## Introduction to Borland SQL Links for Windows (32-bit version)

Borland SQL Links for Windows is a set of BDE-hosted driver connections to database servers. By creating queries, SQL Links emulates full navigation capabilities, enabling users to access and manipulate data in SQL databases by using convenient features in Borland applications.

Any Borland application based upon the Borland Database Engine (BDE) can use Borland SQL Links, including your custom applications built using the BDE API. Borland SQL Links is included with some Borland client/server application development tools such as Delphi Client/Server.

The SQL Links product package includes database drivers for InterBase, DB2, Informix, ORACLE, Sybase, and Microsoft SQL Server databases. These drivers require the installation of the appropriate vendor's server connectivity API or interface software. Borland Database Engine includes built-in drivers for Borland standard databases.

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

{button ,JI("","additionalinformation")} Additional information

{button ,AL("bdedocs")} Other BDE online documentation

**Where to find additional information**

This Online User's Guide describes how Borland SQL Links for Windows works, how to install one or more SQL Links drivers, and how to set up your supported Borland application to access SQL data. See Configuring and connecting to a SQL Server for driver-specific information on required workstation software, configuration parameters, troubleshooting, and data translation.

Depending on how you plan to work with SQL data, you might need the following documentation:

- 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 database server connectivity software documentation
- your SQL database server documentation
- the Borland SQL Links README file, if any (READLINK.TXT)
- the BDE Online Reference (BDE32.HLP) and its Contents file (BDE32.CNT)
- the BDE Configuration Utility Online Guide and Help (BDECFG32.HLP) and its Contents file (BDECFG32.CNT)

**Vendor's server connectivity API**

| Vendor | API/Connectivity software |
|---|---|
| Borland InterBase | PC Client |
| DB2 | IBM CAE version 2.1.1 for Windows 95 and NT |
| Informix | Informix ESQL/C 7.2 for Windows 95 and NT |
| Microsoft | DB-Lib 6.x |
| ODBC Socket | ODBC vendors' drivers |
| Oracle | OCI 7.x |
| Sybase | DB-Lib for Sybase SQL Server version 4.x |
| Sybase | CT-Lib for Sybase Open Client version 10 and later |

**Note:** If a vendor DLL is not found in the vendor's directory, check the WINDOWS\SYSTEM directory.

## The database server

A local area network (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.

A database server is a network server 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 created with Paradox for Windows, Visual dBASE, 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.

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

## SQL

SQL is a descendant of SEQUEL (or Structured English QUEry Language), which was originally designed at IBM. 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.

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

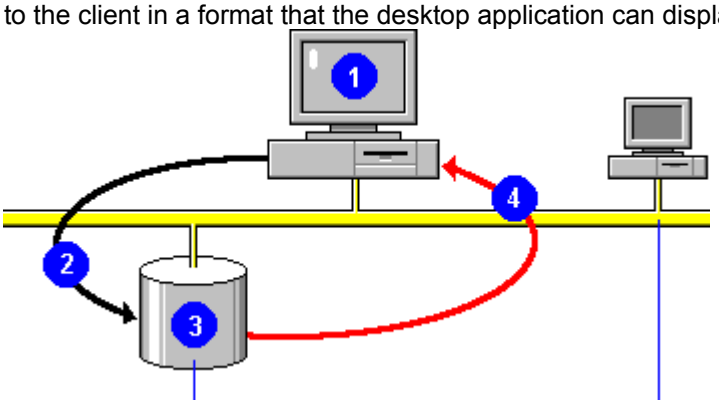
## How SQL Links works

Borland SQL Links for Windows is designed for BDE application users who need to access 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 Links driver and create a 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 on local data. For information on how to use local SQL in your BDE application, see your application's user documentation.

With a SQL Links 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. This picture shows how it works.




1. BDE client queries a SQL database.
2. SQL Links driver, using the workstation's client software, connects to the SQL server and sends the query in that server's SQL dialect.
3. SQL server performs error and syntax checking on the SQL statements, processes the query, and returns the answer to the BDE client.
4. SQL Links driver translates the answer from SQL to client syntax. The client then formats and displays the data to the user.

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

## SQL Links features

Using your BDE application with a SQL Links 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 a SQL server through a SQL Links driver and your BDE application also lends traditional SQL database users some of the advantages inherent in workstation databases, enabling you to

- open a table and clone a cursor
- move in both forward and backward directions through a result set or answer table
- order data using an available index
- set a "bookmark" 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

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

## Requirements

This section lists software that should already be installed and running at the client workstation. It also lists related files and parameters and database access requirements.

| Category | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| System requirements | 1.5 MB of free disk space and operating system that meets the requirements of your BDE application. |
| Access rights | 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 installation directory. This directory is modified during SQL Links installation. (For applications installed on the network server only.) |
| Network protocol software | Network protocol software compatible with both the server network protocol and the client workstation database communication driver. |

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

**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.

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

## Installing the software

**Before you begin:** Be sure your workstation already meets the client workstation requirements for installing Borland SQL Links for Windows. See Requirements

Also, be sure you have already installed your BDE application.

**What happens during installation?**

During installation, the SQL Links SETUP program always:

- Installs a SQL Links driver that enables your application to access your SQL database.
- Adds new options to the BDE configuration file so you can configure the driver default parameters and create an alias to a SQL database. (See the appropriate driver page in the BDE Configuration Utility, BDECFG32.EXE)
- Installs an online Help file you can use when configuring the driver.

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 section.

---

{button ,AL("intro")} Other topics in this Introduction to SQL Links

## Configuring the SQL environment: Overview

Installed with your BDE application is the BDE Configuration Utility (BDECFG32.EXE), which enables users to modify their application configuration. The configuration parameters are stored in the BDE configuration file IDAPI.CFG and in the Windows Registry:

```
HKEY_LOCAL_MACHINE|SOFTWARE|BORLAND|DATABASE ENGINE|SETTINGS
```

This section describes how to use the BDE 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 instructions on how to use the BDE Configuration Utility, see the *BDE Configuration Utility Online Help.*

---

{button ,AL("config")} Other SQL Links configuration topics

{button ,AL("bdedocs")} Other BDE online documentation

## Before you begin

Be sure you have already installed the SQL Links software.

1. Exit all Borland applications and other applications that use BDE.
2. Open your application program group.
3. Click the BDE Configuration Utility icon. The Configuration Utility window appears, open to the Drivers page.

---

{button ,AL("config")} Other SQL Links configuration topics

## Specifying default SQL Links 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 a SQL database.

The SQL Links 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 Drivers page displays all the configuration parameters for that driver.
2. If necessary, use the scroll bars at the side of the Parameters list to examine the complete list of driver configuration parameters.
3. If desired, edit the driver's default configuration. If you leave any categories blank, the BDE Configuration Utility assumes you want to use the driver default.
4. When you are finished, click File|Save.

Your changes take effect the next time BDE is restarted.

---

{button ,AL("config")} Other SQL Links configuration topics

## Common SQL Links driver settings

SQL driver settings determine the way the Borland Database Engine connects to a target SQL server, and how SQL databases are opened. Borland SQL Links drivers must be installed before configuring.

You configure all settings for each SQL Links driver by using the BDE configuration utility, Drivers page. For each selected driver a list of parameter settings is displayed in the right pane. Each driver includes a subset of configuration parameters common to all SQL Links drivers, along with a few special settings specific to that driver.

### Common SQL Links driver configuration parameters

The following configuration parameters are common to all SQL Links drivers:

- VERSION
- TYPE
- DLL
- DLL32
- DRIVER FLAGS
- TRACE MODE
- SERVER NAME
- USER NAME
- OPEN MODE
- SCHEMA CACHE SIZE
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- SCHEMA CACHE TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- ENABLE BCD

### Special SQL Links driver configuration parameters

For information on configuration parameters that are unique to a specific SQL driver, see the driver-specific sections listed below. If you have installed the specific driver, these special parameters appear along with the common parameters when the specific driver is selected in the Drivers page of the BDE Configuration Utility:

- Special SQL Links driver settings for Informix
- Special SQL Links driver settings for ORACLE
- Special SQL Links driver settings for Sybase
- Special SQL Links driver settings for Microsoft SQL Server
- Special SQL Links driver settings for DB2 Server
- Connecting to InterBase (No special settings)

Alternatively, in the BDE Configuration Utility, you can highlight the specific driver name and click the Help button to display the same topics available here.

{button ,AL("config")} Other SQL Links configuration topics

**VERSION**
Internal Borland version number of the SQL Links driver. Do not modify.

**TYPE**

Type of server to which this driver helps you connect. Do not modify.

SERVER represents a SQL server.

FILE represents a standard driver, file-based server, such as local Paradox and dBASE tables.

**DLL**

SQL Links dynamic link library name for the selected 16-bit driver.

In the case of Informix or Sybase drivers, you might need to specify a particular DLL to support a specific server interface.

For internal use only. Do not modify.

**DLL32**

SQL Links dynamic link library name for the selected 32-bit driver.

In the case of Informix or Sybase drivers, you might need to specify a particular DLL to support a specific server interface.

For internal use only. Do not modify.

**DRIVER FLAGS**
Internal product-specific flag. Do not modify unless directed to do so by Borland support personnel.

**Setting Interbase driver flags**
Specifying a Read Committed transaction isolation level can result in incompatibility with applications using earlier versions of SQL Links for InterBase.

Earlier versions of SQL Links supported Repeatable Read only. To maintain transaction support compatibility with earlier versions of SQL Links, use the BDE Configuration Utility to set the DRIVER FLAGS option to 512.

