:

1 In the User Name box, type the name you use on Oracle7 RDBMS.

2 In the Password box, type the password you use on Oracle7 RDBMS.

3 Choose OK.

An application must connect to a data source to access data in it. Different applications connect to data sources at different times. For example, an application might connect to a data source only at your request, or it might connect automatically when it starts. For information about when an application connects to a data source, see the documentation for that application.

See Also

For All Users

Adding, Modifying, and Deleting Oracle7 Data Sources

For Advanced Users

Connection Strings (Advanced)

Initialization and Configuration Files (Advanced)

For Programmers

SQLDriverConnect Implementation (Programming)

Troubleshooting

The following paragraph discuss how to solve problems you might encounter while using the Oracle7 driver.

"Message send failure" error while running Microsoft LAN Manager using TCP/IP

On Microsoft LAN Manager using the TCP/IP protocol, the number of bytes of data that can be sent to or retrieved from Oracle7 RDBMS at one time is controlled by the MAXSENDSIZE keyword in the [sockets] section of TCPUTILS.INI. If you receive a "Message send failure" error, try increasing this value. This error is most likely to occur when SQL_LONGVARCHAR or SQL_LONGVARBINARY data is being sent. If this error message occurs decrease the MAXSENDSIZE parameter in TCPUTILS.INI. For more information, see the Oracle documentation.

Oracle7 ODBC Driver Dialog Box

The Oracle7 ODBC Driver dialog box has the following options.

Data Source Name

A name by which you will identify the data source. For example, "Personnel Data."

Description

A description of the data in the data source. For example, "Hire date, salary history, and current review of all employees."

Network Address

An Oracle connect string that specifies the location of the copy of Oracle RDBMS from which the driver will retrieve data. An Oracle connect string uses the format:

net_prefix:*hostname*[:*SID*]

The arguments in this format are:

Argument	Meaning
net_prefix	Specifies the SQL*Net driver to use. Some common *net_prefixes* are: B: NetBIOS D: DECnet P: Named pipes T: TCP/IP X: SPX/IPX
hostname	The name or alias of the server on which the Oracle RDBMS resides.
SID	The system ID of the database you want to use on the server (optional).

For more information, see the SQL*Net documentation.

User Name

The user login ID for the data source.

Network Config

The name of the Oracle configuration file. This label is read-only and will differ depending on which version of Oracle RDBMS you are running.

Connection Strings (Advanced)

See Also

The connection string for the Oracle7 driver uses the following keywords.

Keyword	Description
DSN	The name of the data source.
DBQ	An Oracle connect string of the form: *net_prefix*:*hostname*[:*SID*] where *net_prefix* specifies the SQL*Net driver to use, *hostname* specifies the name or alias of the server on which Oracle7 RDBMS resides, and *SID* is the optional system ID of the database you want to use on the server. Some common *net_prefixes* are: B: NetBIOS D: DECnet P: Named pipes T: TCP/IP X: SPX/IPX For more information, see the SQL*Net documentation.
UID	The user login ID.
PWD	The user-specified password.

For example, to connect to the Human Resources data source on the server HRSRVR using the NetBIOS SQL*Net driver, the login ID Smith, and the password Sesame, you would use the following connection string:

```
DSN=Human Resources;DBQ=b:HRSRVR;UID=Smith;PWD=Sesame
```

See Also

For All Users

Connecting to an Oracle Data Source

For Programmers

SQLDriverConnect Implementation (Programming)

SQL Statements (Advanced)

See Also

The Oracle7 driver supports the core SQL grammar with a lot of extensions. In addition to Oracle's grammar, the vendor-specific escape sequences outlined in Appendix C of the ODBC specifications are also supported. In accordance with the design of ODBC, the Oracle7 driver will pass native SQL grammar to Oracle7 RDBMS.

The following Help topics describe the SQL grammar implemented by the Oracle7 driver.

For Advanced Users

Limitations to the ODBC SQL Grammar (Advanced)

Unsupported ODBC SQL Grammar (Advanced)

For Programmers

Implementation of the ODBC SQL Grammar (Programming)

See Also

For Advanced Users

Data Types (Advanced)

For Programmers

SQLGetInfo Return Values (Programming)

Limitations to the ODBC SQL Grammar (Advanced)

The only limitation that the Oracle7 ODBC Driver and Oracle RDBMS impose on the ODBC SQL grammar is statement length.

