Connecting to SYBASE and Microsoft SQL Server

Borland
SQL Link

Version 1.0

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

Borland may have patents and/or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents.

COPYRIGHT © 1993 by Borland International. All rights reserved. All Borland products are trademarks or registered trademarks of Borland International, Inc. All other trademarks are the property of their respective owners.

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 for SQL Server enables you to access remote data stored in an SQL Server database without learning Structured Query Language (SQL). This product supports Borland Paradox and Quattro Pro for Windows versions which are licensed for use with SQL Link.

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

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 SYBASE driver, and connect to an SQL database. It also discusses aspects of database access that are unique to SQL Server.

The manual is meant to be used with:

- *The Borland SQL Link User's Guide*
- *READSS.TXT* (the SQL README file for SYBASE)
- Your SQL Server documentation
- Your Borland desktop product 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 What happens during SQL Link installation Installing SQL Link Testing your connection to SQL Server	*Connecting to* SYBASE *and Microsoft* SQL *Server*, Chapter 1
Configuring your Borland desktop product for use with SQL Link Managing remote aliases for SQL databases	*Connecting to* SYBASE *and Microsoft* SQL *Server*, Chapter 2

Topic	Where to find information
Connecting to the SQL server	*Connecting to* SYBASE *and Microsoft* SQL *Server*, Chapter 3
Troubleshooting connection problems	
Supported data types	
Aspects of using SQL Link that are unique for your SQL server	*Connecting to* SYBASE *and Microsoft* SQL *Server*, Chapter 3 Your server-specific README file (i.e., READSS.TXT for SQL Servers)

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.

Chapter

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.

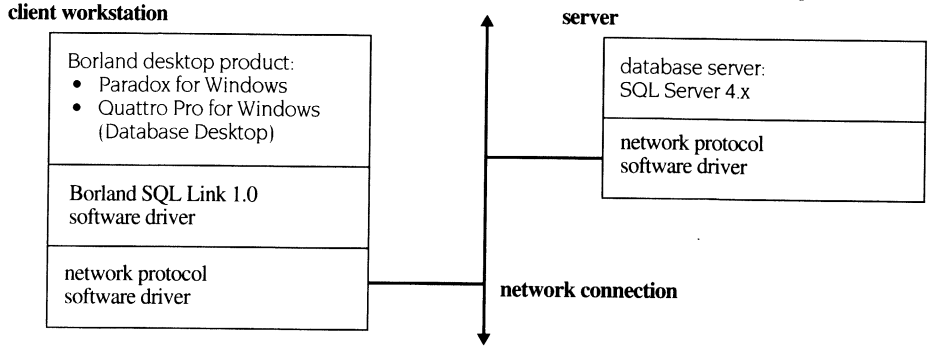
Before you begin

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 SYBASE and Microsoft SQL Server required components



Information you need

To perform the installation, you need to know:

- 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.

SQL Server requirements

Table 1.1 lists software that should already be installed and running at the SQL Server.

Table 1.1 Server software requirements

Category	Description
Database server software:	
SYBASE SQL Server	Version 4.x
Microsoft SQL Server	Version 4.x
Network protocol software:	
SYBASE SQL Server	Supporting the SYBASE Net libraries
Microsoft SQL Server	Supporting SPX/IPX or Named Pipes protocol (Microsoft LAN Manager, 3Com 3+Open, or Novell NetWare Requestor for OS/2)

For more information on network protocol software, see your system administrator.

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 Net library driver

For information on your Borland desktop product hardware and operating system requirements, see your Borland product documentation.

Database access requirements

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

Your database administrator can help you obtain these privileges.

What happens during installation?

In addition to installing the SQL Link files, the SQL Link installation program does the following:

- Installs an additional ODAPI driver that enables your Borland desktop product to access SQL Server databases (SQLD_SS.DLL and supporting files).
- Adds a new option (SYBASE) 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 (READSS.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.)

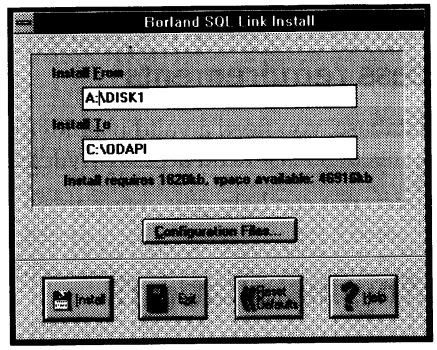
2 If Windows is not already running, but is in the workstation DOS PATH, enter **a:\install** at the workstation DOS prompt. The install procedure loads Windows and displays the initial dialog box. If Windows is already running:

   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 program is complete, the file READSS.TXT appears. Read this file to find out about late-breaking information.

