Connecting to ORACLE

Borland
SQL Link

Version 1.0

Borland International, Inc. 1800 Green Hills Road
P.O. Box 660001, Scotts Valley, CA 95067-0001

RI   10   9   8   7   6   5   4   3   2   1

Contents

Figures

Tables

Introduction

Borland SQL Link 1.0 enables you to access SQL data stored in an ORACLE database using the same tools with which you now access non-SQL data. This product supports Borland Paradox and Quattro Pro for Windows versions which are licensed for use with SQL Link.

Note   Quattro Pro users access SQL data with SQL Link using the Database Desktop (DBD.EXE), not the Quattro Pro product itself (QPW.EXE). Database Desktop, which provides a look and feel compatible with Paradox for Windows, is included on the Quattro Pro installation disks.

SQL Link enables you to access SQL data in one of two ways:

• If you are a Paradox for Windows or a Quattro Pro for Windows user, you can access SQL data by using Borland's Table View and QBE features. Paradox for Windows users can also use forms, reports, and SQL Link's SQL Editor window.

• If you are familiar with Paradox ObjectPAL, you can access SQL data by writing ObjectPAL applications and embedding SQL statements. This provides full access to all of the features and functions of database servers, including stored procedures, triggers, and data dictionaries.

Where to find information

This manual describes how to install SQL Link, configure the SQL Link ORACLE driver, and connect to an ORACLE database. It also discusses aspects of database access that are unique to ORACLE.

The manual is meant to be used with:

• *The Borland SQL Link User's Guide*

• *READORA.TXT* (the SQL README file for ORACLE)

• Your ORACLE server documentation

• Your Borland desktop product's user documentation

The following table lists SQL Link topics and directs you to the corresponding manuals.

Topic	Where to find information
SQL Link installation prerequisites	*Connecting to* ORACLE, Chapter 1
What happens during SQL Link installation	
Installing SQL Link	
Testing your SQL Link installation	
Configuring your Borland desktop product for use with SQL Link	*Connecting to* ORACLE, Chapter 2
Managing aliases for SQL databases	

Topic	Where to find information
Connecting to the SQL server	*Connecting to* ORACLE, Chapter 3
Troubleshooting connection problems	
Supported data types	
Aspects of using SQL Link that are unique for your SQL server	*Connecting to* ORACLE, Chapter 3
	Your server-specific README file

Terms and conventions

The SQL Link manuals use special typefaces to help you distinguish between keys you press, names of objects, menu commands, and text you type. The following table lists these conventions.

Convention	Applies To	Examples
Bold type	Method names, Database Desktop status messages, and text that you type in	**insertRecord** Paradox displays the message **Index error on key field** Type **a:\install**
Italic type	Names of Database Desktop objects, glossary terms, variables, emphasized words	*Answer table, searchButton, searchVal*
ALL CAPS	DOS files and directories, reserved words, operators, types of queries	PARADOX.EXE, C:\WINDOWS, CREATE
Initial Caps	Applications, fields, menu commands	Sample application, Price field, Form | View Data command
Keycap Font	Keys on your computer's keyboard	*F1, Enter*
`Monospaced font`	Code examples, ObjectPAL code	`myTable.open("sites.db")`

The following table lists conventions used for ObjectPAL syntax.

ObjectPAL Convention	Element	Examples	Meaning
Normal font	Keyword	setPosition	Type exactly as shown.
Italic	Fill-in	*tablevar*	Replace with an expression.
{ | } (braces and bar)	Choice	{ Yes | No }	You *must* choose one of the elements separated by the vertical bar.
[ ] (brackets)	Optional	[, *tableVar2*] [ELSE]	You *can* choose whether or not to include this.
* (asterisk)	Repeat	[, *tableVar2*] *	You can repeat this argument.

1

Installing SQL Link

This chapter describes how to install Borland SQL Link.

Once you install SQL Link at the client workstation, you are ready to configure your Borland desktop application to run with SQL Link, as described in Chapter 2.

Note Be sure you have already installed your Borland SQL Link licensed desktop product (either Paradox for Windows or Quattro Pro for Windows with Database Desktop), as described in your desktop product documentation.

Before you begin

This section describes how to prepare for SQL Link installation.