Statements are limited to roughly 2048 bytes. The Oracle7 driver translates parameter markers to ":c*n*", where *n* is the position number of the marker, starting at 0. The Oracle7 driver also translates ODBC escape clauses to native SQL. The resulting statement must be less than 2048 bytes long. This restriction is imposed by SQL*Net running on Microsoft Windows and could have a different size limit for certain vendors.

Unsupported ODBC SQL Grammar (Advanced)

The Oracle7 driver completely supports all SQL statements and clauses in both the core and extended ODBC grammars, including the Integrity Enhancement Facility (IEF), except:

Statement not supported	Description
DELETE	The WHERE CURRENT OF *cursor-name* clause is not supported (positioned delete statement).
IEF	The CASCADE and RESTRICT clauses in the DROP TABLE, DROP VIEW, and REVOKE statements.
UPDATE	The WHERE CURRENT OF *cursor-name* clause is not supported (positioned update statement).

Implementation of the ODBC SQL Grammar (Programming)

The only noteworthy part of the implementation of the ODBC SQL grammar is the implementation of comparison predicates.

If a comparison predicate has a parameter marker as the second expression in the comparison, and the value of that parameter is set to SQL_NULL_DATA with **SQLSetParam**, the comparison will fail. This is consistent with the null predicate grammar in ODBC SQL.

Data Types (Advanced)

See Also

The Oracle7 driver maps Oracle RDBMS data types to ODBC SQL data types. The following table lists all Oracle RDBMS data types and shows the ODBC SQL data types they are mapped to.

Oracle RDBMS SQL data type	ODBC SQL data type
CHAR	SQL_CHAR*
DATE	SQL_TIMESTAMP
FLOAT	SQL_FLOAT
LONG	SQL_LONGVARCHAR
LONG RAW	SQL_LONGVARBINARY
NUMBER(prec.,scale)	SQL_NUMERIC
RAW	SQL_VARBINARY
VARCHAR	SQL_VARCHAR
VARCHAR2	SQL_VARCHAR

Note All conversions in Appendix D of the *Microsoft ODBC SDK Programmer's Reference* are supported for the ODBC SQL data types listed earlier in this topic.

The following Help topics describe the data types implemented by the Oracle7 driver.

For Advanced Users

Implementation of Data Types (Advanced)

Limitations to Data Types (Advanced)

For Programmers

Implementation of Data Types (Programming)

See Also

For Advanced Users

SQL Statements (Advanced)

Implementation of Data Types (Advanced)

For advanced users, the only noteworthy part of the implementation of the data types is the implementation of SQL_TIMESTAMP.

The DATE data type in Oracle RDBMS has both date and time parts. In the native SQL used by Oracle RDBMS, the time part of a DATE column is not returned unless a query specifically requests it. However, because the Oracle7 driver maps the DATE data type to the SQL_TIMESTAMP data type, the driver always returns both the date and time parts of a DATE column.

Limitations to Data Types (Advanced)

The Oracle7 driver and Oracle RDBMS impose the following limitations on the data types.

Limited data type	Description
Literals	Oracle RDBMS limits literals in SQL statements to 255 bytes.
Number of bytes of data	On Microsoft LAN Manager using the TCP/IP protocol, the number of bytes of data that can be sent to or retrieved from Oracle RDBMS is governed by the MAXSENDSIZE parameter in TCPUTILS.INI.
SQL_LONGVARCHAR	The Oracle's limit for SQL_LONGVARCHAR data (LONG data) is 2,147,483,647 bytes. The limiting factor is the client workstation memory.
SQL_LONGVARCHAR and SQL_LONGVARBINARY	Oracle RDBMS allows only a single long data column per table. The long data types are SQL_LONGVARCHAR (LONG) and SQL_LONGVARBINARY (LONG RAW).

Implementation of Data Types (Programming)

For programmers, the only noteworthy part of the implementation of the data types concerns the CHAR, VARCHAR, and VARCHAR2 data types.

For an *fSqlType* value of SQLVARCHAR, **SQLGetTypeInfo** returns the Oracle7 RDBMS data type VARCHAR2. For an *fSqlType* value of SQL_CHAR, **SQLGetTypeInfo** returns Oracle7 RDBMS data type CHAR.

Error Messages (Advanced)

When an error occurs, the Oracle7 driver returns the native error number, the SQLSTATE (an ODBC error code), and an error message. The driver derives this information both from errors detected by the driver and errors returned by Oracle7 RDBMS.

