# SQL Links for Windows User's Guide

## Delphi for Windows



Copyright Agreement

Borland SQL Links for Windows enables users to access and manipulate SQL data using features in Borland applications.

## Supported Borland applications

Any Borland application that supports the Borland Database Engine (BDE) can use Borland SQL Links. The cornerstone of all Borland Windows-based database products, the BDE (also known as IDAPI, the Integrated Database Application Programming Interface) supports Paradox for Windows, dBASE for Windows, Delphi 95, and custom applications built using the BDE API.

For the purpose of this manual, we refer to all supported Borland database products and custom applications as BDE applications.

## Supported databases

Drivers in the SQL Links product package support InterBase, Informix, ORACLE, and SYBASE and Microsoft SQL Server databases.

## Where to find information

This book describes how Borland SQL Links for Windows works, how to install one or more SQL Link drivers, and how to set up your supported Borland application to access SQL data. It includes appendices which provides driver-specific information on required workstation software, configuration parameters, troubleshooting, and data translation.

It is meant to be used with:

- your BDE application user documentation, which describes how to use your product to access both local (workstation-based) and SQL server data
- your BDE application programming documentation, which describes how to write custom applications to use with both local and SQL server data

- your SQL server documentation

Depending on how you plan to work with SQL data, you may not need to read through all of the above books. The following table shows what kind of information each type of SQL Links user needs, and where to find it in SQL Links and related documentation.

**Table Intro.1** Where to find information on SQL Links and related topics

| Who you are | What you do | Information you need | Where to find it |
|---|---|---|---|
| BDE application end-user (no knowledge of SQL) | Use Borland desktop or custom applications to work with both local and SQL data (graphical user interface only) | How to install SQL Links | Borland SQL Links for Windows *User's Guide* |
| | | How to create one or more aliases to the SQL database | Borland SQL Links for Windows *User's Guide* |
| | | How to connect to the SQL database through your BDE application | BDE application user documentation |
| | | How to work with local and SQL data through your Borland desktop graphical user interface (GUI) | BDE application user documentation |
| BDE application end-user (knows SQL) | Use supported application to pass SQL statements to local and/or SQL data | How to install SQL Links | Borland SQL Links for Windows *User's Guide* |
| | | How to create one or more aliases to the SQL database | Borland SQL Links for Windows User's Guide |
| | | How to connect to the SQL database through your BDE application | BDE application user documentation |
| | | How to frame SQL statements the server can accept and work with | SQL server documentation |
| | | How to pass SQL statements from your BDE application directly to the SQL server | BDE application user documentation and/or online Help |
| BDE application developer | Adapt existing applications (or write new ones) for use in a mixed workstation/SQL environment | How to install SQL Links | Borland SQL Links for Windows *User's Guide* |
| | | How to create one or more aliases to the SQL database | Borland SQL Links for Windows *User's Guide* |
| | | Differences between workstation databases and SQL databases | Product-specific online Help for SQL and upsizing issues |
| | | Programming tools available to create applications that will work in a mixed workstation/SQL environment | Product-specific online Help for SQL and upsizing issues, BDE application programmer's reference |
| | | Summary information on data type translation, language driver usage, and other general factors that influence the interaction between the Borland application and the server | Driver-specific appendix in the Borland SQL Links *User's Guide* |
| | | How to frame SQL statements the server can accept and work with | SQL server documentation |

Other important sources of information include the Borland SQL Links README file (READLINK.TXT)and the IDAPI Configuration Utility online Help.

## Terms and conventions

The Borland SQL Links manuals use special typefaces to help you distinguish between keys to press, names of objects, menu commands, and text to be entered. The following table lists these conventions:

**Table Intro.2** SQL Link documentation conventions

| Convention | Applies to | Examples |
|---|---|---|
| **Bold type** | Method names, error and information messages, and text the user types in | **insertRecord**<br>Paradox displays the message **Index error on key field**<br>Enter **a:\install** |
| *Italic type* | Glossary terms, variables, emphasized words | *Answer table, searchButton, searchVal* |
| ALL CAPS | DOS files and directories, reserved words, operators, types of SQL queries | CONNECT.EXE, C:\WINDOWS, CREATE |
| Initial Caps | Applications, fields, menu commands, data types | Sample application, Price field, Form \| View Data command, Interval data type |
| *Keycap Font* | Keys on the computer keyboard | *F1*, *Enter* |
| `Monospaced font` | Code examples | `myTable.open("sites.db")` |

Chapter

1

# How Borland SQL Links work

This chapter provides a general overview of the Borland SQL Links and how they work with a Borland Database Engine (BDE) application.

## Introduction

If you work in a setting where you need to share access to files or printers, you are probably familiar with the idea of a *local area network*, or LAN. A LAN enables workstation users to share files, software, and printer resources stored on dedicated machines called *servers*. Workstations connect to network servers through a system of cabling, communications hardware, and software. In large user populations, two or more LANs can connect through *gateways* to form *wide area networks*, or WANs.

In a network environment, your workstation uses the network server in much the same way as it uses its own hard disk. If your workstation needs access to data stored on the server's hard disk it requests that data from the server. The server sends the requested data over the network and back to your workstation where it is processed locally. However, the network server differs from the workstation in that server data can be accessed by more than one user at the same time.

### The database server

A *database server* is a computer that processes high-level database requests. Although other types of network servers let most processing occur on the user's workstation, database servers are active, with most processing occurring on the database server itself. If your workstation needs access to data stored in a database server, you *query* the server directly. The database server processes the query itself and sends only the answer over the network and back to your workstation.

Since the processing is performed at the server and not at the workstation that originated the request, the workstation becomes a *client* of the database server. In a *client/server system*, multiple clients (users) request the services of the *database server*

through the use of *client applications* such as Paradox for Windows, dBASE for Windows, Delphi, or Borland C++.

Thus, the client and the server share the work of maintaining the database: the server dictates the kind of interactions the client can have with the data, but the client chooses how to display and use the results of those interactions.

## SQL

SQL is a descendant of SEQUEL (or Structured English QUEry Language), which was designed at IBM over twenty years ago. SQL was created as a language for constructing *relational database management systems* (RDBMS) on any hardware platform. The first commercial RDBMS using SQL appeared in 1981, and SQL is now the standard language for network queries across different hardware and software platforms.

SQL is actually a *sublanguage* designed to be embedded in an application programming language. It is so flexible that it can be used for both data manipulation and data definition.

SQL database servers handle requests in logical units of work called *transactions*. A transaction is a group of related operations that must all be performed successfully before the RDBMS will finalize any changes to the database. Transaction processing on database servers ensures that your processing requests are appropriate to the current state of your data.

In SQL, all transactions can be explicitly ended with a statement (command) to either accept or discard the changes. Once you are satisfied that no errors occurred during the transaction, you can end that transaction with a COMMIT statement. The database then changes to reflect the operations you have just performed. If an error occurs, you can abandon the changes with a ROLLBACK statement.

**Note** Many outside sources are available for further information about SQL and RDBMS. An example is C.J. Date's *An Introduction to Database Systems* (Addison-Wesley, Reading, Massachusetts, 1983).

# Borland SQL Links for Windows

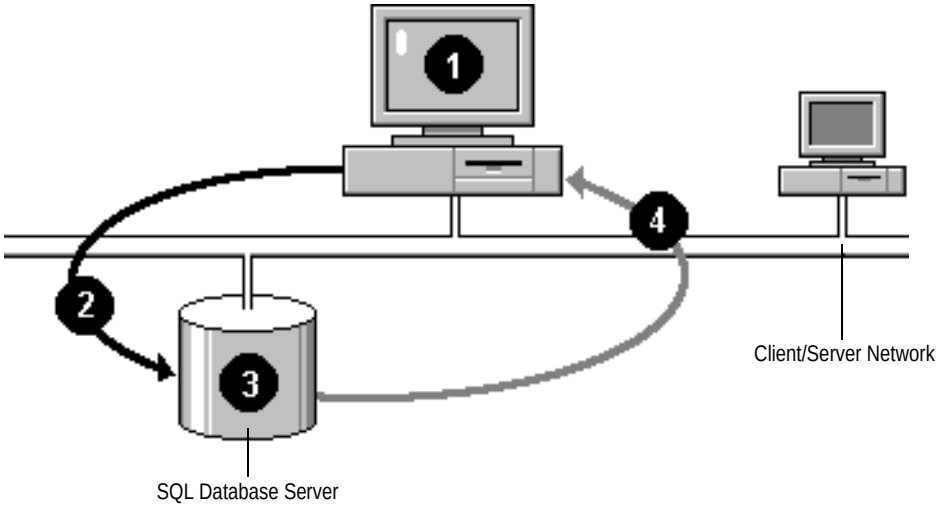
Borland SQL Links for Windows is designed for BDE application users who need to access both *local* (Paradox and dBASE) and SQL databases. You begin with the BDE application installed at your workstation, along with any other client software needed to access the SQL server. Once you install the SQL Link driver and create an SQL driver alias, you can use your BDE application to access SQL data the same way you use it to access a local Paradox or dBASE database:

- through your application's user interface
- through custom applications that may include embedded SQL statements
- by passing SQL statements directly to the server

**Note** BDE applications also support the use of SQL statements against local data. For information on how to use *local SQL* in your BDE application, see your application's user documentation.

With an SQL Link driver installed, you can access SQL data without needing to learn SQL. The driver enables the connection to the SQL server, translates queries into the appropriate SQL dialect, and passes them to the SQL database. When processing is complete, the SQL database returns the answer to the client in a format that the desktop application can display. Figure 1.1 shows how this works.

**Figure 1.1** How SQL Links works




1. BDE client queries an SQL database.
2. SQL Link driver enables connection between the client workstation and the SQL server (translates query into appropriate SQL dialect), and sends the query to the SQL server.
3. SQL server performs error and syntax checking on the SQL statements, processes the query, and displays the answer to the BDE client.
4. SQL Link driver translates the answer from SQL to client syntax. The client then formats and displays the data to the user.

## How the SQL Links work

Using your BDE application with an SQL Link driver actually extends what you can do with your application, lending it some of the advantages inherent in using SQL databases:

- the ability to query the SQL server directly
- support for SQL-style transactions
- improved record caching
- data locking behavior

Accessing an SQL server through an SQL Link driver and your BDE application also lends traditional SQL database users some of the advantages inherent in workstation databases, enabling you to

- move in both directions through a result set or answer table
- order data using an available index

- set a “book mark” on a database location in the current session, and reuse the bookmark in a later session
- work dynamically with the SQL data itself through “live” access to data sources

## Options for querying SQL data

The following sections describe two SQL Link driver options that influence how your application behaves when querying SQL data.

For information on how to use your application to query SQL data, or how to query Paradox or dBASE data using SQL statements, see your BDE application documentation. For information on how to adapt custom BDE applications for use with SQL data, see your product-specific online Help for SQL and upsizing issues.

### Default SQL query mode

Under most circumstances, queries to SQL databases are processed on the SQL server. However, in cases where the SQL server cannot (or should not) process a query, the BDE application processes the query locally.

Types of SQL database queries that are processed locally include

- Heterogenous queries (queries made across more than one type of data)
- Queries that cannot be expressed as a single SQL statement
- Queries that the SQL server does not support

If you want to make sure that all queries originating from your BDE application are processed according to the rules of the SQL server, you can configure the SQL Link driver to block workstation processing of queries. For example, you may want to force remote processing if you are using a non-English language driver that does not properly mimic the SQL server’s conventions for sorting and uppercasing.

To do so, use the IDAPI Configuration Utility to modify the default SQL database alias, setting default SQLQRYMODE to SERVER. (See “Creating and managing SQL aliases,” in Chapter 3.)

### Default pass-through SQL mode

Usually the BDE application enables you to choose whether you want to query the SQL database through the user interface or pass SQL statements directly to the server. Through the IDAPI Configuration Utility, you can also specify whether or not SQL statements passed directly to the server should be automatically committed.

For a discussion of SQLPASSTHRU MODE options, see Chapter 3. For information about using pass-through SQL with your BDE application, see your application’s user documentation.

Chapter 2

# Installing the SQL Link drivers

This chapter describes how to install the Borland SQL Link drivers.

## Client workstation requirements

Table 2.1 lists software that should already be installed and running at the client workstation. It also lists related files and parameters.

**Table 2.1** Client workstation requirements

| Category | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation |
| Hardware and operating system requirements | 1.5 MB of free disk space<br>Hardware and operating system that meets the requirements of your BDE application |
| Access rights (for applications installed on the network server *only*) | If your BDE application 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 application's IDAPI 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 |

For information on the software required at your SQL server and other server-related software requirements, see your driver-specific appendix.

# 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 (SELECT) access privileges for the SQL database.

To obtain these privileges, see your database administrator.

# Installing the software

**Before you begin**

- Be sure your workstation already meets the client workstation requirements for installing Borland SQL Links for Windows.
- Be sure you have already installed your BDE application.

## What happens during installation?

During installation, the SQL Links installation program always:

- Installs an SQL Link driver that enables your application to access your SQL database.
- Adds new options to the IDAPI Configuration Utility so you can configure the driver default parameters and create an alias to an SQL database.
- Installs an online Help file you can use when configuring the driver.
- Installs READLINK.TXT, a text file containing late-breaking information.

Depending on which driver(s) you install, the installation program may install other driver support files or ask you for other information about the SQL server to which you plan to connect. For further information see the "Installation Changes" section of your driver-specific appendix.

## Installation procedure

To run the Borland SQL Links INSTALL program:

1. Insert SQL Links 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** (or **B:INSTALL**) at the workstation DOS prompt. INSTALL loads Windows and displays the opening dialog box.

   If Windows is already running:

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

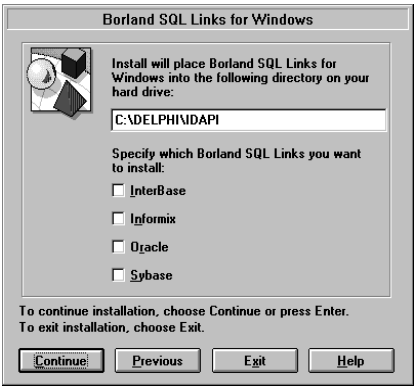
2 Enter **A:INSTALL (or B:INSTALL)** in the Command Line text box. INSTALL displays the opening dialog box.

**Figure 2.1** Opening dialog box

3 Select Next or press *Enter*. INSTALL checks whether the BDE's dynamic link library (IDAPI01.DLL) is present on the workstation disk. If necessary, it displays a dialog box where you can specify the location of this file.

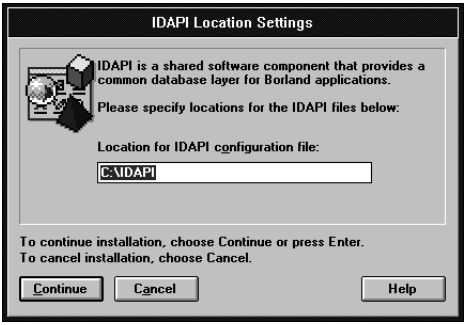
When the location of IDAPI01.DLL is established, INSTALL displays the Borland SQL Links for Windows Directory dialog box. Specify which SQL Link drivers you want to install.