Chapter

2

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 located in the directory you specify for ODAPI files during desktop product installation (usually C:\ODAPI).

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

Once you configure your desktop product you are ready to connect to the network and access a SYBASE or Microsoft SQL Server, 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.

Using the ODAPI configuration utility


ODAPI Configuration Utility

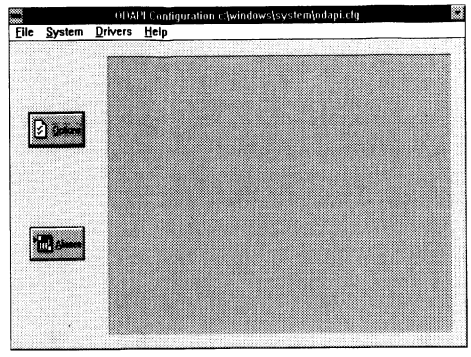
The first time you set up an SYBASE alias the configuration program uses the current driver settings. You must set the default settings that match your installation before you create any aliases for SYBASE databases.

To specify default SQL Server 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.

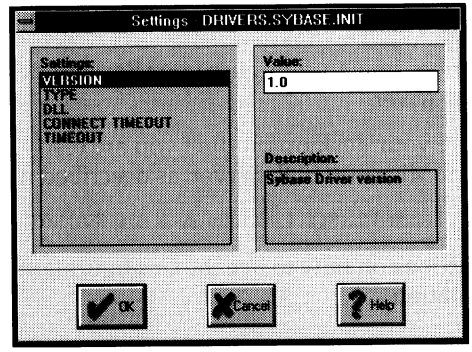
Figure 2.1 ODAPI Configuration window



3 Select the Drivers | SYBASE menu item. Two categories appear: "Init" and "Db Open".

1 To configure the Init category, highlight Init. The Settings dialog box appears.

Figure 2.2 Init Settings dialog box



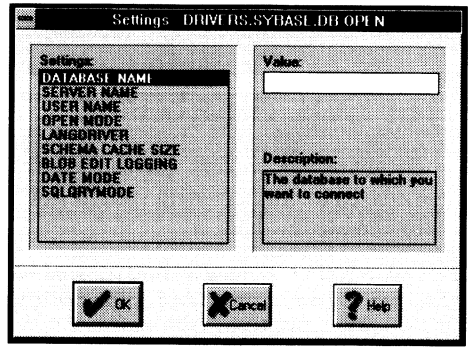
Edit the Settings dialog box to reflect the category you selected. Table 2.1 describes the meaning of each setting you can change.

Table 2.1 SQL Server Init settings

Setting	Meaning
CONNECT TIMEOUT	Specifies the amount of time the workstation waits to attach to the server. The default is 60 seconds.
TIMEOUT	Specifies the maximum amount of time that the workstation waits for results to return from the server. The default is 300 seconds.

2 To configure the Db Open category, highlight Db Open. The Settings dialog box appears.

Figure 2.3 ODAPI.CFG Settings dialog box



Edit the Settings dialog box as needed. Table 2.2 describes the meaning of each setting.

Table 2.2 SQL Server Db open settings

Setting	Meaning
SERVER NAME	Default database server name, as recognized by the SQL server.
USER NAME	Default SQL server user name. Case sensitivity is dependent on server configuration.
OPEN MODE	Default mode in which SQL link opens the SQL database. Possible values are: `READ/WRITE` and `READ ONLY`.
SQLQRYMODE	Specifies the method for handling queries. For possible modes and their meanings, see Table 2.3.
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. Possible values are the short driver names listed in Table 2.4, "Language driver names."
DATABASE NAME	The server database name. Case sensitivity is dependent on server configuration.
BLOB EDIT LOGGING	Controls blob logging on SQL servers. Possible values are TRUE (the default) or FALSE. When set to FALSE this option helps minimize blob space requirements and increase performance. (If you set BLOB EDIT LOGGING to FALSE, you must also set BULKCOPY to ON at the server. For more information, see your SQL Server documentation.)
DATE MODE	Server date format mode that controls the conversion of QBE statements into SQL statements in international environments. Possible values are: 0 for MDY (default), 1 for DMY, 2 for YMD.

Table 2.3 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.