Native Error

For errors that occur in the data source, the Oracle7 driver returns the native error returned to it by Oracle RDBMS. When the Oracle7 driver or the Driver Manager detects an error, the Oracle7 driver returns the Oracle7 error number.

SQLSTATE

For errors that occur in the data source, the Oracle7 driver maps the returned native error to the appropriate SQLSTATE. When the Oracle7 driver detects an error, it generates the appropriate SQLSTATE. When the Driver Manager detects an error, it generates the appropriate SQLSTATE.

Error Message

For errors that occur in the data source, the Oracle7 driver returns an error message based on the message returned by Oracle7 RDBMS. For errors that occur in the Oracle7 driver or the Driver Manager, the Oracle7 driver returns an error message based on the text associated with the SQLSTATE.

Error messages have the following format:

[*vendor*][*ODBC-component*][*data-source*]*error-message*

The prefixes in brackets ([ ]) identify the source of the error. The following table shows the values of these prefixes returned by the Oracle7 driver. When the error occurs in the data source, the [*vendor*] and [*ODBC-component*] prefixes identify the vendor and name of the ODBC component that received the error from the data source.

Error source	Prefix	Value
Driver Manager	[*vendor*] [*ODBC-component*] [*data-source*]	[Microsoft] [ODBC DLL] N/A
Oracle7 Driver	[*vendor*] [*ODBC-component*] [*data-source*]	[ORACLE] [ODBC Oracle Driver] N/A
Oracle7 RDBMS	[*vendor*] [*ODBC-component*] [*data-source*]	[ORACLE] [ODBC Oracle Driver] [Oracle OCI]

Initialization and Configuration Files (Advanced)

The Oracle7 driver uses only the ODBC.INI file, which is in the Windows directory (by default, this is C:\WINDOWS). You can use the ODBC Setup program to edit the ODBC.INI file. Please refer to the ODBC Setup help files for more information. Each section of the file that describes an Oracle data source can include the following keywords.

Keyword	Description
Driver	The full path of the driver dynamic-link library (DLL).
Description	The user-specified description of the data source.
Server	An Oracle connect string of the form: *net_prefix*:*hostname*[:*SID*] where *net_prefix* specifies the SQL*Net driver to use, *hostname* specifies the name or alias of the server on which Oracle RDBMS resides, and *SID* is the

	optional system ID of the database you want to use on the server. Some common *net_prefixes* are: B: NetBIOS D: DECnet P: Named pipes T: TCP/IP X: SPX/IPX For more information, see the SQL*Net documentation.
UserID	The name of the user on Oracle RDBMS.
TranslationDLL	The file name and path of a DLL that translates all data flowing between the application and the data source. (For information on how to specify a translation DLL, see the steps in the section below.)
TranslationOption	A 32-bit value that the Oracle7 driver passes to the translation DLL. The meaning of the value depends on the translation DLL.

For example, an entry in the ODBC.INI file for a data source named My Oracle RDBMS that resides on the ORACLSRV server and uses the NetBIOS SQL*Net driver might be:

```
[My Oracle7 RDBMS]
Driver=C:\WINDOWS\SYSTEM\SQORA7.DLL
Description=The Oracle database on my OS/2 workstation.
Server=B:ORACLSRV
UserID=JohnS
```

To specify a translator for an Oracle data source

1. Open the ODBC.INI file in a text editor such as Notepad.
2. Find the section for the data source. The data source name will be in brackets. For example:

   ```
   [My Oracle7 Data Source]
   ```
3. Add the **TranslationDLL** keyword and the full path of the translator on the line directly after the section name. For example:

   ```
   [My Oracle7 Data Source]
   TranslationDLL=C:\WINDOWS\SYSTEM\MYXLT.DLL
   ```
4. Add the **TranslationOption** keyword and the translation option on the line directly after the **TranslationDLL** keyword. For example:

   ```
   [My Oracle7 Data Source]
   TranslationDLL=C:\WINDOWS\SYSTEM\MYXLT.DLL
   TranslationOption=10070437
   ```
5. Save the ODBC.INI file, and exit the text editor.

SQLGetInfo Return Values (Programming)

The following table lists the C language #defines for the *fInfoType* argument and the corresponding values returned by **SQLGetInfo**. An application can retrieve this information by passing the listed C language #defines to **SQLGetInfo** in the *fInfoType* argument.