**Figure 2.2** Specifying the Borland SQL Links home directory




When you are finished, select Next or press *Enter*. INSTALL displays the IDAPI Location Settings dialog box.

**Figure 2.3** Specifying where to install IDAPI.CFG



4 Specify the location where you want INSTALL to copy an updated version of your IDAPI configuration file.

When you are finished, select Next or press *Enter*.

Depending on which driver you selected, the Installation program may request further information before it proceeds. Your driver-specific appendix describes this information.

5 Once you supply all the necessary information, installation begins.

When installation is complete, you can choose to view the SQL Links README file, READLINK.TXT. Read this file to find out about late-breaking information.

Chapter

3

# Configuring the SQL environment

Installed with your BDE application is the IDAPI Configuration Utility (IDAPICFG.EXE), which enables users to modify their application configuration. The configuration parameters are stored in a binary file named IDAPI.CFG that the application reads at startup. Both files are usually located in the directory specified for BDE files during application installation (normally C:\IDAPI).

This chapter describes how to use the IDAPI Configuration Utility to set up your application's SQL environment. Once your BDE application is configured, you are ready to connect to the network and access the SQL server, as described at the end of this chapter.

For general instructions on how to use the IDAPI Configuration Utility, see your BDE application User's Guide or open the IDAPI Configuration Utility and select the Help button to display online Help.

**Before you begin**

- Be sure you have already installed the SQL Links software, as described in Chapter 2.
- Exit all other Borland applications.
- Open your application program group in the Windows Program Manager.
- Select the IDAPI Configuration Utility icon. The Configuration Utility window appears, open to the Driver Manager page.

## Specifying default SQL Link driver settings

An *alias* is a name and a set of parameters that describe a network resource. BDE applications use aliases to connect with shared databases. An alias is not required to access a local database, but it is required to access an SQL database.

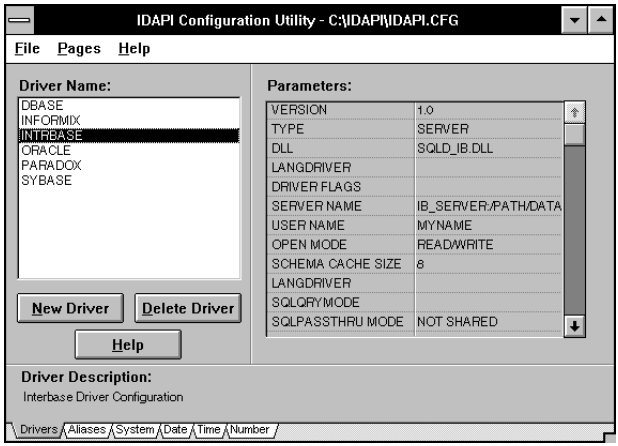
The SQL Link settings for your driver serve as a prototype for each new alias you create. Although you can customize an alias after you create it, it is easier to specify default driver settings that match your installation before you create any aliases for SQL databases. Then every alias you create will inherit these settings.

To specify default driver settings,

1 Highlight the entry for your driver. The Driver Manager displays all the configuration parameters for that driver.

   Use the scroll bars at the side of the Parameters list to examine the driver configuration parameters.

   **Figure 3.1** Sample IDAPI.CFG driver settings




2 If desired, edit the driver's default configuration. If you leave any categories blank, the Configuration Utility assumes you want to use the driver default.

3 When you are finished, select File | Save.

   Your changes take effect the next time you start your application.

The following sections describe configuration parameters that are common to all SQL Link drivers, in alphabetical order. For information on settings that are unique to a particular SQL Link driver, see the driver-specific appendix in this book or highlight the driver name and select Help in the IDAPI Configuration Utility.

## DLL

SQL Link dynamic link library name for the selected driver.

## DRIVER FLAGS

Internal product-specific flag.

## LANGDRIVER

Language driver used to manipulate all data that originates from the SQL server. When the cursor is in the LANGDRIVER field, a scroll bar appears at the right side of the text box; use the scroll bar to display a list of language driver long names that work with your driver. In the United States, the default value is blank.

When a specific language driver is associated with a server alias, your application uses this driver to manipulate all data that originates from the server. This includes all tables you view and all answer tables that result from a query. Servers running on different systems use conventions called character sets (or code pages) to determine how to encode alphabetic data. If you operate in a non-English language environment, your BDE application may use a different character set than the one used by the SQL server. When the character set at your desktop does not match the character set at the SQL server, passing alphabetic data between the two may cause either (or both) of the following:

- Data displays incorrectly on your desktop.
- The wrong characters are recorded in the SQL database.

To prevent this from happening, the SQL Links provide language drivers to convert character data between your application character set and the SQL server character set. This ensures that the characters from your server display correctly on your desktop, and that data you enter in the SQL server database is transmitted reliably.

If your SQL database uses extended character sets, make sure the alias you use to access the SQL server specifies the correct SQL Link language driver. Choose a language driver that uses the same character set in which the server passes data to your BDE application, and a collation sequence that matches your server's collation sequence. If you cannot find an appropriate SQL Link language driver, you may want to modify your alias SQLQRYMODE entry to prevent processing of queries under local database rules. (See "SQLQRYMODE," later in this section.)

## OPEN MODE

Mode in which SQL Links opens the SQL database. Possible values are READ/WRITE and READ ONLY. The default value is READ/WRITE.

**Note** Setting OPEN MODE to READ ONLY affects operations from the user interface, but has no effect on pass-through SQL.

## SCHEMA CACHE SIZE

Number of SQL tables whose schema information will be cached. Possible values are 0 - 32. The default value is 8.

## SCHEMA CACHE TIME

Specifies how long table list information will be cached. (In IDAPI this happens when you call either DbiOpenTableList or DbiOpenFileList.) Possible modes and their meanings are listed in Table 3.1. The default value is –1.

**Table 3.1** SCHEMA CACHE TIME settings

| Setting | Meaning |
|---|---|
| –1 | (Default) The table list is cached until you close the database. |

**Table 3.1** SCHEMA CACHE TIME settings (continued)

| Setting | Meaning |
|---|---|
| 0 | No table lists are cached. |
| 1 through 2147483647 | The table list is cached for the number of seconds specified in the setting. |

Setting this value can increase performance for table and file list retrieval.

## SERVER NAME

Name of the target SQL server. When specified for InterBase servers, includes the full path to the database file, i.e. `servername:/usr/gds/directoryname/databasename.gdb`.

## SQLPASSTHRU MODE

Specifies whether or not the Borland application will be able to access the SQL server via both desktop commands and pass-through SQL in the same alias connection. Possible modes and their meanings are listed in Table 3.2. The default value is SHARED AUTOCOMMIT for Informix; NOT SHARED for all other SQL Link drivers.

**Table 3.2** SQLPASSTHRU MODE settings

| Setting | Meaning |
|---|---|
| NOT SHARED<br>(blank setting) | Pass-through SQL and non-pass-through SQL do NOT share the same connection. |
| SHARED AUTOCOMMIT | Default. Pass-through SQL and non-pass-through SQL will share the same connection, and pass-through SQL will behave in a similar fashion to non-pass-through. This means that, as long as the user is not in an explicit client transaction or batch mode, pass-through SQL statements will be automatically committed. |
| SHARED NOAUTOCOMMIT | Pass-through SQL and non-pass-through SQL will share the same connection, but the SQL driver will *not* automatically commit pass-through statements. In this mode, pass-through behavior is server-dependent. |

SHARED AUTOCOMMIT and SHARED NOAUTOCOMMIT modes do not support all pass-through SQL statements.When SHARED AUTOCOMMIT or SHARED NOAUTOCOMMIT mode is set, do not execute transaction control statements in pass-through SQL; use your BDE application programming language to begin, commit, or roll back the transaction.

When pass-through SQL and non-pass-through queries share a single connection, the record cache does not immediately reflect updates performed in pass-through SQL.

For a discussion of how the application processes pass-through SQL queries, and information about using pass-though SQL with your Borland application, see your application product documentation.

## SQLQRYMODE

Method for handling queries to SQL data. Possible modes and their meanings are listed in Table 3.3. The default value is NULL (blank setting).

**Table 3.3** SQLQRYMODE settings

| Setting | Mode | Meaning |
|---|---|---|
| NULL (blank setting) | Server-local | (Default mode) In server-local query mode, the query goes first to the SQL server. If the server is unable to perform the query, the query is performed locally.<br>For a discussion of how Borland language drivers affect the processing of SQL queries, see "Blocking local processing of queries," below. |
| SERVER | Server-only | In server-only query mode, the query is sent to the SQL 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. |

### How language drivers affect SQL server query processing

Language drivers include information on *sort order* and *uppercasing* conventions used by your SQL server. Whenever a query to an SQL database is processed under the rules of the local (non-SQL) database, the application language driver is used in evaluating character ranges for sorting. If the sort order and uppercasing conventions at your desktop do not match the conventions used at the SQL server, your application may display inconsistent results. (A different sort order can cause the selection of a different set of records.)

To ensure consistent results you may want to set SQLQRYMODE to block local processing of queries, as described in the following section.

### Blocking local processing of queries

Normally queries to SQL databases are processed on the SQL server. However, in some cases, the result of a query executed completely on an SQL server will differ from that of a query executed locally. For example, say you use your BDE application's query function to perform a case-sensitive search on character fields. If the server does not support case-sensitive searches on a character field, then selecting on criteria such as ">A" will produce a different answer set depending on where the query is processed.

If you want to make sure that all queries originating from the BDE application are processed according to the rules of the SQL server, you can configure the SQL Link driver to block local processing of queries.

To do so, use the IDAPI Configuration Utility to modify the SQL database alias, setting SQLQRYMODE to SERVER. (See "Modifying an existing SQL alias.")

The new SQLQRYMODE value takes effect the next time you start your application.

**Note** Types of SQL database queries that are always processed locally will return an error message when SQLQRYMODE is set to SERVER. These include:

- Heterogenous queries (queries made across more than one type of data)
- Queries that cannot be expressed as a single SQL statement
- Queries that the SQL server does not support

### TYPE

Current driver type. SERVER notes that the driver is for connecting to an SQL server; FILE notes that the driver is for connecting to a standard, file-based server.

### USER NAME

Default user name for accessing the SQL server.

### VERSION

Internal Borland version number of the SQL Link driver.

## Creating and managing SQL aliases

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 assigning a name to, and customizing the access parameters for, an SQL server and database. Your SQL alias includes your user name and password on the target SQL server, and is required to access any SQL data through your IDAPI product. A generic SQL alias is automatically created the first time you modify the default SQL Link driver parameters after installation.

The following subsections describe how to use the Configuration Utility to add, modify, and delete aliases. For information on how to use an SQL database alias to connect to the SQL server, see "Connecting to the SQL server."

### Adding a new SQL alias

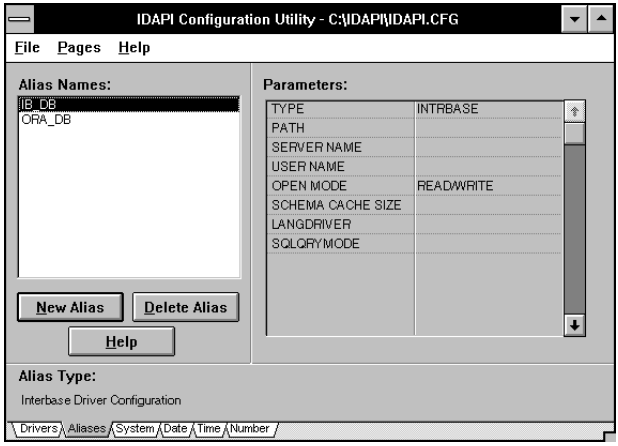
To add a new alias,

1 Select the Alias Manager (Aliases page) and choose the New Alias button. The Add New Alias dialog box appears.

**Figure 3.2** Sample Add New Alias dialog box

2 Enter the new alias name and select the SQL-specific alias type. Then choose OK to begin the setup process. The Alias Manager displays all the configuration parameters you can change to customize the new alias.

**Figure 3.3** Customizing the new alias

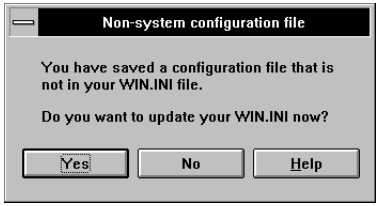



The parameters are the same as those listed in the Driver Manager for your driver type (Drivers page; refer to Figure 3.1). For a description of each setting you can change, refer to"Specifying default SQL Link driver settings."

3 If desired, edit the settings for the category you selected. If you leave any categories blank, the Alias Manager assumes you want to use the default for driver type.

4 When you are finished, select File | Save to save the new alias in the default configuration file; select File | Save as to save the new alias in a configuration file with a different name.

**Note** If you save the new alias in a configuration file with a different name, the IDAPI Configuration Utility displays:




Choose Yes if you want to activate this configuration file next time you start your application. Choose No if you want to keep using the current default configuration file.

Your changes take effect the next time you start your application.

## Modifying an existing SQL alias

This subsection describes how to modify the parameters in an existing SQL server alias. To change a default parameter type for all aliases of that driver type, follow the

procedure described earlier in "Specifying default SQL Link driver settings." When you modify a driver parameter, all aliases that use the default setting for that parameter inherit the new setting.

To modify an existing alias,

1. Scan the list of Alias Names available through the current configuration file. If the alias you want to modify was stored in a different configuration file, use File | Open to load that configuration file.
2. Highlight the name of the alias you wish to modify. The configuration for that alias appears in the Parameters section of the Alias Manager page.

   The parameters are the same as those listed in the Driver Manager (Drivers page; refer to Figure 3.1). For a description of each setting you can change, refer to "Specifying default SQL Link driver settings," earlier in this chapter.
3. Highlight the configuration parameter you wish to change, and enter the desired value. If you leave any categories blank, the Alias Manager assumes you want to use the driver's default value.
4. When you are finished, select File | Save to save the new alias in the default configuration file; select File | Save As to save the new alias in a configuration file with a different name.

Your changes take effect the next time you start your application.

### Deleting an SQL alias

To delete an alias,

1. Scan the list of Alias Names available through the current configuration file. If the alias you want to delete was stored in a different configuration file, use File | Open to load that configuration file.
2. Highlight the name of the alias you wish to modify, and select the Delete Alias button.
3. Select File | Save to save your changes in the default configuration file; select File | Save As to save your changes in a different configuration file.

## Connecting to the SQL server

Whenever you attempt an operation against an SQL 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 ensure that you have the right to access the database.

If your IDAPI files and your SQL Link driver are configured properly, you should be able to use your application's File menu to select a table in your SQL database through the appropriate alias. (Refer to "Creating and managing SQL aliases," earlier in this chapter.)

If you have trouble connecting, see the troubleshooting procedures in your driver-specific appendix and your BDE application documentation.