Figure 1.1 illustrates all the software that must be installed and running before you install SQL Link. The subsections that follow list specific requirements for the servers and client workstations.

For information on network protocol software and network access rights, see your system administrator.

Figure 1.1   SQL Link for ORACLE required components

client workstation

Borland desktop product:
- Paradox for Windows
- Quattro Pro for Windows
(Database Desktop)

Borland SQL Link 1.0
software driver

network protocol
software driver

server

database server:
ORACLE 6.0 RDBMS

network protocol
software driver

network connection
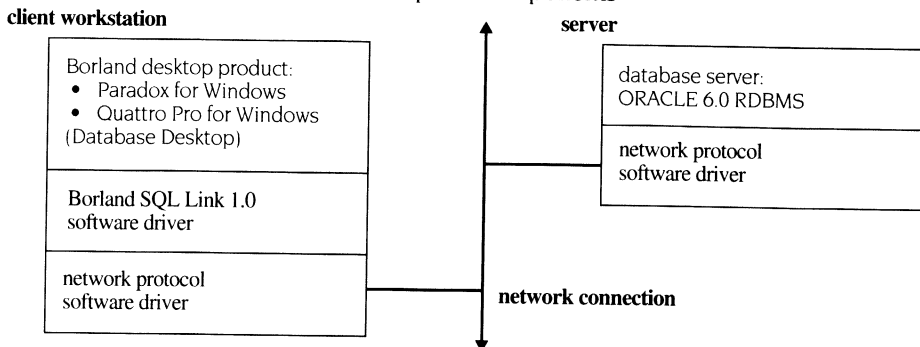
Information you need

To perform the installation, you need the following information:

- The drive (or directory) from which you are installing SQL Link. Usually, this is drive A or drive B.

- The directory where you want to store the SQL Link files. The installation program normally installs the SQL Link files in the same directory with your desktop product ODAPI files. If your desktop product ODAPI files are stored in a directory other than C:\ODAPI, you need to know exactly where that directory is.

ORACLE server requirements

Table 1.1 lists software that should already be installed and running at the ORACLE server.

Table 1.1   Server software requirements

Category	Description
Database server software	ORACLE version 6.0
Network protocol software	ORACLE SQL*NET

Client workstation requirements

Table 1.2 lists software that should already be installed and running at the client workstation. It also lists related files and parameters.

Table 1.2   Client workstation requirements

Category	Description
Borland desktop product(s)	Supported Borland desktop product, installed as required by the product documentation
Hardware and operating system requirements	1.5 MB of free disk space 4 MB RAM (6 MB recommended) Hardware and operating system that meets the requirements of your Borland desktop product
Access rights (for desktop products installed on the network server *only*)	If your Borland desktop product is installed on the shared disk of a network file server, make sure your network user account has Read and Write access rights to the desktop product ODAPI installation directory. This directory is modified during SQL Link installation.
Network protocol software	Network protocol software compatible with both the server network protocol and the client workstation client database communication driver. SQL*Net driver (.EXE or .DLL)

Database access requirements

To access the SQL database, you need a valid user identification and password on the ORACLE server. You also need at least Read access privileges for the SQL database.

Your database administrator can help you obtain these privileges.

What happens during installation?

During installation, the SQL Link installation program:

- Installs an additional ODAPI driver that enables your Borland desktop product to access ORACLE databases (SQLD_ORA.DLL and supporting files).
- Adds a new option (ORACLE) to the Alias Manager dialog box that reflects the presence of the new SQL Link driver.
- Adds new options to the ODAPI Configuration Utility.
- Installs a text file (READORA.TXT) containing information too recent to be included in the printed documentation.

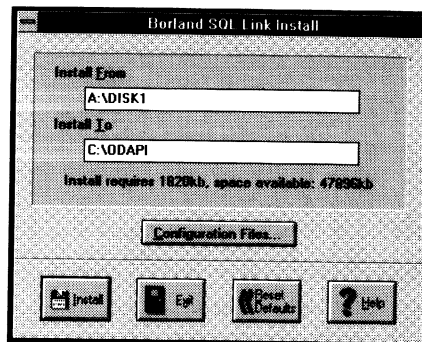# Installing the software

To run the SQL Link INSTALL program:

1 Insert SQL Link Disk 1 in the client workstation external disk drive. (This is usually drive A or drive B.)

2 If Windows is not already running, but is in the workstation DOS PATH, enter **a:\install** at the workstation DOS prompt. The SQL Link Install procedure loads Windows and displays the initial dialog box.