**Default:**
NULL.

**TRACE MODE**

A numeric value (bit mask) specifying how much trace information to log. The Windows OutputDebugString call is used to output the requested information to the debug window. The following table shows which information is logged based on bit settings:

| Bit Settings | Logged Information |
|---|---|
| 0x0001 | prepared query statement |
| 0x0002 | executed query statements |
| 0x0004 | vendor errors |
| 0x0008 | statement options (that is: allocate, free) |
| 0x0010 | connect / disconnect |
| 0x0020 | transaction |
| 0x0040 | BLOB I/O |
| 0x0080 | miscellaneous |
| 0x0100 | vendor calls |

**SERVER NAME**

Name of the target SQL server. Use format appropriate for your server. When specified for InterBase servers, includes the full path to the database file, that is:

```
servername:/usr/gds/directoryname/databasename.gdb
```

**USER NAME**
Default user name for accessing the target SQL server.

**OPEN MODE**

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

**Note:** Setting OPEN MODE to READ ONLY affects operations from the application's user interface, but has no effect on passthrough SQL.

**Default:**
READ/WRITE

**SCHEMA CACHE SIZE**

Number of SQL tables whose schema information will be cached.

**Possible values:**

0 – 32

**Default:**

8

## 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. 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, SQL Links provides 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.

**Extended character sets**

If your SQL database uses extended character sets, make sure the alias you use to access the SQL server specifies the correct SQL Links 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 Links language driver, you may want to modify your alias SQLQRYMODE entry to prevent processing of queries under local database rules.

**Default**

If no language driver is specified, the data will be processed in codepoint order, and in the DOS character set.

## SQLQRYMODE

SQLQRYMODE specifies the mode for handling queries to SQL data. Possible modes and their meanings are listed here.

| Setting | Mode | Meaning |
|---|---|---|
| NULL | Server-local | 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. |

**Default:** NULL (blank setting)

For an explanation of SQL query modes, including passthrough, see Default and passthrough SQL query modes

**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 a 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 a 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 Links driver to block local processing of queries.

To do so, use the BDE 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 BDE is restarted.

**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 database)
- Queries that cannot be expressed as a single SQL statement
- Queries that the SQL server does not support

## SQLPASSTHRU MODE

Specifies whether or not the application will be able to access the SQL server via both desktop commands and passthrough SQL in the same alias connection. Possible modes and their meanings are listed in this table.

| Setting | Meaning |
|---|---|
| SHARED AUTOCOMMIT | Passthrough SQL and non-passthrough SQL share the same connection, and passthrough SQL behaves in a similar fashion to non-passthrough. This means that, as long as the user is not in an explicit client transaction or batch mode, passthrough SQL statements are automatically committed. |
| SHARED NOAUTOCOMMIT | Passthrough SQL and non-passthrough SQL share the same connection, but the SQL driver does not automatically commit passthrough statements. In this mode, passthrough behavior is server-dependent. |
| NOT SHARED | Passthrough SQL and non-passthrough SQL do NOT share the same connection. Updateable SQL queries are not supported with aliases that have the SQLPASSTHRU MODE parameter set to NOT SHARED. (Default) |

**Default:**
The default value is NOT SHARED (or blank setting) for all SQL Links drivers.

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

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

For a discussion of how the application processes passthrough SQL queries, and information about using passthough SQL with your Borland application, see your application product documentation.

For an explanation of SQL query modes, including passthrough, see Default and passthrough SQL query modes.

**SCHEMA CACHE TIME**

Specifies how long table list information will be cached. (In BDE table information is cached when you call either DbiOpenTableList or DbiOpenFileList.)

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

Possible modes and their meanings are listed here.

| Setting | Meaning |
|---|---|
| -1 | The table list is cached until you close the database. (Default) |
| 0 | No table lists are cached. |
| 1 through 2147483647 | The table list is cached for the number of seconds specified in the setting. |

**Default:**

-1

## MAX ROWS

Specifies maximum number of rows that the SQL driver will attempt to fetch for *every* SQL statement sent to the server. This includes schema inquiry queries that the driver sends to the server during a table open to retrieve column, index, and valcheck information.

If a request is made for more than MAX ROWS, then an error is returned (DBIERR_ROWFETCHLIMIT). A return of DBIERR_ROWFETCHLIMIT is similar to DBIERR_EOF, except that it indicates a client-forced EOF when there actually may be more rows available on the server.

You can use the MAX ROWS option as a system governor to prevent users from unintentionally tying up valuable system resources. For example, a database administrator may set up users' configuration files to prevent them from tying up server and network resources if they happen to do a "SELECT * ..." on a huge table. You can set the MAX ROWS option make it impossible for a user to generate a million record query by mistake.

Be aware that if you set MAX ROWS too small, you may not be able to open a table under that database because it cannot get sufficient schema information. If MAX ROWS is not set to a large enough value to retrieve all the required metadata information during table open, then an error is returned and the table cannot be opened. MAX ROWS does not affect non-updateable queries in this way because a DESCRIBE (instead of a schema query) is used to get query column information.

**Default:**
-1 (No limit on fetching rows.)

**WARNING!**

A MAX ROW limitation could break existing BDE applications that fetch until receiving DBIERR_EOF. Such applications must be modified to handle a return of DBIERR_ROWFETCHLIMIT as well as DBIERR_EOF. Users should be able to "see" all rows that have already been fetched, but they should be notified that there may be additional rows on the server.

**BATCH COUNT**

Specifies the number of modified records to be included in a batch before auto-committing. In this way you can adjust the size of a batch to accommodate server transaction logs that are not big enough to handle the whole batch.

In BDE you can override this value by setting the database property, dbBATCHCOUNT. See Borland Database Engine Online Reference.

**Default:**

The number of records that can fit into 32 KB

**ENABLE SCHEMA CACHE**

Specifies whether the BDE caches table schema locally for tables residing on SQL servers. This enhances performance for table opens. Set SCHEMA CACHE DIR to the directory in which the local cache is stored.

The schema cache is designed for static databases. Changes to the tables within the database can cause errors. If you frequently add columns to a table, add or drop indexes from a table, or change validity checks on fields (such as adding or removing "not null"), set ENABLE SCHEMA CACHE to FALSE.

Another good sign that a database is not static enough to use schema cache is if "Unknown SQL Errors" are returned whose error context refers to

"Unknown Column"

"Invalid Bind Type"

"Invalid Type"

"Invalid Type Conversion"

"Column Not a Blob"

**Default:**
FALSE (No local schema caching)

**SCHEMA CACHE DIR**

Specifies the directory in which the local schema cache is stored. Used when ENABLE SCHEMA CACHE is TRUE to cache schema locally when tables reside on a SQL server.

## ENABLE BCD

Specifies whether BDE translates numeric and decimal fields into floating point values or binary coded decimal (BCD) values. BCD values eliminate the rounding errors associated with floating point math (such as a 3 * (2/3) resulting in 2.00000000001). The following table lists field types converted to BCD when ENABLE BCD is set to TRUE:

| Driver | Field types |
|---|---|
| DB2 | DECIMAL and NUMERIC |
| dBASE | All NUMBER and FLOAT fields |
| Informix | DECIMAL, NUMERIC, and MONEY |
| InterBase | DECIMAL and NUMERIC |
| MS SQL | MONEY, SMALLMONEY, DECIMAL, and NUMERIC |
| ODBC | DECIMAL and NUMERIC |
| Oracle | NUMBER and FLOAT |
| Paradox | BCD |
| Sybase | MONEY, SMALLMONEY, DECIMAL, and NUMERIC |

**Note:** A field is translated to BCD only if the target driver supports BCD fields.

With SQL drivers (all non-local drivers), the ENABLE BCD property must be set in the driver area of the configuration. The driver setting can be overridden by setting it in the alias; however, if it is blank in the driver area, setting it in the alias has no effect.

With local drivers, you can only set ENABLE BCD in an alias that is of type STANDARD. This is because a STANDARD alias can be either of dBASE or Paradox type.

**Default:**
FALSE (Convert to floating point values)

## Default and passthrough SQL query modes

There are two SQL Links driver options that influence how your application behaves when querying SQL data:

- Default SQL Query Mode
- Default Passthrough SQL Mode

For information on how to use your application to query SQL data, or how to query Paradox or dBASE data by using SQL statements, see Borland Local SQL Online Guide. 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 database)
- 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 Links 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 BDE Configuration Utility to modify the default SQL database alias, setting default SQLQRYMODE to SERVER. (See Creating and managing SQL aliases)

### Default passthrough 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 BDE Configuration Utility, you can also specify whether or not SQL statements passed directly to the server should be automatically committed.

For a discussion of SQL passthrough options, see SQLPASSTHRU MODE. Also see the parameter, SQLQRYMODE.

For information about using passthrough SQL with your BDE application, see your application's user documentation.

---

{button ,AL("config")} Other SQL Links configuration topics

## Creating and managing SQL aliases: Overview

Your SQL alias includes your user name on the target SQL server, and is required to access any SQL data through your BDE application. A generic SQL alias is automatically created the first time you modify the default SQL Links driver parameters after installation.

To set up a standard alias, use the BDE Configuration Utility to assign a name to, and specify the path name for, a directory containing Paradox or dBASE files.

To set up an alias for a SQL database, use the BDE Configuration Utility to assign a name to, and customize the access parameters for, a SQL server and database.

---

{button ,AL("alias")} SQL alias topics

## Adding a new SQL alias

To add a new alias,

1. Start the BDE Configuration Utility (BDECFG32.EXE).
2. Click the Aliases page and choose the New Alias button. The Add New Alias dialog box appears.
3. Enter the new alias name and select the SQL-specific alias type. Then choose OK to begin the setup process. The Aliases page displays all the configuration parameters you can change to customize the new alias.