**Note** If you saved your new (or modified) alias in a configuration file with a different name, be sure to activate that configuration file in your Windows WIN.INI file before you start your BDE application. One way to do this is to update the IDAPI section of WIN.INI directly, changing the `CONFIGFILE01=` statement to reflect the desired configuration file. Another way to do this is to open and save the file through the IDAPI Configuration Utility. If directed at Save time, the IDAPI Configuration Utility automatically updates your WIN.INI file.

Appendix

A

# Connecting to Informix

This appendix describes how to troubleshoot common Borland Informix SQL Link connection problems, and discusses various topics about using Borland SQL Links that are unique to Informix.

## INFORMIX-SQL server requirements

Table A.1 lists software that should already be installed and running at the INFORMIX-SQL server before you install the Borland Informix SQL Link driver.

**Table A.1** Server software requirements

| Category | Description |
|---|---|
| Database server software | Informix V. 4.10 |
| Network protocol software | TCP/IP or SPX network protocol software compatible with both the database server and the client workstation network protocol |

## Client workstation requirements

Table A.2 lists software that should already be installed and running at the client workstation. It also lists related files and parameters.

**Table A.2** Client workstation requirements

| Category | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Informix PC client software | INFORMIX-NET/ PC Version 4.10. |
| Hardware and operating system requirements | 1.5 MB of free disk space<br>Hardware and operating system that meets the requirements of your Borland desktop product. |

**Table A.2** Client workstation requirements (continued)

| Category | Description |
|---|---|
| Access rights (for desktop products installed on the network server *only*) | If your Borland desktop product is installed on a network file server, make sure your network user account has Read and Write access rights to the product's BDE files (including IDAPICFG.EXE and the BDE configuration file). This directory is modified during SQL Links installation. |
| Network protocol software | TCP or SPX network protocol software compatible with both the server network protocol and the client workstation communication driver. |

# Database access requirements

Accessing the INFORMIX-SQL database requires a valid user identification and password on the Informix server, and at least Read (SELECT) access privileges. If you are a Netware user, you should also belong to the Network group "Informix" on the server where the Informix database resides.

To obtain these privileges, see your database administrator.

# Installation changes

When you install the Borland Informix SQL Link driver, the following changes take place in your workstation system:

**Table A.3** Installation changes for the INFORMIX SQL Link driver

| Item added | Description |
|---|---|
| SQLD_INF.DLL | Dynamic Link Library comprising the new INFORMIX SQL Link driver and its supporting files. |
| INFORMIX driver type | Added to Configuration Utility Driver Manager to enable basic configuration of Borland INFORMIX SQL Link driver. |
| INFORMIX alias type | Added to Configuration Utility Alias Manager to enable creation of an alias that can connect to an SQL Server database. |
| SQLD_INF.HLP | Help file for configuring INFORMIX driver. |
| READLINK.TXT | Borland SQL Links for Windows README file. |
| *.IEM files | Informix message files; any that do not already exist in C:\INFORMIX are added during SQL Links installation. |
| LDLLSQLW.DLL | Informix SQL library. |
| BLROM800.LD | SQL Link ROMAN8 language driver using binary collation sequence. |

# Special SQL Link driver settings

The following paragraphs describe Configuration Utility parameters that are unique to Informix.

## DATABASE NAME

Name of the target SQL database.

## DATE MODE

The format in which the driver sends dates to the SQL server. Valid settings are 0 (MDY) and 1 (DMY).

**Note** This setting must match the DBDATE environment variable on the Informix server.

## DATE SEPARATOR

The character used to separate day, month, and year for DATE MODE information. Valid characters are the forward slash (/), the period (.), and the dash (-).

The default character is the forward slash.

**Note** This setting must match the DBDATE environment variable on the Informix server.

## LOCK MODE

Defines how the server handles a process that tries to access a locked row or table. Table A.4 lists possible values and their meanings. The default value is 5 (blank setting).

**Table A.4** LOCK MODE settings

| Setting | SQL Statement Represented | Meaning |
|---|---|---|
| –1 | SET LOCK MODE TO WAIT | Suspends the process until the lock is released. |
| 0 | SET LOCK MODE TO NOT WAIT | Ends the operation immediately and returns an error code. This is the Informix default, which the Borland Informix driver now overrides. |
| 1–32766 | SET LOCK MODE TO WAIT *n* | (Default=5) Suspends the process until the lock is released, or until the end of the specified number of seconds.[1] |

1. In versions of Informix which do not support the seconds option, any non-zero value causes the process to wait until the lock is released.

**Note** Lock mode has no effect on exclusive locks. If a process attempts to access a row, table, or database that is locked in exclusive mode, the operation ends and an error code is returned.

# Borland language drivers for Informix

The following table lists language drivers available for use with Informix, their corresponding character sets, and Borland collation sequences. Choose the language

driver that uses the same collation sequence as your server, and the same character set as the one your server uses to pass data to your BDE application.

**Table A.5** Borland language drivers for Informix

| Long driver name | Short driver name | Character set | Collation sequence |
|---|---|---|---|
| Borland ENU Latin-1 | BLLT1US0 | ISO8859.1(ANSI) | Binary |
| dBASE FRA cp437 | DB437FR0 | DOS CODE PAGE 437 | dBASE French |
| dBASE FIN cp437 | DB437FI0 | DOS CODE PAGE 437 | dBASE Finnish |
| dBASE ENU cp437 | DB437US0 | DOS CODE PAGE 437 | dBASE English/US |
| dBASE NOR cp865 | DB865NO0 | DOS CODE PAGE 865 | dBASE Norwegian |
| dBASE SVE cp437 | DB437SV0 | DOS CODE PAGE 437 | dBASE Swedish |
| dBASE SVE cp850 | DB850SV1 | DOS CODE PAGE 850 | dBASE Swedish850 |
| dBASE ESP cp437 | DB437ES1 | DOS CODE PAGE 437 | dBASE Spanish |
| dBASE NLD cp437 | DB437NL0 | DOS CODE PAGE 437 | dBASE Dutch |
| dBASE ESP cp850 | DB850ES1 | DOS CODE PAGE 850 | dBASE Spanish850 |
| dBASE ENG cp437 | DB437UK0 | DOS CODE PAGE 437 | dBASE English/UK |
| dBASE ENU cp850 | DB850US0 | DOS CODE PAGE 850 | dBASE English/US |
| dBASE FRC cp863 | DB863CF1 | DOS CODE PAGE 863 | dBASE French Canadian |
| dBASE ENG cp850 | DB850UK0 | DOS CODE PAGE 850 | dBASE English850/UK |
| dBASE ITA cp850 | DB850IT1 | DOS CODE PAGE 850 | dBASE Italian850 |
| dBASE DEU cp850 | BD850DE0 | DOS CODE PAGE 850 | dBASE German850 |
| dBASE FRA cp850 | DB850FR0 | DOS CODE PAGE 850 | dBASE French850 |
| dBASE ITA cp437 | DB437IT0 | DOS CODE PAGE 437 | dBASE Italian |
| dBASE NLD cp850 | DB850NL0 | DOS CODE PAGE 850 | dBASE Dutch |
| dBASE FRC cp850 | DB850CF0 | DOS CODE PAGE 850 | dBASE French Canadian850 |
| dBASE DAN cp865 | DB865DA0 | DOS CODE PAGE 865 | dBASE Danish |
| dBASE DEU cp437 | DB437DE0 | DOS CODE PAGE 437 | dBASE German |
| 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 |

# Troubleshooting common connection problems

If you have problems connecting to the Informix database from within your BDE application,

1. Confirm that the Informix server is on line.

   If not—Consult your Informix database administrator.
   If so—Continue with step 2.

2. Check whether your workstation network software is running and properly configured.

   If you connect to Informix through a TCP/IP network, run PING or its equivalent; if you connect through a Novell network, run SLIST. These programs reflect whether your workstation can "see" the specified server using its network connection.

   If the Informix server does not appear—Consult your network administrator.
   If the Informix server appears—Continue with step 3.

3. Check whether you can establish a connection with the Informix server.

   Use the Informix **isql** utility or an equivalent, and try to connect to the Informix server.

   If **isql** does not work—Consult your Informix database administrator.
   If **isql** works—Continue with step 4.

4. Check whether your BDE application's Informix alias is set up properly.

   If you can connect directly from your workstation but not from within your BDE application, there is probably a problem with your IDAPI.CFG alias setup. Run the BDE Configuration Utility and examine your Informix alias.

For further information about Informix troubleshooting utilities, see your Informix documentation.

# Working with Informix servers

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

**Note** To improve query retrieval time, you may want to use the Informix UPDATE STATISTICS statement.

Table A.6 lists the general items that you might find helpful in working with Informix.

**Table A.6** General information about Informix servers

| Item | Description |
|---|---|
| Dynamic Link Library (DLL) name | SQLD_INF.DLL |
| Case-sensitive for data? | Yes (including pattern matching)[1] |

**Table A.6** General information about Informix servers (continued)

| Item | Description |
|---|---|
| Case-sensitive for objects (such as tables, columns, indexes)? | No (ANSI-compliant databases are case-sensitive on owner name only) |
| Does the server require an explicit request to begin a transaction for multi-statement transaction processing? | Yes (if your database is set up for transactions) |
| Does the server require that you explicitly start a transaction for multi-statement transaction processing in pass-through SQL? | Depends on how the Informix database was set up:<br>ANSI-compliant—NO<br>Logging database—YES<br>Non-logging database—NOT APPLICABLE |
| Implicit row IDs | Yes |
| Blob handles | No |
| Maximum size of single blob read | 64K |

1. Queries with pattern matching are case-insensitive, and therefore processed locally. For further information on query processing in SQL Links, see "Options for querying SQL data" in Chapter 1.

## Informix data type translations

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

Other database operations cause a conversion in the opposite direction, from Informix format to Paradox or dBASE format. For example, suppose you run a local query against one or more SQL tables. During the query, SQL Link converts any data originating in an SQL database to Paradox or dBASE format (depending on the answer format requested) before placing the data in the local answer table.

Tables A.7 through A.12 list Informix, Paradox, and dBASE data types and show how SQL Link translates between them in append, copy, and local query operations.

**Table A.7** Informix to Paradox and dBASE data type translations

| From Informix | To Paradox | To dBASE |
|---|---|---|
| Char ≤255 | Alpha | Character |
| Char >255 | Memo | Memo |
| Smallint | Short | Number {6.0} |
| Integer | Number | Number {11.0} |
| Smallfloat | Number | Float {20.4} |
| Float | Number | Float {20.4} |
| Money | Money | Float |
| Decimal | Number | Float |
| Date | Date | Date |
| DateTime | DateTime | Date |
| Interval | Alpha {25} | Character |
| Serial | Number | Number {11.0} |

**Table A.7** Informix to Paradox and dBASE data type translations (continued)

| From Informix | To Paradox | To dBASE |
|---|---|---|
| Byte | Binary | Memo |
| Text | Memo | Memo |
| Varchar | Alpha | Character |

**Table A.8** Paradox to dBASE and Informix data type translations

| From Paradox | To dBASE | To Informix |
|---|---|---|
| Alpha | Character | Char |
| Number | Float {20.4} | Float |
| Money | Float {20.4} | Money {16.2} |
| Date | Date | Date |
| Short | Number {6.0} | Smallint |
| Memo | Memo | Text |
| Binary | Memo | Byte |
| Formatted Memo | Memo | Byte |
| OLE | Memo | Byte |
| Graphic | Memo | Byte |
| Long | Number {11.0} | Integer |
| Time | Character {>8} | Character {>8} |
| DateTime | Character {>8} | DateTime |
| Bool | Bool | Character |
| AutoInc | Number {11.0} | Integer |
| Bytes | Bytes | Byte |
| BCD | N/A | N/A |

**Table A.9** dBASE to Paradox and Informix data type translations

| From dBASE | To Paradox | To Informix |
|---|---|---|
| Character | Alpha | Char |
| Number[1] | Short, Number | Smallint, Float |
| Float | Number | Float |
| Date | Date | Date |
| Memo | Memo | Text |
| Bool | Bool | Char |
| Bytes | Bytes | Byte |
| Lock | Alpha {24} | Char |

1. The dBASE Number data type translates to different Informix and Paradox data types depending on the WIDTH and DEC specification. A dBASE Number with a WIDTH less than 5 and a DEC equal to 0 translates to the Informix Smallint data type or the Paradox Short data type. dBASE Numbers with a WIDTH greater than 5 or a DEC greater than 0 translate into the Informix Float data type or the Paradox Number data type.

**Table A.10** Paradox physical to BDE logical and dBASE physical data type translations

| Paradox physical | BDE logical | dBASE physical |
|---|---|---|
| fldPDXCHAR | fldZSTRING | fldDBCHAR |
| fldPDXNUM | fldFLOAT | fldDBFLOAT {20.4} |
| fldPDXMONEY | fldFLOAT/fldstMONEY | fldDBFLOAT {20.4} |
| fldPDXDATE | fldDATE | fldDATE |
| fldPDXSHORT | fldINT16 | fldDBNUM {6.0} |
| fldPDXMEMO | fldBLOB/fldstMEMO | fldDBMEMO |
| fldPDXBINARYBLOB | fldBLOB/fldstBINARY | fldDBMEMO |
| fldPDXFMTMEMO | fldBLOB/fldstFMTMEMO | fldDBMEMO |
| fldPDXOLEBLOB | fldBLOB/fldstOLEOBJ | fldDBMEMO |
| fldPDXGRAPHIC | fldBLOB/fldstGRAPHIC | fldDBMEMO |
| fldPDXBLOB | fldPDXMEMO | fldDBMEMO |
| fldPDXLONG | fldINT32 | fldDBNUM {11.0} |
| fldPDXTIME | fldTIME | fldDBCHAR {>8} |
| fldPDXDATETIME | fldTIMESTAMP | fldDBCHAR {30} |
| fldPDXBOOL | fldBOOL | fldDBBOOL |
| fldPDXAUTOINC | fldINT32 | fldDBNUM {11.0} |
| fldPDXBYTES | fldBYTES | fldDBBYTES |
| fldPDXBCD | fldBCD | fldDBCHAR |

**Table A.11** dBASE physical to BDE logical and Paradox physical data type translations

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | |
| | fldINT16 | fldPDXSHORT |
| | else fldFLOAT | fldPDXNUM |
| fldDBMEMO | fldBLOB | fldPDXMEMO |
| fldDBBOOL | fldBOOL | fldPDXBOOL |
| fldDBDATE | fldDATE | fldPDXDATE |
| fldDBFLOAT | fldFLOAT | fldPDXNUM |
| fldDBLOCK | fldLOCKINFO | N/A |
| fldDBBINARY | fldBLOB/fldstTYPEDBINARY | fldPDXBINARYBLOB |
| fldDBOLEBLOB | fldBLOB/fldstDBSOLEOBJ | fldPDXOLEBLOB |

**Table A.12** Informix physical to BDE logical and Paradox and dBASE physical data type translations

| Informix physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldINFCHAR $\leq$ 255 | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldINFCHAR > 255 | fldBLOB | fldstMEMO | fldDBMEMO |
| fldINFSMALLINT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldINFINTEGER | fldINT32 | fldPDXLONG | fldDBNUM {11.0} |
| fldINFSMALLFLOAT | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldINFFLOAT | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |

**Table A.12** Informix physical to BDE logical and Paradox and dBASE physical data type translations (continued)

| Informix physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldINFMONEY | fldFLOAT/ fldstMONEY | fldPDXMONEY | fldDBFLOAT {20.4} |
| fldINFDECIMAL | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldINFDATE | fldDATE | fldPDXDATE | fldDBDATE |
| fldINFDATETIME | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |
| fldINFINTERVAL | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldINFSERIAL | fldINT32 | fldPDXLONG | fldDBNUM {11.0} |
| fldINFBYTE | fldBLOB/fldstBINARY | fldPDXBINARYBLOB | fldDBMEMO |
| fldINFTEXT | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldINFVARCHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |

### Using the Informix Interval data type

BDE applications currently have no data type that is directly equivalent to the Informix Interval data type.