***fInfoType* value (#define)**	**Returned value**
SQL_ACTIVE_CONNECTIONS	Returns 15 (a reasonable and practical limit)
SQL_ACTIVE_STATEMENTS	Always returns 0 as there is no specified limit
SQL_DATA_SOURCE_NAME	Returns "Oracle7".
SQL_DRIVER_HDBC	Handled by the driver manager.
SQL_DRIVER_HENV	Handled by the driver manager.
SQL_DRIVER_HSTMT	Handled by the driver manager.
SQL_DRIVER_NAME	Returns a long pointer to "SQORA7.DLL"
SQL_DRIVER_VER	Returns a long pointer to "current version string"
SQL_FETCH_DIRECTION	Returns SQL_FD_FETCH_NEXT
SQL_ODBC_API_CONFORMANCE	Returns 1 to indicate level 1 conformance.
SQL_ODBC_VER	Handled by the driver manager.
SQL_ROW_UPDATES	Returns a long pointer to the character string "N"
SQL_ODBC_SAG_CLI_CONFORMANCE	Returns 1 to indicate that driver is SAG compliant
SQL_SERVER_NAME	Returns server name supplied at login.
SQL_SEARCH_PATTERN_ESCAPE	Returns a long pointer to the character string "\"
SQL_ODBC_SQL_CONFORMANCE	Returns 1 for core SQL.
SQL_DATABASE_NAME	Returns a long pointer to an empty string .
SQL_DBMS_NAME	Returns a long pointer to the string "Oracle7"
SQL_DBMS_VER	Returns a long pointer to the character string 07.00.0000.
SQL_ACCESSIBLE_TABLES	Returns a long pointer to the character string "Y"
SQL_ACCESSIBLE_PROCEDURES	Returns a long pointer to the character string "Y".
SQL_PROCEDURES	Returns a long pointer to the character string "Y".
SQL_CONCAT_NULL_BEHAVIOR	Returns 1 to indicate that result is concatenation of non-NULL valued columns.
SQL_CURSOR_COMMIT_BEHAVIOR	Returns 2 to indicate that cursors on other statements are unaffected by a commit.
SQL_CURSOR_ROLLBACK_BEHAVIOR	Returns 2 to indicate that cursors on other statements are unaffected by a rollback.
SQL_DATA_SOURCE_READ_ONLY	Returns a long pointer to the character string "N".
SQL_DEFAULT_TXN_ISOLATION	Returns SQL_TXN_REPEATABLE_READ.
SQL_EXPRESSIONS_IN_ORDERBY	Returns a long pointer to the character string "Y"
SQL_IDENTIFIER_CASE	Returns 4 indicating that indentifiers are not case sensitive.
SQL_IDENTIFIER_QUOTE_CHAR	Returns a long pointer to the character ".
SQL_MAX_COLUMN_NAME_LEN	Returns 30.
SQL_MAX_CURSOR_NAME_LEN	Returns 18.
SQL_MAX_OWNER_NAME_LEN	Returns 30.