   The parameters are the same as those listed in the Drivers page for your driver type. For a description of each setting you can change, see Specifying default SQL Links driver settings
4. If desired, edit the settings for the category you selected. If you leave any categories blank, the Aliases page assumes you want to use the default for that driver type.
5. When you are finished, click File|Save to save the new alias in the default configuration file; click File|Save As to save the new alias in a configuration file with a different name.

If you save the new alias in a configuration file with a different name, the BDE Configuration Utility displays a Non-System Configuration File dialog box:

- If you want to activate this configuration file next time BDE is restarted, choose Yes.
- If you want to keep using the current default configuration file, choose No.

Your changes take effect the next time BDE is restarted.

---

{button ,AL("alias")} SQL alias topics

## Modifying an existing SQL alias

To change a default parameter type for all aliases of that driver type, follow the procedure described in Specifying default SQL Links 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 Aliases page. The parameters are the same as those listed in the Drivers page.
3. Highlight the configuration parameter you wish to change, and enter the desired value. If you leave any categories blank, the Aliases page uses the driver's default value.
4. When you are finished, click File|Save to save the new alias in the default configuration file; click File|Save As to save the new alias in a configuration file with a different name.

Click Save or Save As followed by a Yes to make your changes take effect the next time BDE is restarted.

---

{button ,AL("alias")} SQL alias topics

## Deleting a 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, click File|Open to load that configuration file.
2. Highlight the name of the alias to delete, and click the Delete Alias button.
3. Click File|Save to save your changes in the default configuration file; click File|Save As to save your changes in a different configuration file.

---

{button ,AL("alias")} SQL alias topics

## Configuring and connecting to a SQL server

Whenever you attempt an operation against a 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 BDE files and your SQL Links driver are configured properly, you should be able to use your application's File menu to select a table in your SQL database using the appropriate alias. See Creating and managing SQL aliases

If you have trouble connecting, see the troubleshooting procedures in your driver-specific section 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 the Windows Registry before you start your BDE application. One way to do this is to update the BDE section of Registry directly, changing the CONFIGFILE01= statement to reflect the desired configuration file. Another way to do this is to open and save the file by using the BDE Configuration Utility. If directed at Save time, the BDE Configuration Utility automatically updates the Registry.

For complete information on installing, configuring, using, and troubleshooting Borland SQL Links drivers, see these topics:

- Connecting to InterBase
- Connecting to DB2
- Connecting to Informix
- Connecting to Oracle
- Connecting to Sybase
- Connecting to Microsoft SQL Server

## Connecting to InterBase

This section contains topics about using Borland SQL Links that are unique to InterBase. These settings are applicable only if you have purchased and installed the Borland SQL Links Driver for InterBase.

- InterBase requirements
- InterBase installation changes
- Working with InterBase servers
- Troubleshooting InterBase connection problems
- InterBase data type translations
- InterBase field-naming rules

## InterBase requirements

Here is a list of software that should already be installed and running at the InterBase server before you install Borland SQL Links for Windows.

| Server requirements | Description |
|---|---|
| Database server software | InterBase version 4.0 or later |
| 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.

Here is a list of requirements for the client workstation.

| Client requirements | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Requirements | 1.5 MB of free disk space.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 BDECFG32.EXE and the BDE configuration file). This directory is modified during SQL Links installation. |
| Network protocol software | Network protocol software compatible with both the server network protocol and the client workstation client database communication driver. InterBase supports the Windows 95 Winsock API. |
| HOSTS file | HOSTS file containing the name and IP address of each server to which 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<br>If your server is running Domain Name Service (DNS), this information may be configured for you. |
| 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. |

---

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

## InterBase installation changes

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

| Item added | Description |
|---|---|
| SQLINT32.DLL | Dynamic Link Library comprising the new InterBase driver and its supporting files. |
| IB_LICEN.DAT | InterBase license file. |
| INTRBASE driver type | Added to Configuration Utility Drivers page to enable basic configuration of Borland InterBase SQL Links driver. |
| INTRBASE alias type | Added to Configuration Utility Aliases page to enable creation of an alias that can connect to a SQL Server database. |
| INTERBAS.MSG | InterBase message files, usually installed in C:\INTERBAS. |
| GDS32.DLL | InterBase-supplied.DLL. |
| 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. |

---

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

## 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.

Here is a list of the general items that you might find helpful in working with InterBase.

| Item | Description |
|---|---|
| Dynamic Link Library (DLL) name | SQLINT32.DLL |
| Case-sensitive for data? | Yes (including pattern matching) |
| Case-sensitive for objects (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 passthrough 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 |
| Language drivers? | InterBase transliterates data from its current language driver to the language driver BDE is using. |

---

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

## Troubleshooting InterBase connection problems

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

1. Determine if you can connect to the InterBase server from your client workstation by using the Windows ISQL tool that ships with Borland's Windows Client for InterBase. If the connection succeeds, a status message appears.

   If no connection—consult your database administrator.

   If connected—continue with step 2.
2. Verify that your InterBase SQL Links driver is correctly installed.
3. Reinstall SQL Links by following the procedures in Installing the Software.
4. 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 5.

   **Note:** The following steps require a TELNET program and a PING program. These DOS programs are not included in the SQL Links 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.
5. Test the lower-level protocols.

   a) 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 but you still cannot connect, you may have a problem with your InterBase installation. See your database administrator for assistance.

   b) 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 might be a problem with the inet daemon.

      If you cannot PING the server, you might 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 6.
6. Confirm that you have a login set in the InterBase security database, isc4.gdb.

   If so—continue with Step 7.
7. 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 BDE Configuration Utility and examine your InterBase alias.

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

## InterBase data type translations

Certain database operations cause SQL Links 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 Links 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 Links converts any data originating in a SQL database to Paradox or dBASE format (depending on the answer format requested) before placing the data in the local answer table.

The following topics list InterBase, Paradox, and dBASE data types and show how SQL Links translates between them:

- From InterBase data types
- From Paradox data types
- From dBASE data types
- From Paradox physical types
- From dBASE physical types
- From InterBase physical types
- InterBase equivalents to standard SQL data types
- InterBase field-naming rules

---

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

## From InterBase data types (to Paradox, dBASE)

| 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 | DateTime | Date |
| 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 | Binary | Memo |

---

{button ,AL("interbasetranslations")} InterBase data translations

## From Paradox data types (to InterBase, dBASE)

| 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 |

---

{button ,AL("interbasetranslations")} InterBase data translations

## From dBASE data types (to InterBase, Paradox)

| FROM dBASE | TO InterBase | TO Paradox |
|---|---|---|
| Character(n) | VARYING(n) | Alphanumeric(n) |
| Number | Float | SHORT, DOUBLEDOUBLEShort number, NumberNumber |
| Date | DATE | Date |
| Lock | Character {24} | Alpha {24} |
| Bytes | BLOB | Bytes |
| Bool | Character {1} | Bool |
| Memo | BLOB/1 | Memo in append, copy, and local query operations. |

---

{button ,AL("interbasetranslations")} InterBase data translations

## From Paradox physical types (to BDE, dBASE)

| 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 |

---

{button ,AL("interbasetranslations")} InterBase data translations

## From dBASE physical types (to BDE, Paradox)

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

---

{button ,AL("interbasetranslations")} InterBase data translations

## From InterBase physical types (to BDE, Paradox, dBASE)

| 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 < 255 | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldIBCHAR > 255 | fldBLOB | fldSTMEMO | fldDBCHAR |
| fldIBVARYING < 255 | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldIBVARYING > 255 | fldBLOB | fldSTMEMO | fldDBCHAR |
| fldIBDATE | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |
| fldIBBLOB | fldBLOB | fldPDXBINARYBLOB | fldDBMEMO |
| fldIBTEXTBLOB | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |

---

{button ,AL("interbasetranslations")} InterBase data translations

## InterBase equivalents to standard SQL data types

When you use passthrough SQL commands to create or alter an InterBase table, you must use standard SQL data types. Here is a list of standard SQL data types and their corresponding InterBase data types.

| 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.

---

{button ,AL("interbasetranslations")} InterBase data translations

## InterBase field-naming rules

Here is a list of field-naming rules for Paradox, dBASE, and InterBase.

| Naming rule | Paradox | dBASE | InterBase |
|---|---|---|---|
| Max length (characters) | 25 | 10 | 31 |
| Valid characters | 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) |

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

---

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

{button ,AL("interbasetranslations")} InterBase data translations

## Connecting to Informix

This section 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 requirements
- Informix installation changes
- Working with Informix servers
- Special SQL Links driver settings for Informix
- Troubleshooting Informix connection problems
- Borland language drivers for Informix
- Informix data type translations

## INFORMIX-SQL requirements

Here is a list of software that should already be installed and running at the INFORMIX-SQL server before you install the Borland Informix SQL Links driver:

| Server requirements | Description |
|---|---|
| Database server software | Informix version 4.10, 5.01, 6, or 7 |
| Network protocol software | TCP/IP or SPX network protocol software compatible with both the database server and the client workstation network protocol. |

Here is a list of software and related files that should already be installed and running at the client workstation.

| Client requirements | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Informix PC client software | Informix ESQL/C 7.2 for Windows 95 and NT |
| Requirements | 1.5 MB of free disk space. Hardware and operating system that meets the requirements of your Borland desktop product. |
| Access rights | (For desktop products installed on the network server only.)<br>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 the BDE Configuration Utility (BDECFG.EXE) and the BDE configuration file (IDAPI.CFG). 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. |

### Informix 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.

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## Informix installation changes

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

| Item added | Description |
|---|---|
| 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. |
| READLINK.TXT | Borland SQL Links for Windows README file. |
| SQLINF32.DLL | Dynamic Link Library comprising the INFORMIX SQL Link driver and its supporting files. |

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## 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.