Developers can create Interval fields with the Year-to-month qualifier on Informix servers; however, whenever you manipulate Informix Interval data with any range qualifiers through your BDE application, the SQL Link driver translates it into a character string.

For more information on the Informix Interval data type, see the *Informix Guide to SQL: Reference*.

## Transaction isolation levels

Informix supports four different levels of isolation among processes attempting to access data. A database's default isolation level is established according to database type, at the time the database is created:

- Dirty read—Default isolation level for non-logging databases. Provides no isolation.
- Committed read—Default isolation level for logging/non-ANSI databases. Guarantees that every retrieved row is committed in the table at the time that the row is retrieved, but does not prevent other processes from acquiring an exclusive lock on that same row. This makes it possible for another process to acquire an exclusive lock on that row, and modify or delete it, before you commit your transaction.
- Cursor stability—Not applicable to BDE applications.
- Repeatable read—Default isolation level for ANSI-compliant databases. Acquires a shared lock on every row selected during the transaction. This does not prevent another process from acquiring a shared lock on a selected row, but it does prevent another process from modifying any selected row during your transaction.

In some cases, when connecting with your Informix database, your BDE application overrides the current Informix transaction isolation settings. The following table shows under which circumstances these overrides occur.

**Table A.13** Default isolation levels in Informix and SQL Link

| Database | Default isolation level: Informix | Default isolation level: SQL Link |
|---|---|---|
| ANSI | RepeatableRead | CommittedRead |
| Logged | CommittedRead | CommittedRead |
| Non-logged | DirtyRead | DirtyRead |

## Informix field-naming rules

Table A.14 lists field-naming rules for Paradox, dBASE, and Informix.

**Table A.14** Informix field-naming rules

| Naming rule | Paradox | dBASE | Informix |
|---|---|---|---|
| Max length (characters) | 25 | 10 | 18 |
| Valid characters[1] | All | All except punctuation marks, blank spaces, and other special characters | Integers from 0 to 9, uppercase or lowercase letters, and underscore (_) character |
| Must begin with. . . | Any valid character except space | A letter | A letter |

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

Appendix

# B

# Connecting to InterBase

This appendix describes how to troubleshoot common InterBase SQL Link connection problems, and discusses various topics about using Borland SQL Links that are unique to InterBase.

## InterBase server requirements

Table B.1 lists software that should already be installed and running at the InterBase server before you install Borland SQL Links for Windows.

**Table B.1** Server software requirements

| Category | Description |
|---|---|
| Database server software | InterBase version 3.3 or higher |
| Network protocol software | Network protocol software compatible with both the database server and the client workstation network protocol |

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

## Client workstation requirements

Table B.2 lists software that should already be installed and running at the client workstation. It also lists related files and parameters.

**Table B.2** Client workstation

| Category | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |

**Table B.2** Client workstation (continued)

| Category | Description |
|---|---|
| Hardware and operating system requirements | 1.5 MB of free disk space.<br>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 a network file server, make sure your network user account has Read and Write access rights to the product's BDE files (including IDAPICFG.EXE and the BDE configuration file). 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. |
| HOSTS file | A HOSTS file containing the name and IP address of each server that you plan to attach. This file must contain the name and IP address of at least one host. For example:<br>`128.127.50.12 mis_server` |
| SERVICES file | A SERVICES file containing the protocol for InterBase server access. During SQL Links installation, this file is updated to include the line:<br>`gds_db 3050/tcp`<br>**Note:** If you prefer, you can add the line to your SERVICES file manually, after SQL Links installation. |

# Installation changes

When you install the InterBase SQL Link driver, the following items are installed in your workstation system:

**Table B.3** Installation changes for the InterBase SQL Link driver

| Item added | Description |
|---|---|
| SQLD_IB.DLL | Dynamic Link Library comprising the new InterBase driver and its supporting files. |
| INTRBASE driver type | Added to Configuration Utility Driver Manager to enable basic configuration of Borland InterBase SQL Link driver. |
| INTRBASE alias type | Added to Configuration Utility Alias Manager to enable creation of an alias that can connect to an SQL Server database. |
| SQLD_IB.HLP | Help file for configuring InterBase driver. |
| READLINK.TXT | Borland SQL Links for Windows README file. |
| INTERBAS.MSG | InterBase message files, usually installed in C:\INTERBAS. |
| CONNECT.EXE | Utility to test connection between the workstation and the InterBase server; see "TCP/IP interface." |
| REMOTE.DLL<br>GDS.DLL | InterBase-supplied.DLLs. |
| InterBase server specification, to InterBase SERVICES file | The installation updates the workstation SERVICES file to add the correct protocol specification for InterBase server access. The line should be similar to:<br>`gds_db 3050/tcp`<br>For further information, see your database administrator. |

## TCP/IP interface

The following files provide InterBase client applications their interface to Winsock 1.1 compliant TCP/IP products.

**Table B.4** Winsock 1.1 Client files

| File Name | Description |
|---|---|
| MVWASYNC.EXE | Asynchronous communication module |
| VSL.INI | TCP/IP transport initialization file |
| WINSOCK.DLL | Windows Socket DLL |
| MSOCKLIB.DLL | Maps Windows socket calls to VSL driver |

For TCP/IP products that are not Winsock 1.1 compliant, InterBase client applications will require one of the following files.

You can choose not to have the Installation program add a TCP/IP file by specifying "Use existing TCP file" during SQL Links for Windows installation.

**Table B.5** Non-Winsock Compliant TCP Support Files

| File Name | TCP/IP Product |
|---|---|
| M3OPEN.EXE | 3Com 3+Open TCP<br>Digital PATHWORKS<br>Microsoft LAN Manager TCP/IP |
| M3OPEN.DLL | 3Com 3+Open TCP Version 2.0 |
| MBW.EXE | Beame & Whiteside TCP/IP |
| MFTP.EXE | FTP PC/TCP |
| MHPARPA.DLL | HP ARPA Service for DOS |
| MNETONE.EXE | Ungermann-Bass Net/One |
| MNOVLWP.DLL | Novell LAN WorkPlace for DOS |
| MPATHWAY.DLL | Wollongong Pathway Access for DOS |
| MPCNFS.EXE | Sun PC NFS |
| MPCNFS2.EXE | Sun PC NFS v3.5 |
| MPCNFS4.DLL | Sun PC NFS v4.0 |
| MWINTCP.EXE | Wollongong WIN TCP\IP for DOS |

## Other communication protocols

The InterBase Workgroup Server for NetWare supports Novell SPX/IPX protocol. Two client files are required: NWIPXSPX.DLL, and NWCALLS.DLL.

The InterBase Workgroup Server for Widows NT supports Microsoft Named Pipes protocol. No additional client files are required to support Named Pipes, but the client machine must have Microsoft LAN Manager or Windows for Workgroups 3.1.1 installed.

# Testing the InterBase connection

To test whether you can connect to InterBase successfully, use the InterBase Connection Utility (CONNECT.EXE). This utility is stored in the same directory as the BDE files.

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

2 In the Command Line text box, enter the command to run CONNECT. (If you installed BDE files in C:\BDE, the command is **C:\BDE\CONNECT.EXE**.)

3 Choose OK. The InterBase Connect Utility dialog box appears.

**Figure B.1** InterBase Connect Utility dialog box

4 Enter information in each text box:

| Text box | Information required |
|---|---|
| Database Path | The path to an InterBase database, in the format:<br>`servername/usr/databaseDirectory/databaseName.gdb`<br>Be sure to use Unix-style forward-slash characters, and recall that Unix path names are case-sensitive. |
| User Name | A valid user name for the database you specified. |
| Password | A valid password for the user name you specified. |

5 Choose Connect to test your network connection.

If the connection succeeds, a status message appears.

If the connection does not succeed, an error message appears. (For information on how to troubleshoot your installation, see the following section.)

# Troubleshooting common connection problems

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

1 Run the Connection Utility (CONNECT.EXE) to determine if you can connect to the InterBase server from your client workstation. (Refer to "TCP/IP interface.")
   If CONNECT does not work—Consult your database administrator.
   If CONNECT works—Continue with step 2.

2 Verify that your InterBase SQL Link driver is correctly installed.

Reinstall SQL Link by following the procedures in *Getting Started*.

Also, check the SERVICES file for the correct protocol for InterBase server access. The line should be similar to:

```
gds_db   3050/tcp
```

If you are unable to install the driver correctly—Consult your database administrator.

If the driver is correctly installed—Continue with step 3.

**Note** The following steps require a TELNET program and a PING program. These DOS programs are not included in the SQL Link product package, but they are available from your TCP/IP network software vendor. (Your TCP/IP network software package may use different names for these programs.)

If you do not have these programs on your client workstation, ask your network administrator to perform these tests for you.

3 Test the lower-level protocols.

1 Enter the TELNET command to ensure that the TCP libraries are correctly installed.

If the TCP libraries are correctly installed, the `login:` prompt is displayed. Login to the network and check for the presence of the database you are trying to attach.

If the message `can't resolve hostname` is displayed, check your workstation HOSTS file to ensure that you have an entry for your host name and IP address. The entry looks similar to:

```
128.127.50.12          mis_server
```

If TELNET is successful and CONNECT is not, you may have a problem with your InterBase installation. See your database administrator for assistance.

2 PING the server to check that the InterBase server itself is running and visible to your desktop application. (If PING is successful, the message `servername is alive` is displayed.)

If PING is successful but the TELNET command is not, there may be a problem with the **inet** daemon.

If you cannot PING the server, you may have a routing problem. Report the problem to your network administrator.

**Note** If you don't have PING on your DOS client, you can PING the DOS client from the server node (if you have access to the server node). Ask your network administrator for instructions.

If the lower-level protocols do not seem to be running—Consult your database administrator.

If the lower-level protocols are running—Continue with step 4.

4 Confirm that you have a login set in the InterBase security database, isc.gdb.

   If so—Continue with step 5.

5 Check whether your BDE application InterBase alias is set up properly.

   If you can connect directly from your workstation but not from within your BDE application, there is probably a problem with your IDAPI.CFG alias setup. Run the Configuration Utility and examine your InterBase alias.

## Borland language drivers for InterBase

The following table lists language drivers available for use with InterBase and their corresponding InterBase subtypes. The language driver you choose must use the same collation sequence as your server, and the same character set as the one your server uses to pass data to your BDE application. The default can be set at either a database or a table level. (For further information, see your InterBase server documentation.)

InterBase supports subtypes for different fields in the same relation. However, rules of a language driver you specify will apply to a relation as a whole. The result of a query on a relation containing fields of different subtypes may vary according to where it was processed. In such a case, set SQLQRYMODE to SERVER to produce consistent query results.

**Table B.6** Borland language drivers for InterBase

| Long driver name | Short driver name | InterBase subtype |
|---|---|---|
| Paradox "ascii" | ascii | 0 (default), 1, 100, 101 |
| Borland DAN Latin-1 | BLLT1DA0 | 139 |
| DEU LATIN1 | BLLT1DE0 | 144 |
| ENG LATIN1 | BLLT1UK0 | 152 |
| ENU LATIN1 | BLLT1US0 | 153 |
| ESP LATIN1 | BLLT1ES0 | 149 |
| FIN LATIN1 | BLLT1FI0 | 141 |
| FRA LATIN1 | BLLT1FR0 | 142 |
| FRC LATIN1 | BLLT1CA0 | 143 |
| ISL LATIN1 | BLLT1IS0 | 145 |
| ITA LATIN1 | BLLT1IT0 | 146 |
| NLD LATIN1 | BLLT1NL0 | 140 |
| NOR LATIN1 | BLLT1NO0 | 105 |
| PTG LATIN1 | BLLT1PT0 | 154 |
| Paradox INTL | INTL | 102 |
| Pdox NORDAN4 | NORDAN40 | 105 |
| Pdox SWEDFIN | SWEDFIN | 106 |
| SVE LATIN1 | BLLT1SV0 | 151 |

**Note** For information on InterBase subtypes that correspond to dBASE language drivers, contact Borland Technical Support.

# Working with InterBase servers

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

Table B.7 lists the general items that you might find helpful in working with InterBase.

**Table B.7** General information about InterBase servers

| Item | Description |
|---|---|
| Dynamic Link Library (DLL) name | SQLD_IB.DLL |
| Case-sensitive for data? | Yes (including pattern matching) |
| Case-sensitive for objects (such as tables, columns, indexes)? | No |
| Does the server require an explicit request to begin a transaction for multistatement transaction processing? | Yes |
| Does the server require that you explicitly start a transaction for multi-statement transaction processing in pass-through SQL? | No |
| Implicit row IDs | No |
| BLOB handles | InterBase BLOBs have handles. However, InterBase CHAR and VARCHAR columns that are more than 255 characters long are treated as non-handle BLOBs. |
| Maximum size of single BLOBs read (if BLOB handles are not supported) | 32K |

## InterBase data type translations

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

Other database operations cause a conversion in the opposite direction, from InterBase format to Paradox or dBASE format. For example, suppose you run a local query against one or more SQL tables. During the query, SQL Link converts any data originating in an SQL database to Paradox or dBASE format (depending on the answer format requested) before placing the data in the local answer table.

Tables B.8 through B.13 list InterBase, Paradox, and dBASE data types and show how SQL Link translates between them in append, copy, and local query operations.

**Table B.8** InterBase to Paradox and dBASE data type translations

| FROM: InterBase | TO: Paradox | TO: dBASE |
|---|---|---|
| SHORT | Short | Number {6.0} |
| LONG | Number | Number {11.0} |
| FLOAT | Number | Float {20.4} |
| DOUBLE | Number | Float {20.4} |
| DATE[1] | DateTime | Date |

**Table B.8** InterBase to Paradox and dBASE data type translations (continued)

| FROM: InterBase | TO: Paradox | TO: dBASE |
|---|---|---|
| BLOB | Binary | Memo |
| BLOB/1 | Memo | Memo |
| CHAR(1-255) | Alphanumeric(n) | Character(n)[2] |
| CHAR(greater than 255) | Memo | Memo |
| VARYING(1-255) | Alphanumeric(n) | Character(n)[2] |
| VARYING(greater than 255) | Memo | Memo |
| ARRAY[2] | Binary | Memo |

1. From InterBase, QBE maps InterBase DATE to Paradox Date. Copy table maps InterBase DATE to Paradox Char(n).
2. Although an InterBase ARRAY is mapped to Paradox and dBASE data types, the resulting fields appear to be empty when displayed within your client product.