If Windows is already running:

1 Choose File | Run from the Program Manager menu bar. The Run dialog box appears.

2 Enter **a:\install** in the Command Line text box. The Borland SQL Link Install dialog box appears.

Note   Drive A is shown in this example. If your SQL Link software is in Drive B, enter **b:\install**.

Figure 1.2   Borland SQL Link Install dialog box

3 Edit the parameters in the Install dialog box as needed:

Parameter	When to edit
Install To	If you installed ODAPI files in a directory other than C:\ODAPI, enter the name of that directory in this text box.
Configuration Files	If you installed ODAPI.CFG in a directory other than C:\ODAPI, choose the Configuration Files button. The Configuration Files dialog box appears.
	Enter the name of the directory for the new ODAPI.CFG, then choose OK. The program saves a backup copy of the existing ODAPI.CFG as ODAPICFG.BAK.

4 Choose Install to begin the installation.

When the installation is complete, the file READORA.TXT appears. Read this file to find out about late-breaking information.

Configuring your desktop product

When you install your Borland desktop product, you also install the ODAPI Configuration Utility (ODAPICFG.EXE). The Configuration Utility modifies a configuration file (ODAPI.CFG) that your desktop product reads at startup to determine various operating parameters. Both files are usually located in the directory you specify for ODAPI files during desktop product installation (normally C:\ODAPI).

This chapter describes how to configure your Borland desktop product for use with an SQL Link ORACLE driver.

Once you configure your desktop product you are ready to connect to the network and access ORACLE, as described in Chapter 3.

Note Be sure you have already installed the SQL Link software as described in Chapter 1.

For general instructions on how to use the Configuration Utility, see your desktop product documentation.

## Specifying default ORACLE driver settings

ODAPI
Configuration
Utility

The first time you set up an ORACLE alias the configuration program uses the current driver settings. You must set the default settings that match your installation before you create any aliases for ORACLE databases.

To specify default ORACLE driver settings:

1 Open your desktop product program group in the Windows Program Manager.

2 To start the ODAPI Configuration Utility, select the Configuration Utility icon. The ODAPI Configuration window appears.
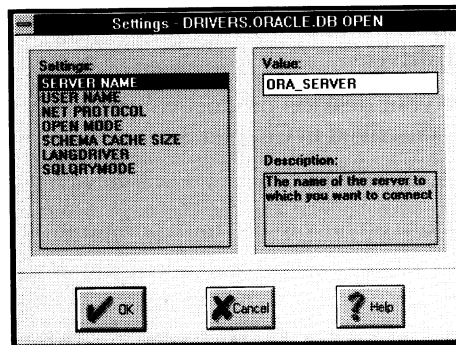
Figure 2.1   ODAPI Configuration window
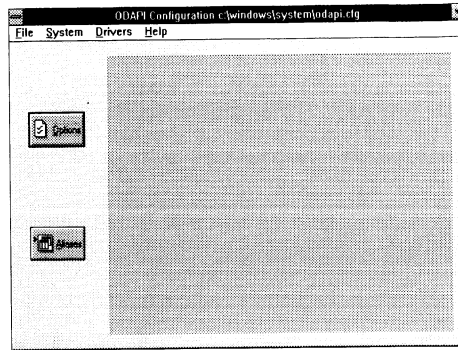
3 Select the Drivers I Oracle menu item. Two categories appear: "Init" and "Db Open".

Note   Although it is possible to modify settings in the Init category, it is not recommended.

4 Highlight the Db Open category. The Settings dialog box appears.

Figure 2.2   ODAPICFG Settings dialog box

5 Edit the Settings dialog box to reflect the category you selected. Table 2.1 describes the meaning of each setting. (For a discussion of alternate ORACLE setting configurations, see "Alternate ORACLE configurations," immediately following this section.