- Informix data type translations
- Informix transaction isolation levels
- Informix field-naming rules

This table lists the general items that you might find helpful in working with Informix.

| Item | Description |
|---|---|
| Dynamic Link Library (DLL) name | SQLINF32.DLL |
| Case-sensitive for data? | Yes (including pattern matching) |
| 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 passthrough 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 |
| Multiple client connections? | Yes (Informix 5 or later) |
| Run-time specification of server information, such as Server Name, User Name, and Password? | Yes |
| Stored procedures? | Yes. See *BDE Online Reference* |

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## Informix transaction isolation levels

Informix supports four 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.

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

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## Special SQL Links driver settings for Informix

These Configuration Utility parameters are unique to Informix:

**COLLCHAR**
Used with DBNLS to specify whether you can access NLS databases. This lets CHAR/VARCHAR database requests from an application be interpreted as NCHAR/NVARCHAR requests. If DBNLS is '2' and COLLCHAR '1', you can access an NLS database.

**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.

**DBNLS**
Used with COLLCHAR to specify whether you can access NLS databases. If DBNLS is NULL, you can't create or access an NLS database. If DBNLS is '2' and COLLCHAR is '1', you can access or create an NLS database.

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

| 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 | Suspends the process until the lock is released, or until the end of the specified number of seconds.<br>Default: 5 |

**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.

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## Troubleshooting Informix connection problems

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

1. Confirm that the Informix server is online.

   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.

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## 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.

| 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 |
| DB2 SQL ANSI | db2andeu | 1252 (ANSI) | Dictionary |
| 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 |

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## 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.

The following topics contain tables showing Informix, Paradox, and dBASE data types and show how SQL Link translates between them in append, copy, and local query operations:

- From Informix data types
- From Paradox data types
- From dBASE data types
- From Paradox physical types
- From dBASE physical types
- From Informix physical types
- Using the Informix Interval data type

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## From Informix data types (to Paradox, dBASE)

| 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} |
| Byte | Binary | Memo |
| Text | Memo | Memo |
| Varchar | Alpha | Character |

The following data types are added when using Informix 5 or later

| From Informix 5 | To Paradox | To dBASE |
|---|---|---|
| NChar <255 | Alpha | Character |
| NChar >255 | Memo | Memo |
| NVarchar | Alpha | Character |

---

{button ,AL("informixdatatranslations")} Informix data translations

## From Paradox data types (to dBASE, Informix)

| 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 |

---

{button ,AL("informixdatatranslations")} Informix data translations

## From dBASE data types (to Paradox, Informix)

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

**Note:** 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.

---

{button ,AL("informixdatatranslations")} Informix data translations

## From Paradox physical types (to BDE, dBASE)

| 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 |

---

{button ,AL("informixdatatranslations")} Informix data translations

## From dBASE physical types (to BDE, Paradox)

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | fldINT16 else fldFLOAT<br>fldPDXSHORT<br>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 |

---

{button ,AL("informixdatatranslations")} Informix data translations

## From Informix physical types (to BDE, Paradox, dBASE)

| Informix physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldINFCHAR < 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} |
| 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 |

---

{button ,AL("informixdatatranslations")} Informix data translations

## Using the Informix Interval data type

BDE applications 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, when 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.*

---

{button ,AL("informixdatatranslations")} Informix data translations

## Informix field-naming rules

This table lists field-naming rules for Paradox, dBASE, and Informix.

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

**Note:** Paradox field names should not contain:
square brackets []
curly braces {}
pipes |
parentheses ()
the combination ->
or the hash symbol # by itself.

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

{button ,AL("informixdatatranslations")} Informix data translations

## Connecting to ORACLE

This section contains topics about using Borland SQL Links that are unique to ORACLE. These settings are applicable only if you have purchased and installed the Borland SQL Links Driver for ORACLE.

- ORACLE requirements
- ORACLE installation changes
- Working with ORACLE servers
- Special SQL Links driver settings for ORACLE
- Troubleshooting ORACLE connection problems
- Borland language drivers for ORACLE
- ORACLE data type translations
- ORACLE field-naming rules

## ORACLE requirements

This table lists software that should already be installed and running at the ORACLE server before you install the Borland ORACLE SQL Links driver.

| Server requirements | Description |
|---|---|
| Database server software | ORACLE version 7.0 or later |
| Network protocol software | ORACLE SQL*NET |

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

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

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

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## ORACLE installation changes

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

| Item added | Description |
|---|---|
| SQLORA32.DLL | Dynamic Link Library comprising the new ORACLE SQL Links driver and its supporting files. |
| ORACLE driver type | Added to Configuration Utility Drivers page to enable basic configuration of Borland ORACLE SQL Links driver. |
| ORACLE alias type | Added to Configuration Utility Aliases page to enable creation of an alias that can connect to an ORACLE database. |

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## Special SQL Links driver settings for ORACLE

The following topics describe driver configuration parameters that are unique to ORACLE:

**ENABLE INTEGERS**

Specifies whether the BDE translates NUMERIC fields with no scale into logical integers.

**Note**: If this setting and ENABLE BCD are both TRUE, ENABLE INTEGERS takes precedence.

**Default:**

FALSE (don't convert NUMERIC fields into logical integers)

**NET PROTOCOL**

Network transport is used to communicate with the SQL server. This table describes NET PROTOCOL options.

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

**VENDOR INIT**

Enter the vendor DLL filename.

**Oracle synonyms**

Synonyms are alternate names for other objects, such as tables or views. The special SQL Links Driver setting for ORACLE, called LIST SYNONYMS, determines whether or not to include synonyms in the schema table returned from BDE API functions DbiOpenTableList and DbiOpenFileList, as shown in the following table.

| Value | Meaning |
|---|---|
| NONE | Do not include any synonyms (default) |
| PRIVATE | Only include private synonyms |
| ALL | Include both private and public synonyms |

Synonyms are supported by a field in the BDE TBLBaseDesc structure called *bSynonym*. The field *bSynonym* is a BOOL16, which is set to TRUE if the object is a synonym.

**Cross database and cross server table access**

You can access Oracle synonyms by using DbiOpenTable. You can access tables on different databases or servers in the same way. For example, you can open "SCOTT"."CUSTOMER"@"DBLINK" on Oracle and pubs.dbo.authors on Sybase.

**Public synonyms**

Oracle has PUBLIC synonyms that appear in the table list when the value of LIST SYNONYMS = ALL. However, to open a PUBLIC synonym, the user must also have SELECT privileges on the base object of the synonym. If the user does not have SELECT privileges and tries to open the PUBLIC SYNONYM, Oracle returns the error "Table or view does not exist".

Oracle has PUBLIC synonyms to a set of dynamic performance tables. Even though these are PUBLIC synonyms, they are accessible only to the DBA user SYS, by default (other users can be granted privileges). These synonym names are in the format, V$... (that is, V$DATABASE, V$ACCESS, and so on).

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## Working with ORACLE servers

This table lists the general items that you might find helpful in working with ORACLE servers.

| Item | Description |
|---|---|
| Dynamic Link Library (DLL) name | SQLORA32.DLL |
| Case-sensitive for data? | Yes |
| Case-sensitive for objects (such as tables, columns and indexes)? | Yes |
| 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 passthrough SQL? | No |
| Implicit row IDs | Yes |
| Blob handles | No |
| Maximum size of single blob read (if blob handles are not supported) | 64K |

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## Troubleshooting ORACLE connection problems

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

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. 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.

   3.1 Use SQL*DBA or SQL*Plus to enter a CONNECT command. For example:

   ```
   SQLDBA> connect jlee/trapper @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;
   ```

   3.2 Verify that your path includes C:\ORACLEx\BIN

   (ORACLEx is the name of your ORACLE directory.)

   3.3 Verify that the file ORACLE\ADMIN\NETWORK\TNSNAMES.ORA contains the correct information to your target Oracle server.

   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 BDECFG32.EXE alias setup. Run the Configuration Utility and examine your ORACLE alias.

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

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## 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.

| Long name | Short 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 |
| DB2 SQL ANSI | db2andeu | 1252 (ANSI) | Dictionary |
| 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 Links ROMAN8 | BLROM800 | ROMAN8 | Binary |

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## ORACLE data type translations

Certain database operations cause SQL Links to convert data from Paradox or 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 Links 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 Links converts any data originating in a SQL database to Paradox or dBASE format (depending on the answer format requested) before placing the data in the local answer table.