**Table B.9** Paradox to InterBase and dBASE data type translations

| FROM: Paradox | TO: InterBase | TO: dBASE |
|---|---|---|
| Alphanumeric(n) | VARYING(n) | Character(n) |
| Number | DOUBLE | Float {20.4} |
| Money | DOUBLE | Float {20.4} |
| Date | DATE | Date |
| Short | SHORT | Number {6.0} |
| Memo | BLOB/1 (Text) | Memo |
| Formatted memo | BLOB (Binary) | Memo |
| Binary | BLOB (Binary) | Memo |
| Graphic | BLOB (Binary) | Memo |
| OLE | BLOB (Binary) | Memo |
| Long | Long | Number {11.0} |
| Time | Character {>8} | Character {>8} |
| DateTime | Date | Character {>8} |
| Bool | Character {1} | Bool |
| AutoInc | Long | Number {11.0} |
| Bytes | BLOB | Bytes |
| BCD | N/A | N/A |

**Table B.10** dBASE to InterBase and Paradox data type translations

| FROM: dBASE | TO: InterBase | TO: Paradox |
|---|---|---|
| Character(n) | VARYING(n) | Alphanumeric(n) |
| Number | SHORT, DOUBLE | Short number, Number |
| Float[1] | DOUBLE | Number |
| Date | DATE | Date |
| Lock | Character {24} | Alpha {24} |
| Bytes | BLOB | Bytes |

**Table B.10** dBASE to InterBase and Paradox data type translations (continued)

| FROM: dBASE | TO: InterBase | TO: Paradox |
|---|---|---|
| Bool | Character {1} | Bool |
| Memo | BLOB/1 | Memo |

1. dBASE data types Number and Float translate to different InterBase 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 InterBase SHORT or Paradox Short Number data types.

**Table B.11** Paradox to BDE logical to dBASE data type translations

| Paradox physical | BDE logical | dBASE |
|---|---|---|
| fldPDXCHAR | fldZSTRING | fldDBCHAR |
| fldPDXNUM | fldFLOAT | fldDBFLOAT {20.4} |
| fldPDXMONEY | fldFLOAT/fldstMONEY | fldDBFLOAT {20.4} |
| fldPDXDATE | fldDATE | fldDATE |
| fldPDXSHORT | fldINT16 | fldDBNUM {6.0} |
| fldPDXMEMO | fldBLOB/fldstMEMO | fldDBMEMO |
| fldPDXBINARYBLOB | fldBLOB/fldstBINARY | fldDBMEMO |
| fldPDXFMTMEMO | fldBLOB/fldstFMTMEMO | fldDBMEMO |
| fldPDXOLEBLOB | fldBLOB/fldstOLEOBJ | fldDBMEMO |
| fldPDXGRAPHIC | fldBLOB/fldstGRAPHIC | fldDBMEMO |
| fldPDXBLOB | fldPDXMEMO | fldDBMEMO |
| fldPDXLONG | fldINT32 | fldDBNUM {11.0} |
| fldPDXTIME | fldTIME | fldDBCHAR {>8} |
| fldPDXDATETIME | fldTIMESTAMP | fldDBCHAR {30} |
| fldPDXBOOL | fldBOOL | fldDBBOOL |
| fldPDXAUTOINC | fldINT32 | fldDBNUM {11.0} |
| fldPDXBYTES | fldBYTES | fldDBBYTES |
| fldPDXBCD | fldBCD | fldDBCHAR |

**Table B.12** dBASE to BDE logical to Paradox data type translations

| dBASE physical | BDE logical | Paradox |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | |
| | fldINT16 | fldPDXSHORT |
| | else fldFLOAT | fldPDXNUM |
| fldDBMEMO | fldBLOB | fldPDXMEMO |
| fldDBBOOL | fldBOOL | fldPDXBOOL |
| fldDBDATE | fldDATE | fldPDXDATE |
| fldDBFLOAT | fldFLOAT | fldPDXNUM |
| fldDBLOCK | fldLOCKINFO | fldPDXCHAR {24} |

**Table B.12** dBASE to BDE logical to Paradox data type translations (continued)

| dBASE physical | BDE logical | Paradox |
|---|---|---|
| fldDBBINARY | fldBLOB/ fldstTYPEDBINARY | fldPDXBINARYBLOB |
| fldDBOLEBLOB | fldBLOB/fldstDBSOLEOBJ | fldPDXOLEBLOB |

**Table B.13** InterBase to BDE logical to Paradox and dBASE data type translations

| InterBase physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldIBSHORT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldIBLONG | fldINT32 | fldPDXLONG | fldDBNUM {11.0} |
| fldIBFLOAT | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldIBDOUBLE | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldIBCHAR $\leq$ 255 | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldIBCHAR > 255 | fldBLOB | fldSTMEMO | fldDBCHAR |
| fldIBVARYING $\leq$ 255 | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldIBVARYING > 255 | fldBLOB | fldSTMEMO | fldDBCHAR |
| fldIBDATE | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |
| fldIBBLOB | fldBLOB | fldPDXBINARYBLOB | fldDBMEMO |
| fldIBTEXTBLOB | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |

## InterBase equivalents to standard SQL data types

When you use pass-through SQL commands to create or alter an InterBase table, you must use standard SQL data types. Table B.14 lists standard SQL data types and their corresponding InterBase data types.

**Table B.14** SQL to InterBase data type translations

| FROM: SQL | TO: InterBase |
|---|---|
| SMALLINT | SHORT |
| INTEGER | LONG |
| DATE | DATE |
| CHAR(n) | CHAR(n) |
| VARCHAR(n) | VARYING |
| DECIMAL | LONG |
| FLOAT | FLOAT |
| LONG FLOAT | DOUBLE |
| BLOB | BLOB |

**Note** SQL does not support the InterBase ARRAY data type.

## InterBase system relations/tables

InterBase includes a special set of tables called *system relations*. System relations describe privileges, indexes, SQL table structures, and other items that define relationships

within a database. You can access system relations with pass-through SQL from your desktop product through the SQL Editor (see your desktop application documentation).

Table B.15 lists InterBase system relations you can access through SQL Link.

**Table B.15** Selected InterBase system relations

| Table name | Use |
|---|---|
| RDB$RELATIONS | Lists all tables and views |
| RDB$RELATION_FIELDS | Lists columns of tables and views |
| RDB$INDICES | Lists indexes |

## InterBase field-naming rules

Table B.16 lists field-naming rules for Paradox, dBASE, and InterBase.

**Table B.16** InterBase field-naming rules

| Naming rule | Paradox | dBASE | InterBase |
|---|---|---|---|
| Max length (characters) | 25 | 10 | 31 |
| Valid characters[1] | All | All alphanumeric except punctuation marks, blank spaces, and other special characters | Letters (A-Z, a-z), digits, $, or _ |
| Must begin with. . . | Any valid character except space | A letter | Letters only (A-Z, a-z) |

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

**Note** You cannot use InterBase reserved words for table names. See the *InterBase Language Reference* for a list of reserved words.

Appendix

C

# Connecting to ORACLE

This appendix describes how to troubleshoot common Borland ORACLE SQL Link connection problems, and discusses various topics about using Borland SQL Links that are unique to ORACLE.

## ORACLE server requirements

Table C.1 lists software that should already be installed and running at the ORACLE server before you install the Borland ORACLE SQL Link driver.

**Table C.1** Server software requirements for ORACLE

| Category | Description |
|---|---|
| Database server software | ORACLE version 6.0 or greater |
| Network protocol software | ORACLE SQL*NET |

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

## Client workstation requirements

Table C.2 lists software that should already be installed and running at the client workstation. It also lists related files and parameters.

**Table C.2** Client workstation requirements

| Category | Description |
|---|---|
| Network protocol software | Network protocol software compatible with both the server network protocol and the client workstation client database communication driver.<br>SQL*Net driver (.EXE or.DLL) |

# Installation changes

When you install the Borland ORACLE SQL Link driver, the following changes take place in your workstation system:

**Table C.3** Installation changes for the ORACLE SQL Link driver

| Item added | Description |
|---|---|
| SQLD_ORA.DLL | Dynamic Link Library comprising the new ORACLE SQL Link driver and its supporting files. |
| ORACLE driver type | Added to Configuration Utility Driver Manager to enable basic configuration of Borland ORACLE SQL Link driver. |
| ORACLE alias type | Added to Configuration Utility Alias Manager to enable creation of an alias that can connect to an ORACLE database. |
| SQLD_ORA.HLP | Help file for configuring ORACLE driver. |
| READLINK.TXT | Borland SQL Links for Windows README file. |
| ORA6WIN.DLL | Windows Dynamic Link Library for ORACLE 6. |
| ORA7WIN.DLL | Windows Dynamic Link Library for ORACLE 7. |
| SQL13WIN.DLL | ORACLE-supplied DLL. |
| SQLWIN.DLL | ORACLE-supplied DLL. |
| ORAWE850.LD | Language driver using Western European sort order; based on DOS code page 850. |
| BLROM800.LD | SQL Link ROMAN8 language driver using binary collation sequence. |

# Special SQL Link driver settings

The following paragraphs describe driver configuration parameters that are unique to ORACLE.

## NET PROTOCOL

Network transport used to communicate with the SQL server. Table C.4 describes NET PROTOCOL options.

**Table C.4** NET PROTOCOL settings

| 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 |

**Table C.4** NET PROTOCOL settings (continued)

| Value | Description |
|---|---|
| TCP/IP | Transport Control Protocol/Internet Protocol, as used by UNIX and VAX workstations |
| VINES | Banyan VINES protocol |

### VENDOR INIT

Vendor-specific initialization information; identifies whether ORA6WIN.DLL or ORA7WIN.DLL is installed at the workstation. If no value is specified here, SQL Link looks first for ORA6WIN.DLL, then ORA7WIN.DLL.

## Borland language drivers for ORACLE

The following table lists language drivers available for use with ORACLE, their corresponding character sets, and Borland collation sequences. Choose the language driver that uses the same collation sequence as your server, and the same character set as the one your server uses to pass data to your BDE application.

**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.

**Table C.5** Borland language drivers for ORACLE

| Long driver name | Short driver name | Character set | Collation sequence |
|---|---|---|---|
| Borland ENU Latin-1 | BLLT1US0 | ISO8859.1(ANSI) | Binary |
| dBASE FRA cp437 | DB437FR0 | DOS CODE PAGE 437 | dBASE French |
| dBASE FIN cp437 | DB437FI0 | DOS CODE PAGE 437 | dBASE Finnish |
| dBASE ENU cp437 | DB437US0 | DOS CODE PAGE 437 | dBASE English/US |
| dBASE NOR cp865 | DB865NO0 | DOS CODE PAGE 865 | dBASE Norwegian |
| dBASE SVE cp437 | DB437SV0 | DOS CODE PAGE 437 | dBASE Swedish |
| dBASE SVE cp850 | DB850SV1 | DOS CODE PAGE 850 | dBASE Swedish850 |
| dBASE ESP cp437 | DB437ES1 | DOS CODE PAGE 437 | dBASE Spanish |
| dBASE NLD cp437 | DB437NL0 | DOS CODE PAGE 437 | dBASE Dutch |
| dBASE ESP cp850 | DB850ES1 | DOS CODE PAGE 850 | dBASE Spanish850 |
| dBASE ENG cp437 | DB437UK0 | DOS CODE PAGE 437 | dBASE English/UK |
| dBASE ENU cp850 | DB850US0 | DOS CODE PAGE 850 | dBASE English/US |
| dBASE FRC cp863 | DB863CF1 | DOS CODE PAGE 863 | dBASE French Canadian |
| dBASE ENG cp850 | DB850UK0 | DOS CODE PAGE 850 | dBASE English850/UK |
| dBASE ITA cp850 | DB850IT1 | DOS CODE PAGE 850 | dBASE Italian850 |
| dBASE DEU cp850 | BD850DE0 | DOS CODE PAGE 850 | dBASE German850 |
| dBASE FRA cp850 | DB850FR0 | DOS CODE PAGE 850 | dBASE French850 |
| dBASE ITA cp437 | DB437IT0 | DOS CODE PAGE 437 | dBASE Italian |
| dBASE NLD cp850 | DB850NL0 | DOS CODE PAGE 850 | dBASE Dutch |

**Table C.5** Borland language drivers for ORACLE (continued)

| Long driver name | Short driver name | Character set | Collation sequence |
|---|---|---|---|
| dBASE FRC cp850 | DB850CF0 | DOS CODE PAGE 850 | dBASE French Canadian850 |
| dBASE DAN cp865 | DB865DA0 | DOS CODE PAGE 865 | dBASE Danish |
| dBASE DEU cp437 | DB437DE0 | DOS CODE PAGE 437 | dBASE German |
| Oracle SQL WE850 | ORAWE850 | DOS CODE PAGE 850 | ORACLE multi-lingual Western European sort order |
| 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 |

# Troubleshooting common connection problems

If you have problems establishing an ORACLE connection with SQL Link,

1 Confirm that the ORACLE server is online and the correct SQL*Net listener is running.

   If not—Consult your database administrator.
   If so—Continue with step 2.

**Note** The easiest way to ensure that all of these parameters are set correctly is to reinstall your ORACLE workstation software (for example, 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 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.

   If not—Consult your database administrator.
   If so—Continue with step 3.