Table 2.1   ORACLE driver settings

Setting	Meaning
SERVER NAME	Default ORACLE name of the database server. It is not always necessary to specify a SERVER NAME. For further information, see "Alternate ORACLE option configurations," later in this section.
USER NAME	Default ORACLE server user name.

Table 2.1 ORACLE driver settings (continued)

Setting	Meaning
NET PROTOCOL	Network transport used to communicate with the database server. See table 2.3.
	It is not always necessary to specify a NET PROTOCOL. For further information, see "Alternate ORACLE option configurations," later in this section.
OPEN MODE	Default mode in which SQL Link opens the SQL database. Possible values are: READ/WRITE and READ ONLY.
SCHEMA CACHE SIZE	Default number of tables whose schema information will be cached. Possible values are 0 - 32. The default value is 8.
LANGDRIVER	Default language driver. For possible values see the short driver names in Table 2.3, at the end of this chapter.
SQLQRYMODE	Specifies the method for handling queries. For possible modes and their meanings, see Table 2.2.

Table 2.2 SQLQRYMODE settings

Setting	Mode	Meaning
NULL (blank setting)	Server-local	(Default mode) In server-local query mode, the query goes first to the server. If the server is unable to perform the query, the query is performed locally.
SERVER	Server-only	In server-only query mode, the query is sent to the server. If the server is unable to perform the query, no local processing is performed.
LOCAL	Local-only	In local-only query mode, the query is always performed locally.

Table 2.3 ORACLE NET PROTOCOL specification Parameters

Value	Description
3270	IBM 3270 protocol
APPC	IBM APPC LU 6.2 protocol
ASYNC	Asynchronous (dial-up) access protocol
DECNET	Digital Equipment Corporation DECnet protocol
NAMED PIPES	Named Pipes protocol, as used by OS/2
NETBIOS	NetBios protocol, as used by LAN Manager and other PC LANs
SPX/IPX	SPX/IPX protocol, as used by Novell NetWare
TCP/IP	Transport Control Protocol/Internet Protocol, as used by Unix and VAX workstations
VINES	Banyan VINES protocol

6 When you finish, save your changes and exit the Configuration Utility.

Your changes take effect the next time you start your desktop product.

Alternate ORACLE configurations

It is not always necessary to provide a SERVER NAME or NET PROTOCOL in the
ORACLE driver Settings dialog box.

Omitting NET PROTOCOL

You can omit NET PROTOCOL in either of the following cases:

- The default ODAPI.CFG SERVER NAME parameter specifies the full ORACLE
  server connect string. The full connect string uses the format:

  ```
  SERVER NAME:@x:OracleServer
  ```

  x specifies that the network protocol is SPX/IPX, and OracleServer is the name of the
  server.

  If you choose this method, ORACLE ignores the NET PROTOCOL parameter at
  connection time.

- The ORACLE configuration file (config.ora) includes a valid SQLNET DBNAME
  parameter. The SQL*NET database alias uses the format:

  ```
  SQLNET DBNAME dbname=x:OracleServer
  ```

  dbname is the same name used in the ODAPI.CFG SERVER NAME parameter.

Omitting SERVER NAME and NET PROTOCOL

You can omit both SERVER NAME and NET PROTOCOL in either of the following
cases:

- The ORACLE configuration file includes a valid LOCAL parameter. The LOCAL
  parameter requires the format:

  ```
  LOCAL=x:OracleServer
  ```

  Once a valid LOCAL parameter exists, ORACLE uses the LOCAL value at
  connection time. The connection succeeds as long as your ODAPI.CFG file contains
  no entries for SERVER NAME and NET PROTOCOL.

- The ORACLE configuration file includes a valid REMOTE parameter. The REMOTE
  parameter requires the format:

  ```
  REMOTE=x:OracleServer
  ```

  Once a valid REMOTE parameter exists, ORACLE uses all or part of the REMOTE
  parameter at connection time. The connection succeeds under any of the following
  conditions:

  - Both NET PROTOCOL and SERVER NAME are blank.
  - NET PROTOCOL is valid and SERVER NAME is blank.
  - NET PROTOCOL is blank and SERVER NAME is valid.

Managing aliases for ORACLE databases

An *alias* is a name and a set of parameters that describe a network resource. Borland desktop products use aliases to connect with shared databases. Before you can access a database, you must first create its alias.