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

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

Managing aliases for SQL 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 SQL Server alias

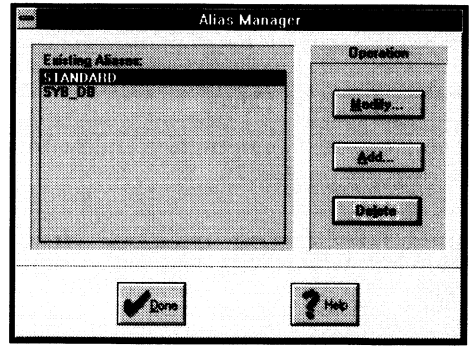
To add a new alias:

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.

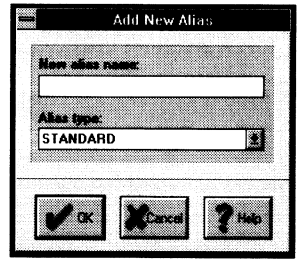
3 Select the Aliases button. The Alias Manager dialog box appears.

Figure 2.4 Alias Manager dialog box.



4 Select Add. The Add New Alias dialog box appears.

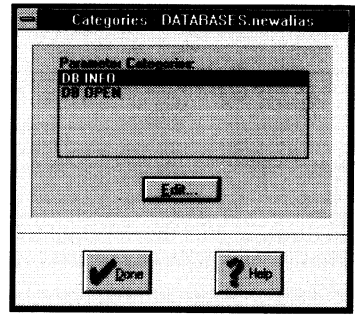
Figure 2.5 Add New Alias dialog box



5 Enter the new alias name and select the SYBASE alias type.

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

Figure 2.6 Categories dialog box



7 Highlight the DB OPEN category, then choose Edit. The Settings dialog box appears (see Figure 2.3, earlier in this chapter).

The Settings dialog box is the same one used to set up the default SYBASE 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 2.2 and 2.3, 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 SQL Server alias

To modify an existing alias:

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

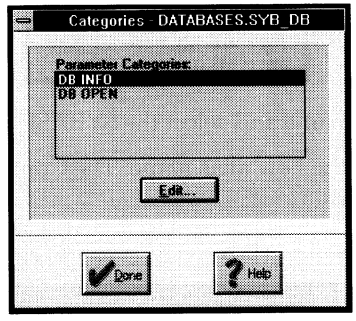
2 To start the ODAPI Configuration Utility, select the Configuration Utility icon. The ODAPI Configuration window appears.

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 2.7 Categories dialog box



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 SYBASE 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 SQL Server 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 SYBASE, their corresponding character sets, and Borland collation sequences. The language driver you use must use a collation sequence that matches the server collation sequence. If none of the language drivers below match the collation sequence at the server, choose a language driver that matches the desired character set.

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

Table 2.4 lists the names of Borland language drivers appropriate for use with SQL.

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'

Table 2.4 Language driver names

Long driver name	Short driver name	Character set	Collation sequence
Pdox ANSI SWEDFIN	ANSISWFN	ISO8859.1 (ANSI)	Paradox 'swedfin'
Pdox ESP ANSI	ANSISPAN	ISO8859.1 (ANSI)	Paradox ESP437
SQL Link ROMAN8	BLROM800	ROMAN8	Binary

Chapter

3

Accessing SQL Server

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

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

Note Be sure you have already configured your Borland desktop product for use with SQL Link as described in Chapter 2.

Connecting to SQL Server

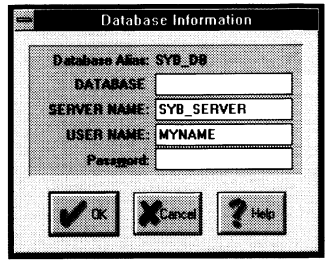
This section describes how to connect automatically or manually to an SQL Server database.

Connecting automatically

Whenever you attempt an operation against a target SQL Server 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 permissions 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.

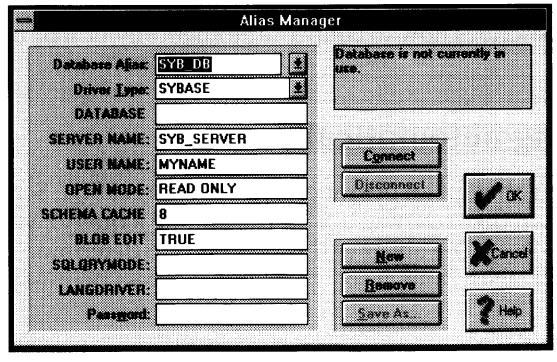
Connecting and disconnecting manually

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 SQL Server 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 SQL Server connection with SQL Link, try to isolate the problem the following way:

1 Use your SYBASE or Microsoft tools to verify the connection at each layer:

   For SYBASE, use DBPING.EXE to check the network connection and ISQL.EXE to verify the server connection.