The following topics contain tables that list ORACLE, Paradox, and dBASE data types and show how SQL Links translates between these data types:

- From ORACLE data types
- From Paradox data types
- From dBASE data types
- From Paradox physical types
- From dBASE physical types
- From ORACLE physical types

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

## From ORACLE data types (to Paradox, dBASE)

| FROM: ORACLE | TO: Paradox | TO: dBASE |
|---|---|---|
| CHAR(n) | VARCHAR(n) | Alphanumeric(n)Character(n) |
| DATE {date/time} | DateTime | Character {date/time} |
| FLOAT | 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 |

---

{button ,AL("oracletranslations")} ORACLE data translations

## From Paradox data types (to ORACLE, dBASE)

| FROM: Paradox | TO: ORACLE | TO: dBASE |
|---|---|---|
| Alphanumeric(n) | CHAR(n) | Character(n) |
| Number | NUMBER | Float {20.4} |
| Money | NUMBER | Float {20.4} |
| Date | DATE {date/time} | 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} |
| Datetime | DATE | Character {>8} |
| Logical | CHARACTER {1} | Bool |
| AutoInc | NUMBER | Number {11.0} |
| Bytes | LONG RAW | BYTES |
| BCD | N/A | N/A |

---

{button ,AL("oracletranslations")} ORACLE data translations

## From dBASE data types (to ORACLE, Paradox)

| FROM: dBASE | TO: ORACLE | TO: Paradox |
|---|---|---|
| Character | CHAR(n) | Alphanumeric(n) |
| Number1 | NUMBER | Short, Number |
| Float | NUMBER | Short, Number |
| Date | DATE {date/time} | Date |
| Logical | CHAR(1) | Bool |
| Memo | LONG | Memo |
| Bytes | LONG RAW | Bytes |
| Lock | CHAR(24) | Alphanumeric {24}. |

---

{button ,AL("oracletranslations")} ORACLE data translations

## From Paradox physical types (to BDE, dBASE)

| 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 |

---

{button ,AL("oracletranslations")} ORACLE data translations

## From dBASE physical types (to BDE, Paradox)

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | fldINT16 else fldFLOAT<br>fldPDXSHORT<br>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 |

---

{button ,AL("oracletranslations")} ORACLE data translations

## From ORACLE physical types (to BDE, Paradox, dBASE)

| ORACLE physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldORACHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldORARAW | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldORADATE | fldTIMESTAMP | fldPDXDATETIME | fldDBCHAR |
| 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} |

---

{button ,AL("oracletranslations")} ORACLE data translations

## ORACLE field-naming rules

Here is a list of field-naming rules for Paradox, dBASE, and ORACLE.

| Naming rule | Paradox | dBASE | ORACLE |
|---|---|---|---|
| Max length (characters) | 25 | 10 | 30 |
| Valid characters | 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 |

**Note:** You may not 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*.

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

{button ,AL("oracletranslations")} ORACLE data translations

## Connecting to Sybase

This section contains topics about using Borland SQL Links that are unique to Sybase. These settings are applicable only if you have purchased and installed the Borland SQL Links Driver for Sybase.

- Sybase requirements
- Sybase installation changes
- Working with Sybase servers
- Special SQL Links driver settings for Sybase
- Troubleshooting Sybase connection problems
- Borland language drivers for Sybase
- Sybase data type translations
- Sybase field-naming rules

## Sybase requirements

Here is a list of software that should already be installed and running at the Sybase server before you install the Borland Sybase SQL Links driver.

| Server requirements | Description |
|---|---|
| Database server software | Version 4.x and earlier with DB-Lib interface<br>OR:<br>System 10 and later with CT-Lib interface |
| Network protocol software | Supporting the Sybase Net libraries |

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

Here is a list of software that should already be installed and running at the client workstation:

| Client requirements | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Requirements | 1.5 MB of free disk space.Hardware and operating system that meets the requirements of your BDE application. |
| Access rights | (For applications installed on the network server only.)<br>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 BDECFG32.EXE and IDAPI.CFG). This directory is modified during SQL Links installation. |
| Network protocol software | Network protocol software compatible with both the server network protocol and the client workstation client database communication driver. |

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## Sybase installation changes

When you install a Borland Sybase SQL Links driver the following changes take place in your workstation system:

| Item added | Description |
|---|---|
| SQLSYB32.DLL | Dynamic Link Library comprising the Sybase driver and its supporting files (for DB-Lib/Version 4.x). |
| SQLSSC32.DLL | Dynamic Link Library comprising the Sybase driver and its supporting files (for CT-Lib/System 10 and later). |
| Sybase driver type | Added to Configuration Utility Drivers page to enable basic configuration of Borland Sybase SQL Links driver. |
| Sybase alias type | Added to Configuration Utility Aliases page to enable creation of an alias that can connect to a Sybase database. |

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## Working with Sybase servers

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

Here is a list of general items that you might find helpful in working with Sybase:

| Item | Description |
|---|---|
| Product name | Sybase SQL Server Version 4.x<br>OR:<br>Sybase SQL Server System 10 and later |
| SQL dialect | Sybase TRANSACT-SQL |
| Dynamic Link Library (DLL) name | SQLSYB32.DLL (for DB-Lib/Version 4.x)<br>SQLSSC32.DLL (for CT-Lib/System 10 and later) |
| 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 passthrough SQL? | Yes |
| Implicit row IDs | No |
| Blob handles | No |
| Maximum size of single BLOB read (if BLOB handles are not supported) | 32K |

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## Special SQL Links driver settings for Sybase

The following topics describe driver configuration parameters that are unique to Sybase:

**APPLICATION NAME**

Specifies an application name used in the sysprocesses table of the master database to help the server identify your process.

**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 log 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).

**HOST NAME**

Specifies a workstation name used in the sysprocesses table of the master database to help the server identify your process. Also appears when sp_who is executed on the server.

**NATIONAL LANG NAME**

Specifies the language to use when displaying error messages. If blank, the default SQL server language is used. Only applies to servers with language support installed.

**TDS PACKET SIZE**

Specifies the size of tabular data stream (TDS) packets exchanged by the client and server. TDS PACKET SIZE can be a value from 0 to 65535. Larger packet sizes can improve performance by reducing the number of packet requests, especially when working with BLOBs. A packet size between 4096 and 8192 is recommended for optimal performance.

To determine the TDS PACKET SIZE settings supported by your server, send the SQL command sp_configure to your Sybase or MS SQL server, then note the settings returned for MAX NETWORK PACKET SIZE and ADDITIONAL NETMEM. If TDS PACKET SIZE is set to a higher value than the maximum network packet size, or ADDITIONAL NETMEM is not set, the following errors can occur:

Error : unknown user name or password

Server error

4002 Login failed

20014 Login incorrect

TDS 5 or later is required to modify the packet size. See the Sybase System Administrator's Guide for more information.

**Default:**
512

**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 Links will wait for query execution to complete before canceling the operation. The default is 3,600 seconds.

**Note:** Applications using earlier versions of SQL Links for Sybase may generate an error message if a transaction isolation level other than Repeatable Read has been specified. In such cases, specify an isolation level of Read Committed when starting a transaction or use the BDE configuration utility to set the Driver Flags parameter to 512.

Although the beginTrans (dBASE) or beginTransaction (Paradox) methods indicate that the isolation level is optional, when using SQL Links for Sybase, specify the Read Committed isolation level.

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## Troubleshooting Sybase connection problems

If you have problems establishing a Sybase connection with SQL Links,

1. Confirm that the Sybase 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 tools to verify the connection at each layer.
4. Use the DOS or Windows PING.EXE program to check the network connection.
5. Use the Sybase SYBPING.EXE to verify connection to your target server.
   If you are not connected—consult your database administrator.
   If you are connected—continue with Step 6.
6. 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 BDECFG32.EXE alias setup. Run the Configuration Utility and examine your Sybase alias.

For more information on your vendor-supplied diagnostic tools, see your Sybase documentation.

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## Borland language drivers for Sybase

The following table lists language drivers available for use with Sybase, 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.

| Long name | Short 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 |
| DB2 SQL ANSI | db2andeu | 1252 (ANSI) | Dictionary |
| 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 Links 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 |

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## Sybase data type translations

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

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

The following topics contain lists of Sybase, Paradox, and dBASE data types and show how SQL Links translates between these data types:

- From Sybase data types
- From Paradox data types
- From dBASE data types
- From Paradox physical types
- From dBASE physical types
- From Sybase physical types

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

## From Sybase data types (to Paradox, dBASE)

| FROM: Sybase | 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} |
| 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) |

If you are using the SQLSSC32.DLL (for CT-Lib/System 10 and later), the following additional data types are available:

| FROM: Sybase | TO: Paradox | TO: dBASE |
|---|---|---|
| Numeric | Number | Float {20.4} |
| Decimal (8,5) | Number | Float {20.4} |
| Double Precision | Number | Float {20.4} |

---

{button ,AL("sybasetranslations")} Sybase data translations

## From Paradox data types (to Sybase, dBASE)

| FROM: Paradox | TO: Sybase | 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 |

---

{button ,AL("sybasetranslations")} Sybase data translations

## From dBASE data types (to Sybase, Paradox)

| FROM: dBASE | TO: Sybase | TO: Paradox |
|---|---|---|
| Character(n) | varchar(n) | Alphanumeric(n) |
| Number | int, float | Number |
| Float number1 | int, float | Number |
| Date | datetime | Date |
| Bool | bit | Bool |
| Lock | character {24.0} | Alphanumeric {24.0} |
| Bytes | image | Bytes |
| Memo | text | Memo |

---

{button ,AL("sybasetranslations")} Sybase data translations

## From Paradox physical types (to BDE, dBASE)

| 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 |

---

{button ,AL("sybasetranslations")} Sybase data translations

## From dBASE physical types (to BDE, Paradox)

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | fldINT16 else fldFLOAT<br>fldPDXSHORT<br>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 |

---

{button ,AL("sybasetranslations")} Sybase data translations

## From Sybase physical types (to BDE, Paradox, dBASE)

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

If you are using Sybase System 10, the following additional physical types are available:

| From Sybase physical type | To BDE logical type | To Paradox physical type | To dBASE physical type |
|---|---|---|---|
| fldSYBDECIMAL | fldFLOAT | fldPDXNUM | fldDBFLOAT(20,4) |
| fldSYBNUMERIC | fldFLOAT | fldPDXNUM | fldDBFLOAT(20,4) |

---

{button ,AL("sybasetranslations")} Sybase data translations

## Sybase field-naming rules

Here is a list of field-naming rules for Paradox, dBASE, and CT-LIB.

| Naming rule | Paradox | dBASE | Sybase |
|---|---|---|---|
| Max length (chars) | 25 | 10 | 30 |
| Valid characters | 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 |

**Note:** You may not use Sybase reserved words for remote table and column names.

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

{button ,AL("sybasetranslations")} Sybase data translations

## Connecting to Microsoft SQL Server

This section contains topics about using Borland SQL Links that are unique to Microsoft SQL Server. These settings are applicable only if you have purchased and installed the Borland SQL Links Driver for Microsoft SQL Server.