Setting up a standard alias consists of assigning a name to, and specifying the path name for, a directory containing Paradox or dBASE files. Setting up an alias for an SQL database consists of specifying such settings as:

• User name

• Server name

• Open mode

• Default SQL query mode

• Schema cache size

• Language driver

Once the SQL database alias is established, use it the same way you use a standard alias. (For more information on using aliases, see your desktop product documentation.)

The following subsections describe how to use the Configuration Utility to add, modify, and delete aliases.

Adding a new ORACLE alias

To add a new alias:

1 Open your desktop product program group in the Windows Program Manager.

2 To start the ODAPI Configuration Utility, select the Configuration Utility icon. The ODAPI Configuration window appears.

3 Select the Aliases button. The Alias Manager dialog box appears.

Figure 2.3   Alias Manager dialog box.
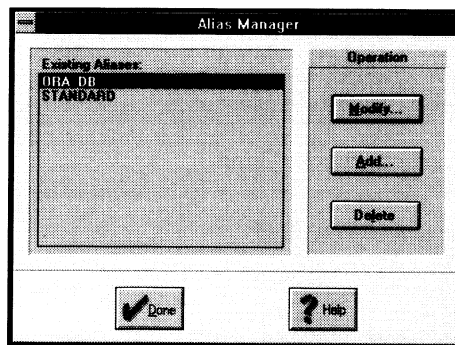
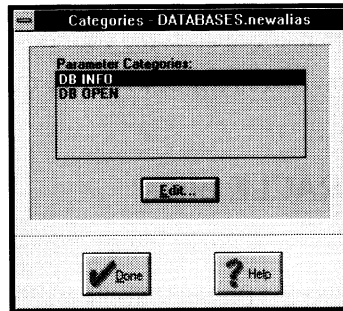4 Select Add. The Add New Alias dialog box appears.

Figure 2.4   Add New Alias dialog box
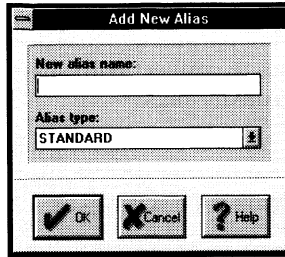
5 Enter the new alias name and select the ORACLE alias type.

6 Choose OK to save the new alias name. The Configuration Manager displays the Categories dialog box.

Figure 2.5   Categories dialog box

7 Highlight the DB OPEN category, then choose Edit. The Settings dialog box appears (see Figure 2.2, earlier in this chapter).

The Settings dialog box is the same one used to set up the default ORACLE driver configuration.

8 Use the Settings dialog box to edit the settings for the category you selected. If you leave any categories blank, the default for driver type is used instead. For a description of each setting you can change, see Tables 2.1 and 2.2, earlier in this chapter.
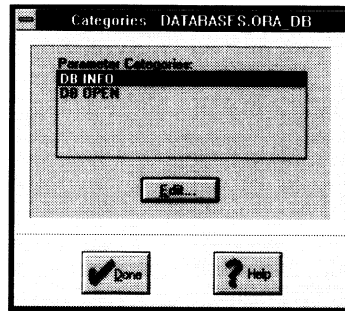
9 When you are finished, choose Done.

Save your changes and exit the Configuration Utility. Your changes take effect the next time you start your desktop product.

Modifying an existing ORACLE alias

To modify an existing alias:

1 Open your desktop product program group in the Windows Program Manager.

2 To start the ODAPI Configuration Utility, select the Configuration Utility icon. The ODAPI Configuration window appears.

3 Choose the Aliases button. This opens the Alias Manager dialog box.

4 Highlight the existing alias for which you want to modify the settings.

5 Choose Modify. The Categories dialog box appears.

Figure 2.6   Modifying an existing alias

6 Highlight the DB OPEN category and select Edit. The Settings dialog box appears (see Figure 2.2, earlier in this chapter).

The Settings dialog box is the same one used to set up the default ORACLE driver configuration.

7 Use the Settings dialog box to edit the settings for the category you selected. If you leave any categories blank, the default for driver type is used instead. For a description of each setting you can change, see Tables 2.1 and 2.2, earlier in this chapter.