SQL_MAX_PROCEDURE_NAME_LEN	Returns 30.
SQL_MAX_QUALIFIER_NAME_LEN	Returns 0 to indicate that qualifiers are not supported.
SQL_MAX_TABLE_NAME_LEN	Returns 30.
SQL_MULT_RESULT_SETS	Returns a long pointer to the character string "N".
SQL_MULTIPLE_ACTIVE_TXN	Returns a long pointer to the character string "Y".
SQL_OUTER_JOINS	Returns a long pointer to the character string "Y".
SQL_OWNER_TERM	Returns a long pointer to "Owner".
SQL_PROCEDURE_TERM	Returns a long pointer to "Procedure"
SQL_QUALIFIER_NAME_SEPARATOR	Returns a long pointer to the character string "."
SQL_QUALIFIER_TERM	Returns a long pointer to an empty string .
SQL_SCROLL_CONCURRENCY	Returns SQL_SCCO_READ_ONLY.
SQL_SCROLL_OPTIONS	Returns SQL_SO_FORWARD_ONLY
SQL_TABLE_TERM	Returns a long pointer to "Table".
SQL_TXN_CAPABLE	Returns 1 to indicate that transactions are supported.
SQL_USER_NAME	Returns long pointer to Login ID
SQL_CONVERT_FUNCTIONS	Returns 0.
SQL_NUMERIC_FUNCTIONS	Returns SQL_FN_NUM_ABS | SQL_FN_NUM_CEILING | SQL_FN_NUM_FLOOR | SQL_FN_NUM_MOD | SQL_FN_NUM_SIGN | SQL_FN_NUM_SQRT.
SQL_STRING_FUNCTIONS	Returns SQL_FN_STR_CONCAT | SQL_FN_STR_INSERT | SQL_FN_STR_LEFT | SQL_FN_STR_LTRIM | SQL_FN_STR_LENGTH | SQL_FN_STR_LOCATE | SQL_FN_STR_LCASE | SQL_FN_STR_REPEAT | SQL_FN_STR_REPLACE | SQL_FN_STR_RIGHT | SQL_FN_STR_RTRIM | SQL_FN_STR_SUBSTRING | SQL_FN_STR_UCASE | SQL_FN_STR_ASCII | SQL_FN_STR_CHAR;
SQL_SYSTEM_FUNCTIONS	Returns SQL_FN_SYS_IFNULL | SQL_FN_SYS_USERNAME.
SQL_TIMEDATE_FUNCTIONS	Returns SQL_FN_TD_CURTIME | SQL_FN_TD_HOUR | SQL_FN_TD_MINUTE | SQL_FN_TD_SECOND | SQL_FN_TD_NOW | SQL_FN_TD_QUARTER | SQL_FN_TD_WEEK | SQL_FN_TD_DAYOFWEEK | SQL_FN_TD_CURDATE | SQL_FN_TD_MONTH | SQL_FN_TD_DAYOFMONTH | SQL_FN_TD_YEAR | SQL_FN_TD_DAYOFYEAR | SQL_FN_TD_QUARTER.
SQL_CONVERT_BIGINT	Returns 0.
SQL_CONVERT_BINARY	Returns 0.
SQL_CONVERT_BIT	Returns 0
SQL_CONVERT_CHAR	Returns 0
SQL_CONVERT_DATE	Returns 0
SQL_CONVERT_DECIMAL	Returns 0.
SQL_CONVERT_DOUBLE	Returns 0
SQL_CONVERT_FLOAT	Returns 0.
SQL_CONVERT_INTEGER	Returns 0

SQL_CONVERT_LONGVARCHAR	Returns 0
SQL_CONVERT_NUMERIC	Returns 0.
SQL_CONVERT_REAL	Returns 0.
SQL_CONVERT_SMALLINT	Returns 0.
SQL_CONVERT_TIME	Returns 0.
SQL_CONVERT_TIMESTAMP	Returns 0.
SQL_CONVERT_TINYINT	Returns 0.
SQL_CONVERT_VARBINARY	Returns 0.
SQL_CONVERT_VARCHAR	Returns 0.
SQL_CONVERT_LONGVARBINARY	Returns 0.
SQL_TXN_ISOLATION_OPTION	Returns SQL_TXN_REPEATABLE_READ.
SQL_ODBC_SQL_OPT_IEF	Returns a long pointer to the character string "N".

ODBC API Functions (Programming)

See Also

The Oracle7 driver supports all core and Level 1 functions. It supports the following Level 2 functions:

SQLMoreResults
SQLNativeSql
SQLParamOptions
SQLProcedures
SQLProcedureColumns

In addition, the Oracle7 driver supports translation DLLs.

The following Help topics describe the ODBC API functions implemented by the Oracle7 driver.

For Programmers

Extensions to ODBC API Functions (Programming)

Implementation of ODBC API Functions (Programming)

Limitations to ODBC API Functions (Programming)

See Also

For Advanced Users

Error Messages (Advanced)

Extensions to ODBC API Functions (Programming)

The only function in the Oracle7 driver that exceeds the specifications in the *Microsoft ODBC SDK Programmer's Reference* is **SQLGetData**.

SQLGetData can retrieve data from any column, regardless of whether there are bound columns to the right of that column.

Implementation of ODBC API Functions (Programming)

The following table describes how the Oracle7 driver implements specific functions.

Function	Description
SQLConnect	If the Oracle workstation configuration file (ORACLE.INI) contains the LOCAL keyword, **SQLConnect** requires only a user ID and password.
SQLDriverConnect	**SQLDriverConnect** uses the DSN, DBQ, UID, and PWD keywords.
SQLMoreResults	Because Oracle doesn't support multiple result sets, **SQLMoreResults** always returns SQL_NO_DATA_FOUND and adjusts the state of the *hstmt* to allocated (if the SQL statement was executed with **SQLExecDirect**) or prepared (if the SQL statement was executed with **SQLExecute**).
SQLSpecialColumns	If **SQLSpecialColumns** is called with the SQL_BEST_ROWID option, it always returns the ROWID column.