3 Use your ORACLE tools to verify the connection at each layer.

   1 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;
```

2 Verify that your AUTOEXEC.BAT file contains the lines:

```
SET CONFIG=C:\ORACLE6\CONFIG.ORA
PATH=C:\;C:\DOS;C:\ORACLEx;C:\ORACLEx\BIN
```

**ORACLEx** is the name of your ORACLE directory.

If not—Consult your database administrator.
If so—Continue with step 4.

4 Check whether your BDE application ORACLE alias is set up properly.

If you can connect directly from your workstation but not from within your BDE application, there is probably a problem with your IDAPI.CFG alias setup. Run the Configuration Utility and examine your ORACLE alias.

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 C.6 lists the general items that you might find helpful in working with ORACLE servers.

**Table C.6** General information about ORACLE servers

| Item | Description |
|---|---|
| Dynamic Link Library (DLL) name | SQLD_ORA.DLL |
| Case-sensitive for data? | Yes |
| Case-sensitive for objects (such as tables, columns, and indexes)? | No |
| Does the server require an explicit request to begin a transaction for multistatement transaction processing? | Yes |
| Does the server require that you explicitly start a transaction for multistatement transaction processing in pass-through SQL? | 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 Paradoxor dBASE format to ORACLE format. For example, a BDE application that copies or appends data

from a local table to an ORACLE table causes SQL Link to convert the local 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 local query against one or more SQL tables. During the query, SQL Link converts any data originating in an SQL database to Paradox or dBASE format (depending on the answer format requested) before placing the data in the local answer table.

Tables C.7 through C.12 list ORACLE, Paradox, and dBASE data types and show how SQL Link translates between these data types.

**Table C.7** ORACLE to Paradox and dBASE data type translations

| FROM: ORACLE | TO: Paradox | TO: dBASE |
|---|---|---|
| CHAR(*n*)<br>VARCHAR(*n*) | Alphanumeric(*n*) | Character(*n*)[1] |
| DATE[2] | DateTime | Date |
| FLOAT[3] | Number | Number {20.4} |
| LONG | Memo | Memo |
| LONG RAW | Binary | Memo |
| NUMBER | Number | Float {20.4} |
| RAW | Binary | Memo |
| VARCHAR2 (CHAR ≤255) | Alphanumeric(*n*) | Character(*n*)[1] |
| VARCHAR2 (CHAR >255) | Memo | Memo |

1. dBASE character data type supports only 254 characters. If you exceed this limit your data will be truncated.
2. From ORACLE, Paradox QBE maps ORACLE DATE to Paradox Date. Copy table maps ORACLE DATE to Paradox Char(*n*).
3. Corresponds to the ANSI FLOAT data type, as differentiated from the standard ORACLE NUMBER data types.

**Table C.8** Paradox to ORACLE and dBASE data type translations

| FROM: Paradox | TO: ORACLE | TO: dBASE |
|---|---|---|
| Alphanumeric(*n*) | CHAR(*n*) | Character(*n*) |
| Number | NUMBER | Float {20.4} |
| Money | NUMBER | Float {20.4} |
| Date | DATE | Character {date/time} |
| Short | NUMBER | Number {6.0} |
| Memo | LONG | Memo |
| Formatted memo | LONG RAW | Memo |
| Binary | LONG RAW | Memo |
| Graphic | LONG RAW | Memo |
| OLE | LONG RAW | Memo |
| Long | NUMBER | Number {11.0} |
| Time | CHARACTER {>8} | Character {>8} |

**Table C.8** Paradox to ORACLE and dBASE data type translations (continued)

| FROM: Paradox | TO: ORACLE | TO: dBASE |
|---|---|---|
| Datetime | DATE | Character {>8} |
| Logical | CHARACTER {1} | Bool |
| AutoInc | NUMBER | Number {11.0} |
| Bytes | LONG RAW | BYTES |
| BCD | N/A | N/A |

**Table C.9** dBASE to ORACLE and Paradox data type translations

| FROM: dBASE | TO: ORACLE | TO: Paradox |
|---|---|---|
| Character | CHAR(*n*) | Alphanumeric(*n*) |
| Number[1] | NUMBER | Short, Number |
| Float[1] | NUMBER | Short, Number |
| Date | DATE | Date |
| Logical | CHAR(1) | Bool |
| Memo | LONG | Memo |
| Bytes | LONG RAW | Bytes |
| Lock | CHAR(24) | Alphanumeric {24} |

1. dBASE data types Number and Float translate to different ORACLE 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 ORACLE NUMBER or Paradox Short Number data types.

**Table C.10** Paradox to BDE logical and dBASE physical data type translations

| Paradox physical | BDE logical | dBASE physical |
|---|---|---|
| fldPDXCHAR | fldZSTRING | fldDBCHAR |
| fldPDXNUM | fldFLOAT | fldDBFLOAT {20.4} |
| fldPDXMONEY | fldFLOAT/fldstMONEY | fldDBFLOAT {20.4} |
| fldPDXDATE | fldDATE | fldDATE |
| fldPDXSHORT | fldINT16 | fldDBNUM {6.0} |
| fldPDXMEMO | fldBLOB/fldstMEMO | fldDBMEMO |
| fldPDXBINARYBLOB | fldBLOB/fldstBINARY | fldDBMEMO |
| fldPDXFMTMEMO | fldBLOB/fldstFMTMEMO | fldDBMEMO |
| fldPDXOLEBLOB | fldBLOB/fldstOLEOBJ | fldDBMEMO |
| fldPDXGRAPHIC | fldBLOB/fldstGRAPHIC | fldDBMEMO |
| fldPDXBLOB | fldPDXMEMO | fldDBMEMO |
| fldPDXLONG | fldINT32 | fldDBNUM {11.0} |
| fldPDXTIME | fldTIME | fldDBCHAR {>8} |
| fldPDXDATETIME | fldTIMESTAMP | fldDBCHAR {30} |
| fldPDXBOOL | fldBOOL | fldDBBOOL |
| fldPDXAUTOINC | fldINT32 | fldDBNUM {11.0} |
| fldPDXBYTES | fldBYTES | fldDBBYTES |
| fldPDXBCD | fldBCD | fldDBCHAR |

**Table C.11** dBASE physical to BDE logical and Paradox physical type translations

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | |
| | fldINT16 | fldPDXSHORT |
| | else fldFLOAT | fldPDXNUM |
| fldDBMEMO | fldBLOB | fldPDXMEMO |
| fldDBBOOL | fldBOOL | fldPDXBOOL |
| fldDBDATE | fldDATE | fldPDXDATE |
| fldDBFLOAT | fldFLOAT | fldPDXNUM |
| fldDBLOCK | fldLOCKINFO | fldPDXCHAR {24} |
| fldDBBINARY | fldBLOB/fldstTYPEDBINARY | fldPDXBINARYBLOB |
| fldDBOLEBLOB | fldBLOB/fldstDBSOLEOBJ | fldPDXOLEBLOB |

**Table C.12** ORACLE physical to BDE logical and Paradox and dBASE physical data type translations

| ORACLE physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldORACHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldORARAW | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldORADATE | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |
| fldORANUMBER | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldORALONG | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldORALONGRAW | fldBLOB/fldstBINARY | fldPDXBINARYBLOB | fldDBMEMO |
| fldORAVARCHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldORAVARCHAR2 ≤255 | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldORA VARCHAR2 >255 | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldORAFLOAT | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |

## 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.

Table C.13 lists ORACLE system tables you can access through SQL Link.

**Table C.13** 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. |
| ALL_CATALOG | Lists all objects accessible to the user. |

**Table C.13** Selected ORACLE system tables (continued)

| Table Name | Use |
|---|---|
| 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 C.14 lists field-naming rules for Paradox, dBASE, and ORACLE.

**Table C.14** 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 (), the combination ->, or the symbol # alone.

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

Appendix

D

# Connecting to SYBASE and Microsoft SQL Servers

This booklet describes how to troubleshoot common Borland SYBASE and Microsoft SQL Server SQL Link connection problems, and discusses various topics about using Borland SQL Links that are unique to SQL Server.

## SQL Server requirements

Table D.1 lists software that should already be installed and running at the SQL Server before you install the Borland SQL Server SQL Link driver.

**Table D.1** Server software requirements for SYBASE

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

# Client workstation requirements

Table D.2 lists software that should already be installed and running at the client workstation. It also lists related files and parameters.

**Table D.2** Client workstation requirements

| Category | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Hardware and operating system requirements | 1.5 MB of free disk space.<br>Hardware and operating system that meets the requirements of your BDE application. |
| Access rights (for applications installed on the network server *only*) | If your BDE application 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 application's BDE files (including IIDAPICFG.EXE and the BDE configuration file). 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.<br>Net library driver. |

# Installation changes

When you install the Borland SQL Server SQL Link driver (called SYBASE in the Configuration Utility), the following changes take place in your workstation system:

**Table D.3** Installation changes for the SYBASE SQL Link driver

| Item added | Description |
|---|---|
| SQLD_SS.DLL | Dynamic Link Library comprising the new SYBASE driver and its supporting files. |
| SYBASE driver type | Added to Configuration Utility Driver Manager to enable basic configuration of Borland SYBASE SQL Link driver. |
| SYBASE alias type | Added to Configuration Utility Alias Manager to enable creation of an alias that can connect to an SQL Server database. |
| SQLD_SS.HLP | Help file for configuring SYBASE driver. |
| READLINK.TXT | Borland SQL Links for Windows README file. |
| W3DBLIB.DLL | SYBASE/Microsoft SQL Server library. |
| DBNMP3.DLL | SYBASE/Microsoft SQL Server Named Pipes library. |
| BLROM800.LD | SQL Link ROMAN8 language driver using a binary collation sequence. |
| SYDC850.LD | SYBASE language driver with a dictionary supporting case-sensitive collation; based on DOS code page 850. |
| SYDC437.LD | SYBASE language driver with a dictionary supporting case-sensitive collation; based on DOS code page 437. |

# Special SQL Link driver settings

The following paragraphs describe driver configuration parameters that are unique to SYBASE.

## Blob EDIT LOGGING

Enables or disables the logging of any edits to blob (Binary Large OBject) fields. Possible values are TRUE (the default) or FALSE. When set to FALSE, this option helps minimize blob space requirements and increase performance.

**Note** 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.

## CONNECT TIMEOUT

Specifies the amount of time the workstation will retry when attempting to attach to the SQL server. The default is 60 seconds.

## DATABASE NAME

Name of the target SQL database.

## DATE MODE

The format in which the driver sends dates to the SQL server. Valid settings are 0 (MDY), 1 (DMY), and 2 (YMD).

## TIMEOUT

Specifies the maximum amount of time that the workstation waits for results to return from the server. The default is 500 seconds.

You may want to change the TIMEOUT value to minimize conflicts between operations. For best performance, determine how much time a complex query would take to complete successfully on your network, then set TIMEOUT to a slightly larger number than that. For example, if you often run complex queries that can take up to 120 seconds to complete, be sure to set the TIMEOUT value to more than 120 seconds.

## MAX QUERY TIME

Specifies the maximum amount of time that SQL Link will wait for query execution to complete before canceling the operation. The default is 3,600 seconds.

# Borland language drivers for SQL Server

The following table lists language drivers available for use with SQL Server, their corresponding character sets, and Borland collation sequences. The language driver you choose must use the same collation sequence as your server, and the same character set as the one your server uses to pass data to your BDE application.

**Table D.4** Borland language drivers for SYBASE

| Long driver name | Short driver name | Character set | Collation sequence |
|---|---|---|---|
| Borland ENU Latin-1 | BLLT1US0 | ISO8859.1(ANSI) | Binary |
| dBASE FRA cp437 | DB437FR0 | DOS CODE PAGE 437 | dBASE French |
| dBASE FIN cp437 | DB437FI0 | DOS CODE PAGE 437 | dBASE Finnish |
| dBASE ENU cp437 | DB437US0 | DOS CODE PAGE 437 | dBASE English/US |
| dBASE NOR cp865 | DB865NO0 | DOS CODE PAGE 865 | dBASE Norwegian |
| dBASE SVE cp437 | DB437SV0 | DOS CODE PAGE 437 | dBASE Swedish |
| dBASE SVE cp850 | DB850SV1 | DOS CODE PAGE 850 | dBASE Swedish850 |
| dBASE ESP cp437 | DB437ES1 | DOS CODE PAGE 437 | dBASE Spanish |
| dBASE NLD cp437 | DB437NL0 | DOS CODE PAGE 437 | dBASE Dutch |
| dBASE ESP cp850 | DB850ES1 | DOS CODE PAGE 850 | dBASE Spanish850 |
| dBASE ENG cp437 | DB437UK0 | DOS CODE PAGE 437 | dBASE English/UK |
| dBASE ENU cp850 | DB850US0 | DOS CODE PAGE 850 | dBASE English/US |
| dBASE FRC cp863 | DB863CF1 | DOS CODE PAGE 863 | dBASE French Canadian |
| dBASE ENG cp850 | DB850UK0 | DOS CODE PAGE 850 | dBASE English850/UK |
| dBASE ITA cp850 | DB850IT1 | DOS CODE PAGE 850 | dBASE Italian850 |
| dBASE DEU cp850 | BD850DE0 | DOS CODE PAGE 850 | dBASE German850 |
| dBASE FRA cp850 | DB850FR0 | DOS CODE PAGE 850 | dBASE French850 |
| dBASE ITA cp437 | DB437IT0 | DOS CODE PAGE 437 | dBASE Italian |
| dBASE NLD cp850 | DB850NL0 | DOS CODE PAGE 850 | dBASE Dutch |
| dBASE FRC cp850 | DB850CF0 | DOS CODE PAGE 850 | dBASE French Canadian850 |
| dBASE DAN cp865 | DB865DA0 | DOS CODE PAGE 865 | dBASE Danish |
| dBASE DEU cp437 | DB437DE0 | DOS CODE PAGE 437 | dBASE German |
| 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' |

**Table D.4** Borland language drivers for SYBASE (continued)

| Long driver name | Short driver name | Character set | Collation sequence |
|---|---|---|---|
| Pdox ESP ANSI | ANSISPAN | ISO8859.1 (ANSI) | Paradox ESP437 |
| SQL Link ROMAN8 | BLROM800 | ROMAN8 | Binary |
| SYBASE SQL Dic437 | SYDC437 | DOS CODE PAGE 437 | SYBASE dict. with case-sensitivity |
| SYBASE SQL Dic850 | SYDC850 | DOS CODE PAGE850 | SYBASE dict. with case-sensitivity |

# Troubleshooting common connection problems

If you have problems establishing an SQL Server connection with SQL Link,

1 Confirm that the SQL Server is online.
   If not—Consult your database administrator.
   If so—Continue with step 2.

2 Verify that the network layer is functioning by trying to share files and print jobs to the spooler.
   If you are unsuccessful—Consult your database administrator.
   If you succeed—Continue with step 3.

3 Use your SYBASE or Microsoft tools to verify the connection at each layer.

   For SYBASE, use the DOS DBPING.EXE program or the Windows WDPING.EXE program to check the network connection. Use the DOS ISQL.EXE program to verify the server connection.