8 When you are finished, choose Done.

Save your changes and exit the Configuration Utility. Your changes take effect the next time you start your desktop product.

Deleting an ORACLE alias

To delete an alias:

1 Open your desktop product program group in the Windows Program Manager.

2 To start the ODAPI Configuration Utility, select the Configuration Utility icon. The ODAPI Configuration window appears.

3 Choose the Aliases button to open the Alias Manager dialog box.

4 Highlight the existing alias you want to delete.

5 Choose Delete.

6 When you are finished, choose Done.

Save your changes and exit the Configuration Utility. Your changes take effect the next time you start your desktop product.

Borland language drivers

When a specific language driver is associated with a server alias, your desktop product uses this to manipulate all data that originates from the server. This includes all tables you view in Table View and all Answer tables that result from a query.

The following table lists language drivers available for use with ORACLE, their corresponding character sets, and Borland collation sequences. Choose the language driver that matches your ORACLE server character set and collation sequence. If none of the language drivers below match, choose a language driver that matches the desired character set.

Note If you need to use a language driver that does not properly mimic the ORACLE server collation sequence, be sure to set SQLQRYMODE to SERVER in your ORACLE database alias. For further information about language drivers, see the *Borland SQL Link User's Guide*.

Table 2.4   Language driver names

Long driver name	Short driver name	Character set	Collation sequence
Paradox 'ascii'	ascii	DOS CODE PAGE 437	Binary
Paradox 'intl'	intl	DOS CODE PAGE 437	Paradox 'intl'
Paradox 'intl' 850	intl850	DOS CODE PAGE 850	Paradox 'intl' 850
Paradox 'nordan'	nordan	DOS CODE PAGE 865	Paradox 'nordan'
Paradox 'nordan40'	nordan40	DOS CODE PAGE 865	Paradox 'nordan40'
Paradox 'swedfin'	swedfin	DOS CODE PAGE 437	Paradox 'swedfin'
Paradox ANSI INTL	ANSIINTL	ISO8859.1 (ANSI)	Paradox 'intl'
Paradox ESP 437	SPANISH	DOS CODE PAGE 437	Paradox ESP 437
Paradox ISL 861	iceland	DOS CODE PAGE 861	Paradox ISL 861
Pdox ANSI INTL850	ANSII850	ISO8859.1 (ANSI)	Paradox 'intl' 850
Pdox ANSI NORDAN4	ANSINOR4	ISO8859.1 (ANSI)	Paradox 'nordan40'
Pdox ANSI SWEDFIN	ANSISWFN	ISO8859.1 (ANSI)	Paradox 'swedfin'
Pdox ESP ANSI	ANSISPAN	ISO8859.1 (ANSI)	Paradox ESP437
SQL Link ROMAN8	BLROM800	ROMAN8	Binary

3

Accessing ORACLE

This chapter describes how to connect to an ORACLE database and troubleshoot common connection problems. It also discusses various topics about using Borland SQL Link that are unique to ORACLE.

Once you know how to access an ORACLE database through SQL Link, you are ready to start using SQL Link to display and manipulate ORACLE data described in the *SQL Link User's Guide*.

**Note** Be sure you have already configured SQL Link as described in Chapter 2.

Connecting to ORACLE

The following sections describe how to connect automatically or manually to an ORACLE database. For a discussion of alternate ORACLE connection configuration, see "Alternate ORACLE configurations," in Chapter 2.

Connecting automatically

Whenever you attempt an operation against a target ORACLE database for the first time in a session (like opening a table or running a query), you trigger an automatic connection process. The object of this process is to determine whether you have the right to access the database, and, if so, what kind of access permission you have (read only or read/write).

As the first step in this process, SQL Link displays the Database Information dialog box.

Figure 3.1   Database Information dialog box
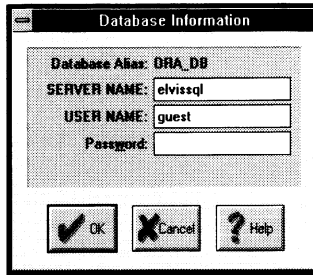
To complete the connection, enter your password.