   For Microsoft SQL Server, use the DOS system administrator facility (SAF.EXE) to verify the connection.

2 Enter a valid query using either tool to verify your connection. For example:

```
select @@version
```

   This query shows the server version number and verifies that you are connected to the server.

3 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 your vendor-supplied diagnostic tools, see your SQL Server documentation.

Working with SQL Servers

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

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

Table 3.1 General information about SQL Servers

Item	Description
Product name	SYBASE
SQL dialect	SYBASE
Dynamic Link Library (DLL) name	SQLD_SS.DLL
Case-sensitive for data?	As installed
Case-sensitive for objects (such as tables, columns, and indexes)?	As installed
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?	Yes

Table 3.1 General information about SQL Servers

Item	Description
Implicit row IDs	No
Blob handles	No
Maximum size of single blob read (if blob handles are not supported)	32K

SQL Server data type translations

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

Other database operations cause a conversion in the opposite direction, from SQL Server 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 SQL Server, Paradox, and dBASE data types and show how SQL Link translates between these data types.

Table 3.2 SQL Server to Paradox and dBASE data type translations

FROM: SQL Server	TO: Paradox	TO: dBASE
char(n)	Alphanumeric(n)	Character(n)
binary(n)	Binary	Memo
bit	Alphanumeric(1)	Boolean
datetime	Date[1]	Date
float	Number	Number 20.4
image	Binary	Memo
int	Number	Number 10.0
money	Currency	Number 20.4
real	Number	Number 20.4
smalldatetime	Date	Date
smallint	Short number	Number 6.0
smallmoney	Currency	Number 20.4
text	Memo	Memo
timestamp	Binary	Memo
tinyint	Short number	Number 6.0
varbinary(n)	Binary	Memo
varchar(n)	Alphanumeric(n)	Character(n)

1. QBE maps SYBASE DATE to Paradox Date. Copy table maps SYBASE DATE to Paradox Char(n).

Table 3.3 Paradox to SQL Server and dBASE data type translations

FROM: Paradox	TO: SQL Server	TO: dBASE
Alphanumeric(n)	varchar(n)	Character(n)
Number	float	Number 20.4
Currency	money	Number 20.4
Date	datetime	Date
Short number	smallint	Number 6.0
Memo	text	Memo
Formatted memo	text	Memo
Binary	image	Memo
Graphic	image	Memo
OLE	image	Memo

Table 3.4 dBASE to SQL Server and Paradox data type translations

FROM: dBASE	TO: SQL Server	TO: Paradox
Character(n)	varchar(n)	Alphanumeric(n)
Number[1]	int, float	Number
Float number[1]	int, float	Number
Date	datetime	Date
Boolean	bit	Alphanumeric (1)
Memo	text	Memo

1. dBASE data types Number and Float number translate to different SQL Server and Paradox data types depending on the WIDTH and DEC specification. dBASE Number and Float values with a WIDTH less than 5 and a DEC equal to 0 translate to an SQL Server SMALLINT or Paradox Short Number data types.

SQL Server system tables

SQL Server 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 *Borland SQL Link User's Guide).*

Table 3.5 lists SQL Server system tables you can access through SQL Link.

Table 3.5 Selected SQL Server System Tables

Table name	Use
syscolumns	Lists each column in a table or view
sysdepends	Lists each procedure, view, or table referenced by a procedure, view, or trigger
sysindexes	Lists each clustered index, nonclustered index, and table with no index
syslogins	Lists each valid user account
sysobjects	Lists each object for the database

Table 3.5 Selected SQL Server System Tables

Table name	Use
sysprotects	Lists user permission information
sysusers	Lists each user and group for the database

SQL Server field-naming rules

Table 3.6 lists field-naming rules for Paradox, dBASE, and SQL Server.

Table 3.6 Field-naming rules

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

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

Note You cannot use SQL Server reserved words for remote table and column names. See the SQL Server *Programmer's Reference* for a list of reserved words.

Index

Borland

Corporate Headquarters: 1800 Green Hills Road, P.O. Box 660001, Scotts Valley, CA 95067-0001, (408) 431-1000. Offices in: Australia, Belgium, Canada, Denmark, France, Germany, Hong Kong, Italy, Japan, Korea, Latin America, Malaysia, Netherlands, New Zealand, Singapore, Spain, Sweden, Taiwan, and United Kingdom • Part # SQS1110WW21770 • BOR 6063