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

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

   ```
   select @@version
   ```

   If you are not connected—Consult your database administrator.
   If you are connected—Continue with step 4.

4 Check whether your BDE application SYBASE alias is set up properly.

   If you can connect directly from your workstation but not from within your BDE application, there is probably a problem with your IDAPI.CFG alias setup. Run the Configuration Utility and examine your SYBASE alias.

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 D.5 lists the general items that you might find helpful in working with SYBASE or Microsoft servers:

**Table D.5** General information about SQL Servers

| Item | Description |
|---|---|
| Product name | SYBASE |
| SQL dialect | SYBASE TRANSACT-SQL |
| 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 request to begin a transaction for multi-statement transaction processing? | Yes |
| Does the server require that you explicitly start a transaction for multi-statement transaction processing in pass-through SQL? | Yes |
| 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, a BDE application that copies or appends local data to an SQL Server table causes SQL Link to convert the local 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 local query against one or more remote tables. During the query, SQL Link converts any data originating in an SQL database to Paradox or dBASE format (depending on the type of answer required) before placing the data in the local answer table.

Tables D.6 through D.11 list SQL Server, Paradox, and dBASE data types and show how SQL Link translates between these data types.

**Table D.6** 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) | Bool |
| DateTime | DateTime | Date |
| Float | Number | Float {20.4} |
| Float4 | Number | Float {20.4} |
| Money4 | Money | Float {20.4} |
| DateTime4 | DateTime | Date |
| Image | Binary | Memo |
| Int | Number | Number {11.0} |

**Table D.6** SQL Server to Paradox and dBASE data type translations (continued)

| FROM: SQL Server | TO: Paradox | TO: dBASE |
|---|---|---|
| Money | Money | Number {20.4} |
| SmallDateTime | DateTime | Date |
| SmallFloat | Number | Number |
| SmallInt | Short | Number {6.0} |
| SmallMoney | Money | Number {20.4} |
| Text | Memo | Memo |
| TimeStamp | Binary | Memo |
| TinyInt | Short | Number {6.0} |
| VarBinary(n) | Binary | Memo |
| VarChar(n) | Alphanumeric(n) | Character(n) |

**Table D.7** Paradox to SQL Server and dBASE data type translations

| FROM: Paradox | TO: SQL Server | TO: dBASE |
|---|---|---|
| Alphanumeric(n) | varchar(n) | Character(n) |
| Number | float | Float {20.4} |
| Money | money | Float {20.4} |
| Date | datetime | Date |
| Short | smallint | Number {6.0} |
| Memo | text | Memo |
| Formatted memo | image | Memo |
| Binary | image | Memo |
| Graphic | image | Memo |
| OLE | image | Memo |
| Long | Int | Number {11.0} |
| Time | Character {>8} | Character {>8} |
| DateTime | DateTime | Character {>8} |
| Bool | Bit | Bool |
| AutoInc | Int | Number {11.0} |
| Bytes | Image | Bytes |
| BCD | N/A | N/A |

**Table D.8** 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 |
| Bool | bit | Bool |
| Lock | character {24.0} | Alphanumeric {24.0} |
| Bytes | image | Bytes |
| 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.

**Table D.9** Paradox physical to BDE logical and dBASE physical data type translations

| Paradox physical | BDE logical | dBASE physical |
|---|---|---|
| fldPDXCHAR | fldZSTRING | fldDBCHAR |
| fldPDXNUM | fldFLOAT | fldDBFLOAT {20.4} |
| fldPDXMONEY | fldFLOAT/fldstMONEY | fldDBFLOAT {20.4} |
| fldPDXDATE | fldDATE | fldDATE |
| fldPDXSHORT | fldINT16 | fldDBNUM {6.0} |
| fldPDXMEMO | fldBLOB/fldstMEMO | fldDBMEMO |
| fldPDXBINARYBLOB | fldBLOB/fldstBINARY | fldDBMEMO |
| fldPDXFMTMEMO | fldBLOB/fldstFMTMEMO | fldDBMEMO |
| fldPDXOLEBLOB | fldBLOB/fldstOLEOBJ | fldDBMEMO |
| fldPDXGRAPHIC | fldBLOB/fldstGRAPHIC | fldDBMEMO |
| fldPDXBLOB | fldPDXMEMO | fldDBMEMO |
| fldPDXLONG | fldINT32 | fldDBNUM {11.0} |
| fldPDXTIME | fldTIME | fldDBCHAR {>8} |
| fldPDXDATETIME | fldTIMESTAMP | fldDBCHAR {30} |
| fldPDXBOOL | fldBOOL | fldDBBOOL |
| fldPDXAUTOINC | fldINT32 | fldDBNUM {11.0} |
| fldPDXBYTES | fldBYTES | fldDBBYTES |
| fldPDXBCD | fldBCD | fldDBCHAR |

**Table D.10** dBASE physical to BDE logical and Paradox physical data type translations

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | |
| | fldINT16 | fldPDXSHORT |
| | else fldFLOAT | fldPDXNUM |
| fldDBMEMO | fldBLOB | fldPDXMEMO |
| fldDBBOOL | fldBOOL | fldPDXBOOL |
| fldDBDATE | fldDATE | fldPDXDATE |
| fldDBFLOAT | fldFLOAT | fldPDXNUM |
| fldDBLOCK | fldLOCKINFO | fldPDXCHAR {24} |
| fldDBBINARY | fldBLOB/fldstTYPEDBINARY | fldPDXBINARYBLOB |
| fldDBOLEBLOB | fldBLOB/fldstDBSOLEOBJ | fldPDXOLEBLOB |

**Table D.11** SQL Server physical to BDE logical, Paradox physical and dBASE physical data type translations

| SQL Server physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldSYBCHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldSYBVARCHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldSYBINT | fldINT32 | fldPDXLONG | fldDBNUM {11.0} |
| fldSYBSMALLINT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldSYBTINYINT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldSYBFLOAT | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldSYBMONEY | fldFLOAT/ fldstMONEY | fldPDXMONEY | fldDBFLOAT {20.4} |
| fldSYBTEXT | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldSYBBINARY | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldSYBVARBINARY | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldSYBIMAGE | fldBLOB/fldstBINARY | fldPDXBINARYBLOB | fldDBMEMO |
| fldSYBBIT | fldBOOL | fldPDXBOOL | fldDBBOOL |
| fldSYBDATETIME | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |
| fldSYBTIMESTAMP | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldSYBFLOAT4 | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldSYBMONEY4 | fldFLOAT/ fldstMONEY | fldPDXMONEY | fldDBFLOAT {20.4} |
| fldSYBDATETIME4 | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |

## 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.

Table D.12 lists SQL Server system tables you can access through SQL Link.

**Table D.12** 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 |
| sysprotects | Lists user permission information |
| sysusers | Lists each user and group for the database |

## SQL Server field-naming rules

Table D.13 lists field-naming rules for Paradox, dBASE, and SQL Server.

**Table D.13** SQL Server field-naming rules

| Naming rule | Paradox | dBASE | SQL Server |
|---|---|---|---|
| Max length (characters) | 25 | 10 | 30 |
| Valid characters[1] | All | All alphanumeric 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


## Symbols

*.IEM files 24

## Numerics

3+Open 55
3Com 3+Open TCP 35
3Com 3+Open TCP Version 2.0 35

## A

access rights 9
Add New Alias dialog box 18
Alias Manager 19
aliases
  adding new 18
  and LANGDRIVER setting 15
  creating and managing (general) 18
  customizing 19
  defined 13
  deleting 20
  modifying 17, 19
  setting up 18
  using with a local database 13
  using with an SQL databases 13
Aliases page
  *See* Alias Manager
ANSI-compliant databases (Informix only) 28
Asynchronous (dial-up) access protocol 46
AUTOEXEC.BAT file
  and Oracle connection 49

## B

Banyan VINES 47
BDE 1
  custom applications designed with 1
  default home directory 13
  IDAPI01.DLL 11
BDE API 1
BDE application
  defined 1
  user documentation 2
Beame & Whiteside TCP/IP 35
BLOB EDIT LOGGING 57
blob handles
  Informix 28
  InterBase 39
  Oracle 49
  SYBASE 60
blob reads
  Informix 28
  InterBase 39
  Oracle 49
  SYBASE 60
BLROM800.LD 24, 46, 56
Borland C++ 6
Borland Database Engine
  *See* BDE
Borland SQL Links for Windows Destination Drive dialog box 11
Borland SQL Links for Windows dialog box 11

## C

C.J. Date 6
case-sensitive searches 17
case-sensitivity for data
  Informix 27
  InterBase 39
  Oracle 49
  SYBASE 60
case-sensitivity for objects
  Informix 28
  InterBase 39
  Oracle 49
  SYBASE 60
character sets
  *See* language drivers
client
  applications 6
  defined 5
client workstation requirements
  *See* workstation requirements
Client/Server
  defined 5
COMMIT 6
COMMITED READ 31
common connection problems
  Informix 27
  InterBase 36
  Oracle 48
  SYBASE 59
configuration
  before you begin 13
  how the alias inherits 13
  leaving categories blank during 14
  sample IDAPI.CFG driver settings 14
  specifying default driver settings (general) 13
configuration file
  activating in WIN.INI 19
  default location 13
  saving 19
configuration parameters
  BLOB EDIT LOGGING 57
  CONNECT TIMEOUT 57
  DATABASE NAME 25, 57
  DATE MODE 25, 57
  DATE SEPARATOR 25
  DLL 14
  DRIVER FLAGS 14
  LANGDRIVER 14
  LOCK MODE 25
  MAX QUERY TIME 57
  NET PROTOCOL 46
  OPEN MODE 15
  SCHEMA CACHE SIZE 15
  SCHEMA CACHE TIME 15
  SERVER NAME 16
  SQLPASSTHRU MODE 16
  SQLQRYMODE 15, 17
  TIMEOUT 57
  TYPE 18
  USER NAME 18
  VENDOR INIT 47
  VERSION 18
Configuration Utility 13, 17
  Alias Manager (Aliases page) 19
  Driver Manager (Drivers page) 19
  icon 13
  IDAPI.CFG 13
  IDAPICFG.EXE 13
  Non-system Configuration File dialog box 19
CONNECT TIMEOUT 57
CONNECT utility 36
CONNECT.EXE 34, 36
CURSOR STABILITY 31
customizing aliases 19

## D

data type translation

dBASE physical to BDE logical 30, 41, 52, 62
dBASE to InterBase and Paradox 40
dBASE to Oracle and Paradox 51
dBASE to Paradox and Informix 29
dBASE to SYBASE and Paradox 61
Informix physical to BDE logical 30
Informix to Paradox and dBASE 28
InterBase physical to BDE logical 42
InterBase to Paradox and dBASE 39
Oracle physical to BDE logical 52
Oracle to Paradox and dBASE 50
Paradox physical to BDE logical 30, 41, 51, 62
Paradox to dBASE and Informix 29
Paradox to InterBase and dBASE 40
Paradox to Oracle and dBASE 50
Paradox to SYBASE and dBASE 61
SQL to InterBase 42
SYBASE physical to BDE logical 63
SYBASE to Paradox and dBASE 60
database
local (defined) 6
maintenance in a Client/Server environment 6
database access requirements 10, 24
DATABASE NAME 25, 57
database server
*See* server
databases
ANSI-compliant (Informix only) 28
logging (Informix only) 28
non-logging (Informix only) 28
DATE MODE 25, 57
DATE SEPARATOR 25
dBASE 1, 6
field-naming rules 32
dBASE data type translation
to BDE logical 30, 41, 52, 62
to InterBase and Paradox 40
to Oracle and Paradox 51
to Paradox and Informix 29
to SYBASE and Paradox 61
dBASE for Windows
as a local database 6
dBASE language drivers
for Informix 26
for InterBase 38
for Oracle 47
for SYBASE 58
DBDATE (Informix only) 25
DBNMP3.DLL 56
DBPING.EXE (Sybase SQL Server only) 59
DECnet protocol 46
deleting an alias 20
Delphi 1, 6
dialog boxes
Add New Alias 18
Borland SQL Links for Windows 11
Borland SQL Links for Windows Destination Drive 11
IDAPI Location Settings 11
InterBase Connect Utility 36
Non-system Configuration File 19
Digital PATHWORKS 35
DIRTY READ 31
DLL 14
DRIVER FLAGS 14
Driver Manager 19
Drivers page
*See* Driver Manager

## F

field-naming rules 32
for Informix 32
for InterBase 43
Oracle 53
SYBASE 64
Files
SAF.EXE 59
FTP PC/TCP 35

## G

gateways 5
GDS.DLL 34

## H

hardware requirements
workstation 9
HOSTS file (InterBase only) 37
How SQL Links Works (figure) 7
HP ARPA Service for DOS 35

## I

IBM 3270 protocol 46
IBM APPC LU 6.2 protocol 46
IDAPI 13
*See also* BDE
defined 1
IDAPI Configuration Utility 8
*See also* Configuration Utility
IDAPI Location Settings dialog box 11
IDAPI.CFG 13
sample driver settings 14
IDAPI01.DLL 11
IDAPICFG.EXE 13
implicit row IDs
Informix 28
InterBase 39
Oracle 49
SYBASE 60
inet daemon (InterBase only) 37
Informix
common connection problems 27
field-naming rules 32
installation changes 24
isql utility 27
language drivers for 25
server requirements 23
special SQL Link driver settings for 24
transaction isolation levels 31
UPDATE STATISTICS statement 27
workstation requirements 23
INFORMIX alias setup 27
INFORMIX alias type 24
Informix data type translation
to BDE logical 30
to Paradox and dBASE 28
using the Interval data type 31
INFORMIX driver type 24
Informix message files 24
Informix servers
DBDATE variable 25
working with (general) 27
INFORMIX-NET/PC 23
INFORMIX-SQL server requirements 23
INSTALL program 10
installation

SQL Links
*See* software installation
installation changes
for Informix 24
for InterBase 34
for Oracle 46
for SYBASE 56
INTERBAS.MSG 34
InterBase
checking TCP libraries 37
common connection problems 36
Connection Utility 34, 36
field-naming rules 43
HOSTS file 37
inet daemon 37
installation changes 34
language drivers for 38
security database (isc.gdb) 38
server requirements 33
server specification 34
SERVICES file 37
setting SQLQRYMODE 38
system relations 42
using PING 37
using TELNET 37
workstation requirements 33
InterBase data type translation
to BDE logical 42
to Paradox and dBASE 39
to standard SQL 42
InterBase message files 34
InterBase server requirements 33
InterBase servers
working with (general) 39
InterBase-supplied DLLs 34
Interval data type (Informix) 31
INTRBASE alias setup 38
INTRBASE alias type 34
INTRBASE driver type 34
*isc.gdb* (InterBase only) 38
isql utility (Informix only) 27

## L

LAN 5
LAN Manager 35, 46, 55
LANGDRIVER 14
and local query processing 15
choosing 15
impact on SQLQRYMODE 15
language drivers
effect on SQL query processing 17
for Informix 25
for InterBase 38
for Oracle 47
for SYBASE 58
Oracle-supplied 48
sort order 17
SYBASE-supplied 59
uppercasing 17
LDLLSQLW.DLL 24
Local Area Network 5
local database
using an alias with 13
local databases
defined 6
local query processing 8
local SQL
defined 7
LOCK MODE 25
settings 25
logging database (Informix only) 28
long driver names
*See* language drivers

## M

MAX QUERY TIME 57
message files
Informix 24
InterBase 34
Microsoft Named Pipes 35
modifying an alias 19

## N

Named Pipes 35, 55
Named Pipes protocol 46
NET PROTOCOL 46
settings 46
NetBios protocol 46
NetWare Requestor 55
network installations
workstation requirements 9
network protocol software
workstation requirements 9
network server
*See* server
non-logging database (Informix only) 28