If the connection is successful, your desktop product continues with the operation you requested. The database to which you connected remains connected for the rest of the current session.
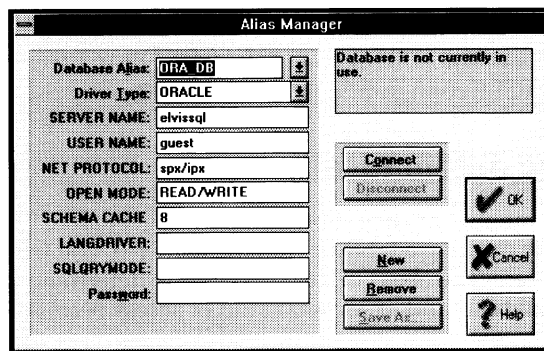
If the connection fails, an error message appears.

Connecting and disconnecting manually

If you ever want to connect to a database without first performing a database action, you can connect manually.

To connect manually:

1 Select the Files | Aliases menu item. The Alias Manager dialog box appears.

Figure 3.2   Alias Manager dialog box

2 Select the alias for the database to which you want to connect. If you need to change any settings, do so now. If the alias represents an ORACLE database, the Alias Manager displays the Connect and Disconnect buttons and some additional text boxes.

   1 To connect manually, enter your password and choose Connect. If the connection is successful, the database to which you connected remains connected for the remainder of the current session. If the connection fails, an error message appears.

2 To disconnect manually, choose Disconnect.

Troubleshooting common connection problems

If you have problems establishing an ORACLE connection with SQL Link, try to isolate the problem the following way:

1 Use your ORACLE tools to verify the connection at each layer:

* Use SQL*DBA or SQL*Plus to enter a CONNECT command. For example:
  ```
  SQLDBA> connect jlee/trapper @p:MIS_SERVER;
  ```
  You can then enter a SQL command to test the connection. Each command should end with a semicolon. For example:
  ```
  SQLDBA> select * from dictionary;
  ```
* Verify that your AUTOEXEC.BAT file contains the lines:
  ```
  SET CONFIG=C:\ORACLE6\CONFIG.ORA
  PATH=C:\;C:\DOS;C:\ORACLE6;C:\ORACLE6\BIN
  ```
  \ORACLE6 is the name of your ORACLE 6.0 directory.

The easiest way to ensure that all of these parameters are set correctly is to run the ORACLE DOS installation program on your workstation (type **orainst** at the DOS prompt). This sets up the appropriate directories, copies the appropriate drivers and programs, creates the configuration file, and sets up the correct environment variables.

2 Make sure that your server and the correct SQL*Net listener are running.

3 If you have file and print services, verify that the network layer is functioning by trying to share files and print jobs to the spooler.

4 Use hardware diagnostics to make sure your network interface card is working properly.

For more information on ORACLE diagnostic tools, see your ORACLE documentation.

Working with ORACLE servers

This section provides information about ORACLE servers and their implementation of SQL. The topics discussed in this section cover aspects of ORACLE that differ from other SQL database products.

Table 3.1 lists the general items that you might find helpful in working with ORACLE servers:

Table 3.1   General information about ORACLE servers

Item	Description
Product name	ORACLE
Dynamic Link Library (DLL) name	SQLD_ORA.DLL
Case-sensitive for data?	Yes
Case-sensitive for objects (such as tables, columns, and indexes)?	No

Table 3.1   General information about ORACLE servers

Item	Description
Does the server require an explicit **begin Transaction()** for multistatement transaction processing in ObjectPAL?	Yes
Does the server require that you explicitly start a transaction for multistatement transaction processing in SQL pass-through?	No
Implicit row IDs	Yes
Blob handles	No
Maximum size of single blob read (if blob handles are not supported)	64K

ORACLE data type translations

Certain database operations cause SQL Link to convert data from Paradox or dBASE format to ORACLE format. For example, an ObjectPAL application that copies or appends data from a local Paradox table to an ORACLE table causes SQL Link to convert the Paradox data to ORACLE format before performing the copy or append operation.

Other database operations cause a conversion in the opposite direction, from ORACLE format to Paradox or dBASE format. For example, suppose you run a QBE (Query By Example) against one or more remote tables. During the query, SQL Link converts any data originating in a remote database to Paradox or dBASE format (depending on the Answer Table Type specified in the Answer Table Properties dialog box) before placing the data in the local answer table.