**NUMERIC and DECIMAL types**

BDE supports the Microsoft SQL Driver 6.0 NUMERIC and DECIMAL data types. These types are translated to logical fldFLOAT, but they are retrieved from the server in the native NUMERIC format so that a user is guaranteed no data loss when working with the physical data (xltNONE).

- Microsoft SQL Server requirements
- Microsoft SQL Server installation changes
- Working with Microsoft SQL servers
- Special SQL Links driver settings for Microsoft SQL Server
- Troubleshooting Microsoft SQL Server connection problems
- Borland language drivers for Microsoft SQL Server
- Microsoft SQL Server data type translations
- Microsoft SQL Server field-naming rules

## Microsoft SQL Server requirements

Here is a list of software that should already be installed and running at the Microsoft SQL Server before you install the Borland SQL Links driver for Microsoft SQL Server.

| Server requirements | Description |
|---|---|
| Database server software | Version 4.x and 6.x |
| Network protocol software | 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.

Here are the requirements for the client workstation, including a list of software that should already be installed and running at the client workstation.

| Client requirements | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Requirements | 1.5 MB of free disk space.Hardware and operating system that meets the requirements of your BDE application. |
| Access rights | (For applications installed on the network server only.)<br>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 BDECFG32.EXE and the IDAPI.CFG). This directory is modified during SQL Links installation. |
| Network protocol software | Network protocol software compatible with both the server network protocol and the client workstation client database communication driver. |

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## Microsoft SQL Server installation changes

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

| Item added | Description |
|---|---|
| SQLMSS32.DLL | Dynamic Link Library comprising the Microsoft SQL driver and its supporting files. |
| Driver type | Added to Configuration Utility Drivers page to enable basic configuration of Borland Microsoft SQL Server SQL Links driver. |
| Alias type | Added to Configuration Utility Aliases page to enable creation of an alias that can connect to a Microsoft SQL Server database. |

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## Working with Microsoft SQL servers

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

Here is a list of general items that you might find helpful in working with Microsoft SQL Server:

| Item | Description |
|---|---|
| Product name | Microsoft SQL Server |
| SQL dialect | Microsoft TRANSACT-SQL |
| Dynamic Link Library (DLL) name | SQLMSS32.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 passthrough SQL? | Yes |
| Implicit row IDs | No |
| Blob handles | No |
| Maximum size of single BLOB read (if BLOB handles are not supported) | 32K |

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## Special SQL Links driver settings for Microsoft SQL Server

The following topics describe driver configuration parameters that are unique to Microsoft SQL Server:

**APPLICATION NAME**
Specifies an application name used in the sysprocesses table of the master database to help the server identify your process.

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

**HOST NAME**
Specifies a workstation name used in the sysprocesses table of the master database to help the server identify your process. Also appears when sp_who is executed on the server.

**NATIONAL LANG NAME**
Specifies the language to use when displaying error messages. If blank, the default SQL server language is used. Only applies to servers with language support installed.

**TDS PACKET SIZE**
Specifies the size of tabular data stream (TDS) packets exchanged by the client and server. TDS PACKET SIZE can be a value from 0 to 65535. Larger packet sizes can improve performance by reducing the number of packet requests, especially when working with BLOBs. A packet size between 4096 and 8192 is recommended for optimal performance.

To determine the TDS PACKET SIZE settings supported by your server, send the SQL command sp_configure to your Sybase or MS SQL server, then note the settings returned for MAX NETWORK PACKET SIZE and ADDITIONAL NETMEM. If TDS PACKET SIZE is set to a higher value than the maximum network packet size, or ADDITIONAL NETMEM is not set, the following errors can occur:

Error : unknown user name or password

Server error

4002 Login failed

20014 Login incorrect

TDS 5 or later is required to modify the packet size. See your System Administrator's Guide for more information.

**Default:**
512

**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.

**DATABASE NAME**
Name of the target SQL database.

**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.

**DATE MODE**

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

**MAX QUERY TIME**

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

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## Troubleshooting Microsoft SQL Server connection problems

If you have problems establishing a Microsoft SQL Server connection with SQL Links,

1. Confirm that the Microsoft 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 Microsoft tools to verify the connection at each layer.
4. Use the Microsoft SQL Server utility (ISQL.EXE) to verify the connection.
   If you are not connected—consult your database administrator.
   If you are connected—continue with Step 5.
5. Check whether your BDE application 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 BDECFG32.EXE alias setup. Run the Configuration Utility and examine the alias.

For more information on your vendor-supplied diagnostic tools, see your Microsoft SQL Server documentation.

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## Borland language drivers for Microsoft SQL Server

The following table lists language drivers available for use with SQL servers, 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.

| Long name | Short 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 |
| DB2 SQL ANSI | db2andeu | 1252 (ANSI) | Dictionary |
| 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 Links 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 |

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## Microsoft SQL Server data type translations

Certain database operations cause SQL Links to convert data from Paradox or dBASE format to SQL Server format. For example, a BDE application that copies or appends local data to a SQL Server table causes SQL Links to convert the local data to Microsoft 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 Links converts any data originating in a SQL database to Paradox or dBASE format (depending on the type of answer required) before placing the data in the local answer table.

The following topics contain lists of SQL Server, Paradox, and dBASE data types and show how SQL Links translates between these data types:

- From Microsoft SQL server data types
- From Paradox data types
- From dBASE data types
- From Paradox physical types
- From dBASE physical types
- From Microsoft SQL Server physical types

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## From Microsoft SQL Server data types (to Paradox, dBASE)

| 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} |
| 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) |

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

## From Paradox data types (to SQL Server, dBASE)

| 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 |

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

## From dBASE data types (to SQL Server, Paradox)

| FROM: dBASE | TO: SQL Server | TO: Paradox |
|---|---|---|
| Character(n) | varchar(n) | Alphanumeric(n) |
| Number | int, float | Number |
| Float number1 | int, float | Number |
| Date | datetime | Date |
| Bool | bit | Bool |
| Lock | character {24.0} | Alphanumeric {24.0} |
| Bytes | image | Bytes |
| Memo | text | Memo |

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

## From Paradox physical types (to BDE, dBASE)

| 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 |

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

## From dBASE physical types (to BDE, Paradox)

| dBASE physical | BDE logical | Paradox physical |
|---|---|---|
| fldDBCHAR | fldZSTRING | fldPDXCHAR |
| fldDBNUM | if( iUnits2=0 && iUnits1<5 ) | fldINT16 else<br>fldFLOAT<br>fldPDXSHORT<br>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 |

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

## From Microsoft SQL Server physical types (to BDE, Paradox, dBASE)

| From MS SQL physical type | To BDE logical type | To Paradox physical type | To dBASE physical type |
|---|---|---|---|
| fldMSSBINARY | fldBYTES | fldPDXBYTES | fldDBMEMO |
| fldMSSBIT | fldBOOL | fldPDXBOOL | fldDBBOOL |
| fldMSSCHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldMSSDATETIME | fldTIMESTAMP | fldPDXDATETIME | fldDBDATE |
| fldMSSDATETIME4 | fldTIMESTAMP | fldPDXDATETIME | fldDBDBDATE |
| fldMSSDECIMAL | fldFLOAT | fldPDXNUM | fldDBFLOAT(20,4) |
| fldMSSFLOAT | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldMSSFLOAT4 | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldMSSIMAGE | fldBLOB/fldstBINARY | fldPDXBINARYBLOB | fldDBMEMO |
| fldMSSINT | fldINT32 | fldPDXLONG | fldDBNUM {11.0} |
| fldMSSMONEY | fldFLOAT/fldstMONEY | fldPDXMONEY | fldDBNUM {20.4} |
| fldMSSMONEY4 | fldFLOAT/fldstMONEY | fldPDXMONEY | fldDBNUM {20.4} |
| fldMSSNUMERIC | fldFLOAT | fldPDXNUM | fldDBFLOAT(20,4) |
| fldMSSSMALLINT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldMSSTEXT | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldMSSTIMESTAMP | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldMSSTINYINT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldMSSVARBINARY | fldVARBYTES | fldPDXBINARYBLOB | fldDBMEMO |
| fldMSSVARCHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

## Microsoft SQL Server field-naming rules

Here is a list of field-naming rules for Paradox, dBASE, and Microsoft SQL Server.

| Naming rule | Paradox | dBASE | SQL Server |
|---|---|---|---|
| Max length (chars) | 25 | 10 | 30 |
| Valid characters | 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 |

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

---

{button ,AL("mstranslations")} Microsoft SQL Server data translations

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

## InterBase driver configuration parameters

Click on the following parameter names for pop-up help configuring the InterBase driver with the BDE Configuration Utility.

- VERSION
- TYPE
- DLL
- DLL32
- DRIVER FLAGS
- TRACE MODE
- SERVER NAME
- USER NAME
- OPEN MODE
- SCHEMA CACHE SIZE
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- SCHEMA CACHE TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- ENABLE BCD

---

{button ,AL("connectingtointerbase")} Connecting to InterBase topics

## Informix driver configuration parameters

Click on the following parameter names for pop-up help configuring the Informix driver with the BDE Configuration Utility.