Novell LAN WorkPlace for DOS 35
Novell SPX/IPX 35

## O

OPEN MODE 15
effect on pass-through SQL 15
operating system requirements
workstation 9
ORA6WIN.DLL 46
and VENDOR INIT 47
ORA7WIN.DLL 46
and VENDOR INIT 47
Oracle
common connection problems 48
field-naming rules 53
installation changes 46
language drivers for 47
orainst program 48
special SQL Link driver settings for 46
SQL*DBA 48
SQL*Net driver 45
SQL*Plus 48
system tables 52
workstation requirements 45
ORACLE alias setup 49
ORACLE alias type 46
Oracle data type translation
to BDE logical 52
to Paradox and dBASE 50
ORACLE driver type 46
Oracle server requirements 45
Oracle servers
working with (general) 49
orainst program (Oracle only) 48
ORAWE850.LD 46

## P

Paradox 1, 6
field-naming rules 32
Paradox data type translation
to BDE logical 30, 41, 51, 62
to dBASE and Informix 29
to InterBase and dBASE 40
to Oracle and dBASE 50
to SYBASE and dBASE 61
Paradox for Windows
as a local database 6
Paradox language drivers
for Informix 26
for InterBase 38
for Oracle 48
for SYBASE 58
pass-through SQL
and SQLPASSTHRU MODE 16
PING 37

## Q

queries
blocking local processing of 8

heterogenous 8
improving retrieval time (Informix only) 27
in a database server environment 5
multiple-statement 8
processing 5
processing locally 8
processing on the SQL server 8
processing SQL queries locally 8
types always processed locally 17
query processing
and SQLQRYMODE 17

## R

RDBMS
defined 6
READLINK.TXT 10, 12, 24, 34, 46, 56
README file
*See* READLINK.TXT
REMOTE.DLL 34
REPEATABLE READ 31
ROLLBACK 6

## S

SAF.EXE (Microsoft SQL Server only) 59
sample IDAPI.CFG driver settings 14
SCHEMA CACHE SIZE 15
SCHEMA CACHE TIME 15
searches
case-sensitive 17
SEQUEL 6
server
accessing network server data 5
database 5
defined 5
in a network environment 5
querying a database server 5
workstation as a client of 5
SERVER NAME 16
format for InterBase 16
server requirements
Informix 23
InterBase 33
Oracle 45
SQL Server 55
SERVICES file 34
SERVICES file (InterBase only) 37
setting up aliases 18
short driver name
*See* language drivers
short driver names
*See* language drivers
software installation
and IDAPI01.DLL 11
before you begin 10
information requested during 12
INSTALL program 10
opening dialog box 11
specifying the home directory 11
specifying where to install IDAPI.CFG 11
what happens during 10
software prerequisites
workstation 9
special SQL Link driver settings
for Informix 24
for Oracle 46
for SYBASE 57
SPX/IPX 35
SPX/IPX protocol 46, 55
SQL
defined 6
pass-through
*See* pass-through SQL
recommended reading 6
tables 15
uses of 6
SQL aliases
*See* aliases
SQL database
accessing with SQL Links 6
using an alias with 13
SQL databases
access requirements 10
SQL Link drivers
for SYBASE 59
SQL Links
advantages for SQL users 8
advantages for workstation users 7
description 6
how they work 7
installation 10
options for querying SQL data 8
options for SQL data access 6
product description 1
query processing in 17
READLINK.TXT file 10
specifying home directory 11
supported Borland applications 1
supported databases 1
using a BDE application with 6
where to find information 2
writing custom applications for 6
SQL Links language drivers
for Informix 26
for Oracle 48
SQL Links users
BDE application developer 2
BDE application end-user 2
SQL Server
*See* SYBASE
SQL Server requirements 55
SQL Servers
working with (general) 60
SQL servers (general)
documentation 2
SQL statements
COMMIT 6
ROLLBACK 6
SQL*DBA 48
SQL*Plus 48
SQL.EXE (Sybase SQL Server only) 59
SQL13WIN.DLL 46
SQLD_IB.DLL 34, 39
SQLD_IB.HLP 34
SQLD_INF.DLL 24, 27
SQLD_INF.HLP 24
SQLD_ORA.DLL 46, 49
SQLD_ORA.HLP 46
SQLD_SS.DLL 56, 60
SQLD_SS.HLP 56
SQLPASSTHRU MODE 16
and transaction control statements 16
settings 16
support of pass-through SQL in 16
SQLQRYMODE 15, 17
blocking local query processing with 17
errors when set to SERVER 17
InterBase considerations 38
settings 17
SQL-style transactions 7
SQLWIN.DLL 46
Sun PC NFS 35
Sun PC NFS v3.5 35
Sun PC NFS v4.0 35
SYBASE

common connection problems 59
DBPING.EXE (Sybase SQL Server) 59
field-naming rules 64
installation changes 56
language drivers for 58
SAF.EXE (Microsoft SQL Server) 59
special SQL Link driver settings for 57
SQL.EXE (Sybase SQL Server) 59
system tables 63
WDPING.EXE (Sybase SQL Server) 59
workstation requirements 56
SYBASE alias setup 59
SYBASE alias type 56
SYBASE data type translation
to BDE logical 63
to Paradnx and dBASE 60
SYBASE driver type 56
SYBASE Net libraries 55
SYBASE TRANSACT-SQL 60
SYDC437.LD 56
SYDC850.LD 56
system relations (InterBase only) 42
system tables
for Oracle 52
for SYBASE 63

## T

TCP libraries (InterBase only) 37
TELNET 37
Terms and conventions 3
text files
READLINK.TXT 10, 12
TIMEOUT 57
transaction isolation levels
committed read 31
cursor stability 31
dirty read 31
repeatable read 31
transaction processing
Informix 28
InterBase 39
Oracle 49
SYBASE 60
transactions
defined 6
ending 6
isolation levels in (Informix only) 31
SQL-style 7
TYPE 18

## U

Ungermann-Bass Net/One 35
USER NAME 18
User's Guide
terms and conventions 3

## V

VENDOR INIT 47
VERSION 18

## W

W3DBLIB.DLL 56
WAN 5
WDPING.EXE (Sybase SQL Server only) 59
Where to find information 2
Wide Area Network 5
WIN.INI 19
Wollongong Pathway Access for DOS 35
Wollongong WIN TCPIP for DOS 35
workstation requirements
for Informix 23
for InterBase 33
for Oracle 45
for SYBASE 56
hardware 9
HOSTS file (InterBase only) 34
network access 9
network protocol software 9
network protocol software (Oracle) 45
operating system 9
SERVICES file (InterBase only) 34

# User's Guide

Version 2.5

# Borland®
# SQL Links for Windows

Borland International, Inc., 100 Borland Way
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 © 1995 by Borland International. All rights reserved. All Borland products are trademarks or registered trademarks of Borland International, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders.

Portions of this program copyright © 1983–1990 by Novell, Inc. All rights reserved

Printed in the U.S.A.

2E0R195
9596979899- 987654321
W1

# Contents


**Introduction** 1

Supported Borland applications . . . . . 1
Supported databases . . . . . 1
Where to find information. . . . . 1
Terms and conventions . . . . . 3

Chapter 1
**How Borland SQL Links work** 5

Introduction . . . . . 5
The database server . . . . . 5
SQL . . . . . 6
Borland SQL Links for Windows. . . . . 6
How the SQL Links work . . . . . 7
Options for querying SQL data. . . . . 8
Default SQL query mode . . . . . 8
Default pass-through SQL mode . . . . . 8

Chapter 2
**Installing the SQL Link drivers** 9

Client workstation requirements . . . . . 9
Database access requirements . . . . . 10
Installing the software . . . . . 10
What happens during installation? . . . . . 10
Installation procedure . . . . . 10

Chapter 3
**Configuring the SQL environment** 13

Specifying default SQL Link driver settings . . 13
DLL . . . . . 14
DRIVER FLAGS . . . . . 14
LANGDRIVER . . . . . 14
OPEN MODE . . . . . 15
SCHEMA CACHE SIZE. . . . . 15
SCHEMA CACHE TIME . . . . . 15
SERVER NAME. . . . . 16
SQLPASSTHRU MODE . . . . . 16
SQLQRYMODE. . . . . 17
How language drivers affect SQL server query processing . . . . . 17
Blocking local processing of queries. . . . . 17
TYPE . . . . . 18
USER NAME . . . . . 18
VERSION . . . . . 18
Creating and managing SQL aliases. . . . . 18
Adding a new SQL alias. . . . . 18
Modifying an existing SQL alias . . . . . 19
Deleting an SQL alias . . . . . 20
Connecting to the SQL server . . . . . 20

Appendix A
**Connecting to Informix** 23

INFORMIX-SQL server requirements . . . . . 23
Client workstation requirements . . . . . 23
Database access requirements. . . . . 24
Installation changes. . . . . 24
Special SQL Link driver settings . . . . . 24
DATABASE NAME . . . . . 25
DATE MODE. . . . . 25
DATE SEPARATOR . . . . . 25
LOCK MODE. . . . . 25
Borland language drivers for Informix . . . . . 25
Troubleshooting common connection problems . . . . . 27
Working with Informix servers . . . . . 27
Informix data type translations . . . . . 28
Using the Informix Interval data type. . . . . 31
Transaction isolation levels. . . . . 31
Informix field-naming rules . . . . . 32

Appendix B
**Connecting to InterBase** 33

InterBase server requirements. . . . . 33
Client workstation requirements . . . . . 33
Installation changes. . . . . 34
TCP/IP interface . . . . . 35
Other communication protocols . . . . . 35
Testing the InterBase connection . . . . . 36
Troubleshooting common connection problems . . . . . 36
Borland language drivers for InterBase . . . . . 38
Working with InterBase servers. . . . . 39
InterBase data type translations . . . . . 39
InterBase equivalents to standard SQL data types . . . . . 42
InterBase system relations/tables . . . . . 42
InterBase field-naming rules . . . . . 43

Appendix C
**Connecting to ORACLE** 45

ORACLE server requirements . . . . . 45
Client workstation requirements . . . . . 45
Installation changes. . . . . 46
Special SQL Link driver settings . . . . . 46
NET PROTOCOL . . . . . 46

VENDOR INIT . . . . . 47
Borland language drivers for ORACLE . . . . . 47
Troubleshooting common connection problems . . . . . 48
Working with ORACLE servers . . . . . 49
ORACLE data type translations . . . . . 49
ORACLE system tables . . . . . 52
ORACLE field-naming rules . . . . . 53

Appendix D
**Connecting to SYBASE and Microsoft SQL Servers 55**

SQL Server requirements . . . . . 55
Client workstation requirements . . . . . 56
Installation changes . . . . . 56
Special SQL Link driver settings . . . . . 57
Blob EDIT LOGGING . . . . . 57
CONNECT TIMEOUT . . . . . 57
DATABASE NAME . . . . . 57
DATE MODE . . . . . 57
TIMEOUT . . . . . 57
MAX QUERY TIME . . . . . 57
Borland language drivers for SQL Server . . . . 58
Troubleshooting common connection problems 59
Working with SQL Servers . . . . . 59
SQL Server data type translations . . . . . 60
SQL Server system tables . . . . . 63
SQL Server field-naming rules . . . . . 64

**Index 65**

# Tables


1 Where to find information on SQL Links and related topics . . . . . 2
2 SQL Link documentation conventions . . . . . 3
2.1 Client workstation requirements . . . . . 9
3.1 SCHEMA CACHE TIME settings . . . . . 15
3.2 SQLPASSTHRU MODE settings . . . . . 16
3.3 SQLQRYMODE settings . . . . . 17
A.1 Server software requirements . . . . . 23
A.2 Client workstation requirements . . . . . 23
A.3 Installation changes for the INFORMIX SQL Link driver . . . . . 24
A.4 LOCK MODE settings . . . . . 25
A.5 Borland language drivers for Informix . . . . 26
A.6 General information about Informix servers . 27
A.7 Informix to Paradox and dBASE data type translations . . . . . 28
A.8 Paradox to dBASE and Informix data type translations . . . . . 29
A.9 dBASE to Paradox and Informix data type translations . . . . . 29
A.10 Paradox physical to BDE logical and dBASE physical data type translations . . . . . 30
A.11 dBASE physical to BDE logical and Paradox physical data type translations . . . . . 30
A.12 Informix physical to BDE logical and Paradox and dBASE physical data type translations . . . . . 30
A.13 Default isolation levels in Informix and SQL Link . . . . . 32
A.14 Informix field-naming rules . . . . . 32
B.1 Server software requirements . . . . . 33
B.2 Client workstation . . . . . 33
B.3 Installation changes for the InterBase SQL Link driver . . . . . 34
B.4 Winsock 1.1 Client files . . . . . 35
B.5 Non-Winsock Compliant TCP Support Files . 35
B.6 Borland language drivers for InterBase . . . . 38
B.7 General information about InterBase servers . . . . . 39
B.8 InterBase to Paradox and dBASE data type translations . . . . . 39
B.9 Paradox to InterBase and dBASE data type translations . . . . . 40
B.10 dBASE to InterBase and Paradox data type translations . . . . . 40
B.11 Paradox to BDE logical to dBASE data type translations . . . . . 41
B.12 dBASE to BDE logical to Paradox data type translations . . . . . 41
B.13 InterBase to BDE logical to Paradox and dBASE data type translations . . . . . 42
B.14 SQL to InterBase data type translations . . . . 42
B.15 Selected InterBase system relations . . . . . 43
B.16 InterBase field-naming rules . . . . . 43
C.1 Server software requirements for ORACLE . . 45
C.2 Client workstation requirements . . . . . 45
C.3 Installation changes for the ORACLE SQL Link driver . . . . . 46
C.4 NET PROTOCOL settings . . . . . 46
C.5 Borland language drivers for ORACLE . . . . 47
C.6 General information about ORACLE servers . . . . . 49
C.7 ORACLE to Paradox and dBASE data type translations . . . . . 50
C.8 Paradox to ORACLE and dBASE data type translations . . . . . 50
C.9 dBASE to ORACLE and Paradox data type translations . . . . . 51
C.10 Paradox to BDE logical and dBASE physical data type translations . . . . . 51
C.11 dBASE physical to BDE logical and Paradox physical type translations . . . . . 52
C.12 ORACLE physical to BDE logical and Paradox and dBASE physical data type translations . . . . . 52
C.13 Selected ORACLE system tables . . . . . 52
C.14 ORACLE field-naming rules . . . . . 53
D.1 Server software requirements for SYBASE . . 55
D.2 Client workstation requirements . . . . . 56
D.3 Installation changes for the SYBASE SQL Link driver . . . . . 56
D.4 Borland language drivers for SYBASE . . . . 58
D.5 General information about SQL Servers . . . . 60
D.6 SQL Server to Paradox and dBASE data type translations . . . . . 60
D.7 Paradox to SQL Server and dBASE data type translations . . . . . 61
D.8 dBASE to SQL Server and Paradox data type translations . . . . . 61
D.9 Paradox physical to BDE logical and dBASE physical data type translations . . . . . 62
D.10 dBASE physical to BDE logical and Paradox physical data type translations . . . . . 62
D.11 SQL Server physical to BDE logical, Paradox physical and dBASE physical data type translations . . . . . 63
D.12 Selected SQL server system tables . . . . . 63
D.13 SQL Server field-naming rules . . . . . 64

# Figures

1.1 How SQL Links works. . . . . .7
2.1 Opening dialog box . . . . . 11
2.2 Specifying the Borland SQL Links home directory . . . . . 11
2.3 Specifying where to install IDAPI.CFG . . . . 11
3.1 Sample IDAPI.CFG driver settings . . . . . 14
3.2 Sample Add New Alias dialog box . . . . . 18
3.3 Customizing the new alias . . . . . 19
B.1 InterBase Connect Utility dialog box . . . . . 36