Tables 3.2, 3.3, and 3.4 list ORACLE, Paradox, and dBASE data types and show how SQL Link translates between these data types.

Table 3.2   ORACLE to Paradox and dBASE data type translations

FROM: ORACLE	TO: Paradox	TO: dBASE
CHAR(n)	Alphanumeric(n)	Character(n)[1]
NUMBER	Number	Number 20.4
DATE[2]	Date	Date
LONG	Memo	Memo
RAW	Binary	Memo
LONG RAW	Binary	Memo

1. dBASE character data type supports only 254 characters. If you exceed this limit your data will be truncated.

2. For example, from ORACLE, QBE maps ORACLE DATE to Paradox Date. Copy table maps ORACLE DATE to Paradox Char(n).

Table 3.3   Paradox to ORACLE and dBASE data type translations

FROM: Paradox	TO: ORACLE	TO: dBASE
Alphanumeric(n)	CHAR(n)	Character(n)
Number	NUMBER	Number 20.4
Currency	NUMBER	Number 20.4
Date	DATE	Date
Short number	NUMBER	Number 6.0
Memo	LONG	Memo
Formatted memo	LONG	Memo
Binary	LONG RAW	Memo
Graphic	LONG RAW	Memo
OLE	LONG RAW	Memo

Table 3.4   dBASE to ORACLE and Paradox data type translations

FROM: dBASE	TO: ORACLE	TO: Paradox
Character(n)	CHAR(n)	Alphanumeric(n)
Number*	NUMBER	Short number, Number
Float number[1]	NUMBER	Short number, Number
Date	DATE	Date
Boolean	CHAR(1)	Alphanumeric (1)
Memo	LONG	Memo

1.   dBASE data types Number and Float translate to different ORACLE and Paradox data types depending on theWIDTH and DEC specification. dBASE Number and Float values with aWIDTH less than 5 and a DEC equal to 0 translate to ORACLE NUMBER or Paradox Short Number data types.

ORACLE System Tables

ORACLE includes a special set of system tables. System tables describe privileges, indexes, SQL table structures, and other items that define relationships within a database. You can access system tables with pass-through SQL from your desktop product through the SQL Editor (see the "Advanced concepts" chapter of the *SQL Link User's Guide)*.

Table 3.5 lists ORACLE system tables you can access through SQLink.

Table 3.5   Selected ORACLE System Tables

Table Name	Use
ACCESSIBLE_COLUMNS	Lists the columns of all tables and clusters.
ACCESSIBLE_TABLES	Lists the tables and views accessible to the user.
ALL_COL_GRANTS_MADE	Lists the columns that the owner has granted other users access to.
ALL_COL_GRANTS_RECD	Lists the columns the user can access.
ALL_INDEXES	Describes indexes on tables accessible to the user.
ALL_TABLES	Lists all tables accessible to the user.

Table 3.5   Selected ORACLE System Tables

Table Name	Use
ALL_CATALOG	Lists all objects accessible to the user.
ALL_TAB_GRANTS_MADE	Lists the tables that the owner has granted other users access to.
ALL_TAB_GRANTS_RECD	Lists the tables the user can access.
ALL_USERS	Lists information on all users of the database.
CONSTRAINT_DEFS	Lists constraint definitions on accessible tables.

ORACLE field-naming rules

Table 3.6 lists field-naming rules for Paradox, dBase, and ORACLE.

Table 3.6   ORACLE field-naming rules

Naming rule	Paradox	dBASE	ORACLE
Max length (characters)	25	10	30
Valid characters[1]	All	All except punctuation marks, blank spaces, and other special characters	A-Z, 0-9, -, $, #
Must begin with . . .	Any valid character except space	A letter	A letter

1. Paradox field names should not contain square brackets [], curly braces {}, pipes |, quotes " ", parentheses (), or the combination ->, or the symbol # alone.

Note   You cannot use ORACLE reserved words for remote table names, or quoted object names. For a list of reserved words and other naming restrictions, see the ORACLE *Programmer's Reference*.

Index

Borland