- VERSION
- TYPE
- DLL
- DLL32
- DRIVER FLAGS
- TRACE MODE
- SERVER NAME
- DATABASE NAME
- USER NAME
- OPEN MODE
- SCHEMA CACHE SIZE
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- LOCK MODE
- DATE MODE
- DATE SEPARATOR
- SCHEMA CACHE TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- ENABLE BCD
- COLLCHAR
- DDBNLS

---

{button ,AL("connectingtoinformix")} Connecting to Informix topics

## DB2 driver configuration parameters

Click on the following parameter names for pop-up help configuring the DB2 driver with the BDE Configuration Utility.

- VERSION
- TYPE
- DLL
- DLL32
- DRIVER
- DRIVER FLAGS
- TRACE MODE
- USER NAME
- DB2 DSN
- OPEN MODE
- SCHEMA CACHE SIZE
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- SCHEMA CACHE TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- ENABLE BCD
- ROWSET SIZE

---

{button ,AL("connectingtoDB2")} Connecting to DB2 topics

## ORACLE driver configuration parameters

Click on the following parameter names for pop-up help configuring the ORACLE driver with the BDE Configuration Utility.

- VERSION
- TYPE
- DLL
- DLL32
- VENDOR INIT
- DRIVER FLAGS
- TRACE MODE
- SERVER NAME
- USER NAME
- NET PROTOCOL
- OPEN MODE
- SCHEMA CACHE SIZE
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- SCHEMA CACHE TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- ENABLE BCD
- ENABLE INTEGERS
- LIST SYNONYMS
- ROWSET SIZE

---

{button ,AL("connectingtooracle")} Connecting to ORACLE topics

**ROWSET SIZE**

Specifies the number of rows to retrieve from the server in a single fetch, and the number of records to insert at a time when using DbiWriteBlock.

**Default:**

The default value is 20 (20 records per server fetch, 20 records inserted at a a time).

**VENDOR INIT**
Enter the vendor DLL filename.

**NET PROTOCOL**

Network transport is used to communicate with the SQL server. This table describes NET PROTOCOL options.

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

**LIST SYNONYMS**

Synonyms are alternate names for other objects, such as tables or views. The special SQL Links Driver setting for ORACLE, called LIST SYNONYMS, determines whether or not to include synonyms in the schema table returned from BDE API functions DbiOpenTableList and DbiOpenFileList, as shown in the following table.

| Value | Meaning |
|---|---|
| NONE | Do not include any synonyms (default) |
| PRIVATE | Only include private synonyms |
| ALL | Include both private and public synonyms |

Synonyms are supported by a field in the BDE TBLBaseDesc structure called *bSynonym*. The field *bSynonym* is a BOOL16, which is set to TRUE if the object is a synonym.

**Cross database and cross server table access**

You can access Oracle synonyms by using DbiOpenTable. You can access tables on different databases or servers in the same way. For example, you can open "SCOTT"."CUSTOMER"@"DBLINK" on Oracle and pubs.dbo.authors on Sybase.

**Public synonyms**

Oracle has PUBLIC synonyms which will show up in the table list when the value of LIST SYNONYMS = ALL. However, to open a PUBLIC synonym, the user must also have SELECT privileges on the base object of the synonym. If the user does not have SELECT privileges and tries to open the PUBLIC SYNONYM, Oracle returns the error "Table or view does not exist".

Oracle has PUBLIC synonyms to a set of dynamic performance tables. Even though these are PUBLIC synonyms, they are accessible only to the DBA user SYS, by default (other users can be granted privileges). These synonym names are in the format, V$... (that is, V$DATABASE, V$ACCESS, and so on).

## Sybase driver configuration parameters

Click on the following parameter names for pop-up help configuring the Sybase driver with the BDE Configuration Utility.

- VERSION
- TYPE
- DLL
- DLL32
- VENDOR INIT
- CONNECT TIMEOUT
- TIMEOUT
- DRIVER FLAGS
- TRACE MODE
- DATABASE NAME
- SERVER NAME
- USER NAME
- OPEN MODE
- SCHEMA CACHE SIZE
- BLOB EDIT LOGGING
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- DATE MODE
- SCHEMA CACHE TIME
- MAX QUERY TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- HOST NAME
- APPLICATION NAME
- NATIONAL LANG NAME
- ENABLE BCD
- TDS PACKET SIZE

---

{button ,AL("connectingtosybase")} Connecting to Sybase topics

**VENDOR INIT**
Enter the vendor DLL filename.

**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.

Default: 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 Links will wait for query execution to complete before canceling the operation. The default is 3,600 seconds.

**Note:** Applications using earlier versions of SQL Links for Sybase may generate an error message if a transaction isolation level other than Repeatable Read has been specified. In such cases, specify an isolation level of Read Committed when starting a transaction or use the BDE configuration utility to set the Driver Flags parameter to 512.

Although the beginTrans (dBASE) or beginTransaction (Paradox) methods indicate that the isolation level is optional, when using SQL Links for Sybase, specify the Read Committed isolation level.

## Microsoft SQL Server driver configuration parameters

Click on the following parameter names for pop-up help configuring the Microsoft SQL Server driver with the BDE Configuration Utility.

- VERSION
- TYPE
- DLL
- DLL32
- VENDOR INIT
- CONNECT TIMEOUT
- TIMEOUT
- DRIVER FLAGS
- TRACE MODE
- DATABASE NAME
- SERVER NAME
- USER NAME
- OPEN MODE
- SCHEMA CACHE SIZE
- BLOB EDIT LOGGING
- LANGDRIVER
- SQLQRYMODE
- SQLPASSTHRU MODE
- DATE MODE
- SCHEMA CACHE TIME
- MAX QUERY TIME
- MAX ROWS
- BATCH COUNT
- ENABLE SCHEMA CACHE
- SCHEMA CACHE DIR
- HOST NAME
- APPLICATION NAME
- NATIONAL LANG NAME
- ENABLE BCD
- TDS PACKET SIZE

---

{button ,AL("connectingtoms")} Connecting to Microsoft SQL Server

**VENDOR INIT**
Enter the vendor DLL filename.

**CONNECT TIMEOUT**

Specifies the amount of time the workstation will retry when attempting to attach to the Microsoft SQL server.

Default: 60 seconds.

**TIMEOUT**

Specifies the maximum amount of time that the workstation waits for results to return from the server.

Default: 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.

**DATABASE NAME**
Name of the target SQL database.

**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 Microsoft SQL Server documentation.

**DATE MODE**

The format in which the driver sends dates to the Microsoft SQL server.

Valid settings are 0 (MDY), 1 (DMY), and 2 (YMD).

**MAX QUERY TIME**

Specifies the maximum amount of time that SQL Links will wait for query execution to complete before canceling the operation.

Default: 3,600 seconds.

## Connecting to DB2

The Borland SQL Links Driver for DB2 is a driver for DB2, IBM's next generation interface for its DATABASE 2 Client/Server product.

This section contains topics about using Borland SQL Links that are unique to the DB2 driver. This information is applicable only if you have installed the Borland SQL Links Driver for DB2.

- DB2 requirements
- DB2 installation changes
- Working with DB2
- Creating a DB2 alias
- Special SQL Links driver settings for B2
- Troubleshooting DB2 connection problems
- Borland language drivers for DB2
- DB2 data type translations
- DB2 field-naming rules

## DB2 requirements

Here is a list of software that should already be installed and running at the DB2 server before you install the Borland SQL Links driver for DB2.

| Server requirements | Description |
|---|---|
| Database server software | DB2 for OS/2 or IBM MVS |
| Network protocol software | Supporting IPX/SPX, TCP/IP, 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.

Here is a list of requirements for the client workstation.

| Client requirements | Description |
|---|---|
| BDE application(s) | Supported BDE application, installed as required by the product documentation. |
| Requirements | 1.5 MB of free disk space. Hardware and operating system that meets the requirements of your BDE application. |
| Access rights | (For applications installed on the network server only.)<br>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 BDECFG32.EXE and the IDAPI.CFG). This directory is modified during SQL Links installation. |
| Network protocol software | Network protocol software compatible with both the server network protocol and the client workstation client database communication driver. |
| DB2 Client Application Enabler | DB2 Client Application Enabler version 2.1 must be installed. Perform the required binding by clicking the Client Setup icon. For details, see the IBM DB2 manual *Installing and Using DB2 Client for Windows*. |

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## DB2 installation changes

When you install the Borland DB2 SQL Links driver, the following changes take place in your workstation system:

| Item added | Description |
|---|---|
| SQLDB232.DLL | The DB2 driver (Dynamic Link Library). |
| DB2 driver type | Added to Configuration Utility Driver page to enable basic configuration of Borland DB2 SQL Links driver. |
| DB2 alias type | Added to Configuration Utility Alias page to enable creation of an alias that can connect to a DB2 database. |
| READLINK.TXT | Borland SQL Links for Windows README file. |

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## Working with DB2

This section provides general information about DB2 and its implementation of SQL.

| Item | Description |
|---|---|
| Product name | SQL Links driver for DB2 |
| SQL dialect | ANSI 92 standard |
| Dynamic Link Library (DLL) name | SQLDB232.DLL |
| Case-sensitive for data? | Yes |
| Case-sensitive for objects (such as tables, columns, and indexes)? | Yes |
| Does the server require that you explicitly start a transaction for multi-statement transaction processing in passthrough SQL? | Yes |
| Implicit row IDs | No |
| Blob handles | Yes |
| Maximum size of single BLOB read (if BLOB handles are not supported) | 64KB |

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## Creating a DB2 Alias

Use the Alias Page to add, delete, or modify a DB2 alias.