SQLDriverConnect Implementation (Programming)

If the Oracle workstation configuration file (ORACLE.INI) contains the LOCAL parameter, **SQLDriverConnect** requires only a user ID and password.

The **SQLDriverConnect** connection string uses the following keywords:

Keyword	Description
DSN	The name of the data source as listed in the ODBC.INI file.
DBQ	An Oracle connect string of the form: *net_prefix*:*hostname*[:*SID*] where *net_prefix* specifies the SQL*Net driver to use, *hostname* specifies the name or alias of the server on which the Oracle RDBMS resides, and *SID* is the optional system ID of the database you want to use on the server. Some common *net_prefixes* are: B: NetBIOS D: DECnet P: Named pipes T: TCP/IP X: SPX/IPX

	For more information, see the SQL*Net documentation.
UID	The user login ID.
PWD	The user-specified password.

Limitations to ODBC API Functions (Programming)

The following functions in the Oracle7 driver do not meet the specifications in the *Microsoft ODBC SDK Programmer's Reference*.

Function	Description
SQLFetch	Only FETCH_NEXT is available.
SQLMoreResults	Because Oracle does not support multiple result sets, more results are never available. It will always returns SQL_NO_DATA_FOUND.
SQLCancel	Because the Oracle7 driver uses Oracle Call Interface (OCI) and OCI does not support asynchronous processing, **SQLCancel** is equivalent to **SQLFreeStmt(***hstmt***,SQL_CLOSE)**.
SQLGetConnectOption and **SQLSetConnectOption**	These functions support only the SQL_ACCESS_MODE, SQL_AUTOCOMMIT, SQL_TRANSLATE_DLL, and SQL_TRANSLATE_OPTION options.
SQLGetCursorName and **SQLSetCursorName**	These functions get and set the cursor name in accordance with the *Microsoft ODBC SDK Programmer's Reference*. However, you cannot use the cursor name in an SQL statement because the Oracle7 driver doesn't support positioned update and delete statements.
SQLGetStmtOption and **SQLSetStmtOption**	These functions support only the SQL_MAX_LENGTH, SQL_MAX_ROWS, and SQL_NOSCAN options.

Implementation Issues (Programming)

The following implementation-specific issues might affect the use of the Oracle7 driver.

Issue	Description
Oracle Call Interface (OCI)	The Oracle7 driver accesses data in an Oracle data source through OCI.
Setup DLL	The ODBC Administrator calls the function **ConfigDSN** when users configure data sources. For the Oracle7 driver, this function is in a setup DLL

	(SQORAST7.DLL).
Transactions	Some DDL statements can be explicitly committed or rolled back. For more information, see the Oracle documentation.

API

Application programming interface. A set of routines that an application, such as Microsoft Access, uses to request and carry out lower-level services.

Conformance Levels

Some applications can only use drivers that support certain levels of functionality, or conformance levels. For example, an application might require that drivers be able to prompt the user for the password to a data source. This ability is part of the Level 1 conformance level for the application programming interface (API).

ODBC drivers conform to one of three API levels (Core, Level 1, or Level 2) and one of three SQL grammar levels (Minimum, Core, or Extended). Drivers may support some of the functionality in levels above their stated level.

For detailed information about what is in the various conformance levels, programmers should see the *Microsoft ODBC SDK Programmer's Reference*.

DBMS

Database management system. The software used to organize, analyze, search for, update, and retrieve data.

DDL

Data definition language. Any SQL statement that can be used to define data objects and their attributes. Examples include CREATE TABLE, DROP VIEW, and GRANT statements.

DLL

Dynamic-link library. A set of routines that one or more applications can use to perform common tasks. The ODBC drivers are DLLs.

DML

Data manipulation language. Any SQL statement that can be used to manipulate data. Examples include UPDATE, INSERT, and DELETE statements.

ODBC

Open Database Connectivity. A Driver Manager and a set of ODBC drivers that enable applications to access data using SQL as a standard language.

SQL

Structured Query Language. A language used for retrieving, updating, and managing data.

Translation Option

An option that specifies how a translator translates data. For example, a translation option might specify the character sets between which a translator translates character data. It might also provide a key for encryption and decryption.

Translator

A dynamic-link library (DLL) that translates all data passing between an application, such as Microsoft Access, and a data source. The most common use of a translator is to translate character data between different character sets. A translator can also perform tasks such as encryption and decryption or compression and expansion.