**Alias Name** lists all the available aliases.

**New Alias** enables you to add a new alias.

**Delete Alias** enables you to delete any alias that is highlighted in the Alias Name box.

**Parameters** shows all the parameters with their current values.

**Description** briefly notes the purpose of the selected parameter.

**To add a new alias:**

1 Click the New Alias button. The Add New Alias dialog box appears.

2 Enter the new alias name and select the DB2 alias type. Then choose OK to begin the setup process. The Aliases page displays all the configuration parameters you can change to customize the new alias.

   The parameters are the same as those listed in the Drivers page for the DB2 driver type.

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

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

If you save the new alias in a configuration file with a different name, the BDE Configuration Utility displays a Save As dialog box:

- If you want to activate this configuration file next time you start your application, choose Yes.
- If you want to keep using the current default configuration file, choose No.

Your changes take effect the next time BDE is restarted.

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## Special SQL Links driver settings for DB2

The following topic describes driver configuration parameters unique to DB2. You set these parameters in the DB2 drivers page of the BDE Configuration Utility.

**DB2 DSN**

DB2 Data Source Name: the name of a connection to a DB2 database. This is a DB2 client alias for a database existing on the DB2 server. You create this alias by using the DB2 Client Setup utility.

Default: DB2_SERVER (the database created when DB2 is installed)

**DRIVER**

The name of the DB2 driver.

Default: IBM DB2 DRIVER

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## Troubleshooting DB2 connection problems

If you have problems establishing a DB2 connection with SQL Links,

1 Confirm that the DB2 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 Verify the connection at each layer.

4 Use the DB2 utility Client Setup to verify the connection.

If you are not connected, consult your database administrator.

If you are connected, continue with Step 5.

5 Verify that only the default values are used for the following parameters in the DB2CLI.INI file:

LONGDATACOMPAT = 0
[BLOB and CLOB columns should be treated as their native type]

AUTOCOMMIT = 1

GRAPHIC = 0

If these parameters are not set to these values, the DB2 driver's behavior is unpredictable.

6 Check whether your BDE application 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 BDECFG.EXE alias setup. Run the Configuration Utility and examine the alias.

For more information on your vendor-supplied diagnostic tools, see your DB2 documentation.

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## Borland language drivers for DB2

The following table lists language drivers available for use with SQL servers, 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.

| Long name | Short 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 |
| DB2 SQL ANSI | db2andeu | 1252 (ANSI) | Dictionary |
| 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 Links 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 |

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## DB2 data type translations

Certain database operations cause SQL Links to convert data from Paradox or dBASE format to DB2 format. For example, a BDE application that copies or appends local data to a DB2 table causes SQL Links to convert the local data to DB2 format before performing the copy or append operation.

Other database operations cause a conversion in the opposite direction, from DB2 format to Paradox or dBASE format. For example, suppose you run a local query against one or more remote tables. During the query, SQL Links converts any data originating in a SQL database to Paradox or dBASE format (depending on the type of answer required) before placing the data in the local answer table.

The following topics contain lists of DB2, Paradox, and dBASE data types and show how SQL Links translates between them:

- From DB2 data types
- From Paradox data types
- From dBASE data types
- From DB2 physical types

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

## From DB2 data types (to Paradox, dBASE)

| FROM: DB2 | TO: Paradox | TO: dBASE |
|---|---|---|
| BLOB | Binary | Memo |
| Char(n) | Alphanumeric(n) | Character(n) |
| Char(n) for bit data | Bytes | Memo |
| CLOB | Memo | Memo |
| Date | Date | Date |
| Decimal | Number | Number |
| DBCLOB | NOT SUPPORTED | |
| Double | Number | Number {20.4} |
| Float | Number | Number {20.4} |
| Integer | Long | Number {11.0} |
| Long VarChar | Memo | Memo |
| Long VarChar(n) for bit data | Binary | Memo |
| Numeric | Number | Number |
| SmallInt | Short | Number {6.0} |
| Time | Time | Character |
| TimeStamp | TimeStamp | Character |
| VarChar(n), n <=255 | Alphanumeric(n) | Character(n) |
| VarChar(n), n >255 | Memo | Memo |
| VarChar(n) for bit data | Binary | Memo |

**Note:** Although BDE supports NUMERIC and DECIMAL data types, inserting data for these types into a database on the DB2 server is not supported, pending a software update from IBM.

---

{button ,AL("db2translations")} DB2 data translations

## From Paradox data types (to DB2, dBASE)

| FROM: Paradox | TO: DB2 | TO: dBASE |
|---|---|---|
| Alphanumeric(n) | VarChar(n) | Character(n) |
| AutoInc | Integer | Number {11.0} |
| BCD | Double | Number {20.4} |
| Binary | long VarChar(n) for bit data | Memo |
| Bool | Char(1) | Bool |
| Bytes | Char(n) for bit data | Memo |
| Date | Date | Date |
| DateTime | TimeStamp | Character {>8} |
| Formatted memo | BLOB | Memo |
| Graphic | BLOB | Binary |
| Long | Integer | Number {11.0} |
| Memo | CLOB | Memo |
| Money | Double | Number {20.4} |
| Number | Double | Number {20.4} |
| OLE | BLOB | OLE |
| Short | Small int | Number {6.0} |
| Time | Time | Character {>8} |

**Note:** Although BDE supports NUMERIC and DECIMAL data types, inserting data for these types into a database on the DB2 server is not supported, pending a software update from IBM. Char(n) for bit data is binary. VarChar(n) for bit data is varbinary. long VarChar(n) for bit data is longvarbinary.

---

{button ,AL("db2translations")} DB2 data translations

## From dBASE data types (to DB2, Paradox)

| FROM: dBASE | TO: DB2 | TO: Paradox |
|---|---|---|
| Bool | Char(1) | Bool |
| Bytes | Char(n) for bit data | Bytes |
| Character(n) | VarChar(n) | Alphanumeric(n) |
| Date | Date | Date |
| Float | Double | Number |
| Lock | Char(24) | Alphanumeric {24.0} |
| Memo | CLOB | Memo |
| Number | Double | Number |
| OLE | BLOB | OLE |
| Binary | BLOB | Graphic |

**Note:** Although BDE supports NUMERIC and DECIMAL data types, inserting data for these types into a database on the DB2 server is not supported, pending a software update from IBM. Bytes in the dBASE driver is for temporary tables only.

---

{button ,AL("db2translations")} DB2 data translations

## From DB2 physical types (to BDE, Paradox, dBASE)

| DB2 physical | BDE logical | Paradox physical | dBASE physical |
|---|---|---|---|
| fldDB2BINARY | fldBYTES | fldPDXBYTES | fldDBMEMO |
| fldDB2BLOB | fldBLOB/fldstBINARY | fldPDXBINARYBLOB | fldDBMEMO |
| fldDB2CHAR | fldZSTRING | fldPDXCHAR | fldDBCHAR |
| fldDB2CLOB | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldDB2DATE | fldDATE | fldPDXDATE | fldDBDATE |
| fldDB2DBCLOB | NOT SUPPORTED | | |
| fldDB2DECIMAL | fldFLOAT | fldPDXNUM | fldDBFLOAT {20.4} |
| fldDB2DOUBLE | fldFLOAT | fldPDXNUM | fldDBNUM {20.4} |
| fldDB2FLOAT | fldFLOAT | fldPDXNUM | fldDBNUM {20.4} |
| fldDB2INT | fldINT32 | fldPDXLONG | fldDBNUM {11.0} |
| fldDB2LONGVARCHAR | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |
| fldDB2LONGVARBINARY | fldBLOB/fldstBINARY | fldPDXBINARYBLOB | fldDBMEMO |
| fldDB2NUMERIC | fldFLOAT | fldPDXNUM | fldDBFLOAT |
| fldDB2SMALLINT | fldINT16 | fldPDXSHORT | fldDBNUM {6.0} |
| fldDB2TIME | fldTIME | fldPDXTIME | fldDBCHAR |
| fldDB2TIMESTAMP | fldTIMESTAMP | fldPDXDATETIME | fldDBCHAR |
| fldDB2VARBINARY | fldVARBYTES | fldPDXBINARYBLOB fldDBMEMO | |
| fldDB2VARCHAR, n <= 255 fldZSTRING | fldPDXCHAR | fldDBCHAR | |
| fldDB2VARCHAR, n > 255 | fldBLOB/fldstMEMO | fldPDXMEMO | fldDBMEMO |

**Note:** Although BDE supports NUMERIC and DECIMAL data types, inserting data for these types into a database on the DB2 server is not supported, pending a software update from IBM.

---

{button ,AL("db2translations")} DB2 data translations

## DB2 field-naming rules

Here is a list of field-naming rules for Paradox, dBASE, and DB2.

| Naming rule | Paradox | dBASE | DB2 |
|---|---|---|---|
| Max length (chars) | 25 | 10 | 18 |
| Valid characters | All | All alphanumeric except punctuation marks, blank spaces, and other special characters. | All alphanumeric except punctuation marks, blank spaces, and other special characters. |
| Must begin with . . . | Any valid character except space | A letter | A letter |

**Note:** You may not use DB2-reserved words for remote table and column names. See the IBM *DATABASE 2 Client/Server SQL Reference*, Chapter 3 “Characters” and “Tokens” for reserved words.

---

{button ,AL("connectingtodb2")} Connecting to DB2 topics

{button ,AL("db2translations")} DB2 data translations