

ODBC Desktop Database Drivers

The following topics discuss the ODBC Desktop Database Drivers. This set of drivers consists of the Microsoft Access, Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, and Text drivers. Some information in the following topics is not applicable to all these drivers. When that is the case, the information is clearly marked as applying to one or more, but not all, of the drivers.

For All Users

The following topics discuss the ODBC Desktop Database Drivers and how to install them.

[Overview](#)

[Hardware and Software Requirements](#)

[Setting Up the ODBC Driver](#)

[Data Source Conversion](#)

[Adding, Modifying, and Deleting a Data Source](#)

[Connecting to a Data Source](#)

For Advanced Users

The following topics discuss how to use the ODBC Desktop Database Drivers directly.

[Connection Strings \(Advanced\)](#)

[SQL Statements \(Advanced\)](#)

[Data Types \(Advanced\)](#)

[Error Messages \(Advanced\)](#)

For Programmers

The following topics provide programming information on the ODBC Desktop Database Drivers. They are intended for application programmers and require knowledge of the Open Database Connectivity (ODBC) application programming interface (API).

[SQLGetInfo Return Values \(Programming\)](#)

[ODBC API Functions \(Programming\)](#)

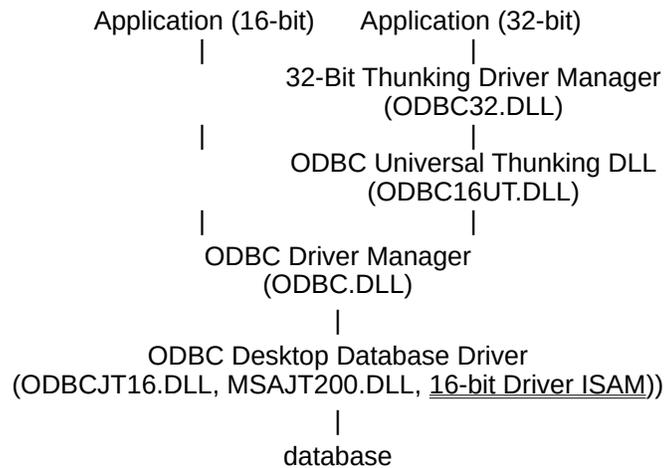
[Implementation Issues \(Programming\)](#)

Overview

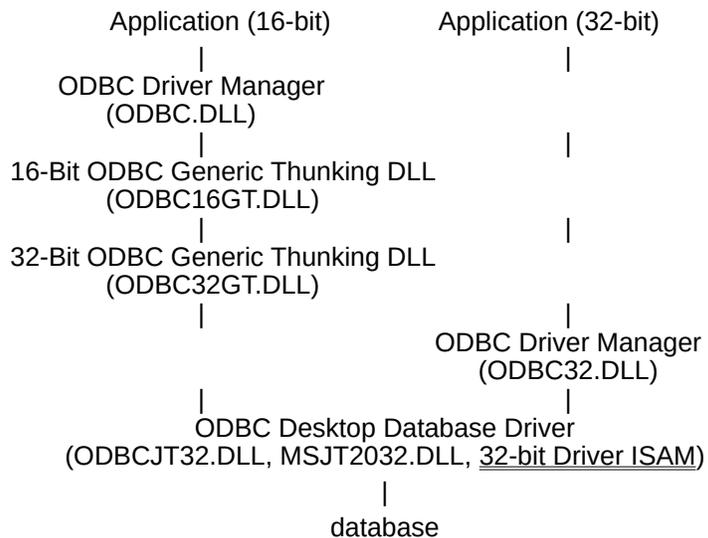
See Also

The ODBC Desktop Database Drivers allow you to open and query a database through the Open Database Connectivity (ODBC) interface.

The application/driver architecture for Win16/Win32s applications is:



The application/driver architecture for Win32 applications is:



Driver ISAMs

The ODBC Desktop Database Driver ISAMs are as follows:

| Driver | 16-bit ISAM | 32-bit ISAM | Applicability |
|-------------------------|---------------------------------|---------------------------------|---|
| Microsoft Access | None (Included in MSAJT200.DLL) | None (Included in MSJT2032.DLL) | Microsoft Access versions 1.0, 1.1, 2.0 |
| Btrieve | BTRV200.DLL | not available | Btrieve versions 5.1 and 6.x |
| dBase | XBS200.DLL | MSXB2032.DLL | dBase versions III and IV |
| Microsoft Excel | MSXL2016.DLL | MSXL2032.DLL | Microsoft Excel versions 3.0, 4.0, 5.0 (For versions earlier than 3.0, see note below) |
| Microsoft FoxPro | XBS200.DLL | MSXB2032.DLL | Microsoft FoxPro versions 2.0, 2.5, 2.6 |
| Paradox | PDX200.DLL | MSPX2032.DLL | Paradox versions 3.5, 4.0, 4.5 |
| Text | MSTX2016.DLL | MSTX2032.DLL | ASCII Text |

Note The Microsoft Excel driver is not able to read Microsoft Excel files whose format is earlier than version 3.0. These files can be opened in a later release of Microsoft Excel and saved in version 3, 4, or 5 format that the driver is able to read.

See Also

For All Users

[Adding, Modifying, and Deleting a Data Source](#)

[Connecting to a Data Source](#)

[Hardware and Software Requirements](#)

[Setting Up the ODBC Driver](#)

Hardware and Software Requirements

See Also

Hardware Requirements

To use the ODBC Desktop Database Drivers, you must have:

- An IBM-compatible personal computer with an 80386 or higher processor and a VGA or higher-resolution graphics card.
- A hard disk with 6MB of free disk space for 16-bit or 32-bit drivers.
- At least 4MB of random-access memory (RAM).

Software Requirements

To access data with the ODBC driver, you must have:

- The ODBC driver.
- The ODBC Driver Manager 2.0 (16-bit ODBC.DLL or 32-bit ODBC32.DLL).
- A computer running MS-DOS 3.3 or later. (MS-DOS 5.0 or later is recommended.)
- Microsoft Windows 3.1 or later to install and run 16-bit drivers, or Microsoft Windows NT 3.5 or later to install and run 32-bit drivers. (Note that 32-bit applications can be run on Win32s, and that 16-bit applications can be run on Windows on Windows (WOW) on Windows NT.)
- The stack size for an application using a Microsoft ODBC driver should be at least 20K.

When using Microsoft Windows NT 3.5 or later, the 32-bit driver is thread-safe, but only through the use of a global semaphore that controls access to the driver. Concurrent use of the driver will be very limited under Windows NT.

When running multiple 16-bit applications on Windows on Windows, the applications must be run in separate memory spaces. (The same memory space cannot be used because ODBC does not support multiple environments in the same process.) To run an application in a separate memory space, highlight the application's icon in the Microsoft Windows NT Program Manager, choose the File-Properties command, then choose Run in Separate Memory Space.

Driver-Specific Hardware and Software Requirements

- The **Microsoft Access, dBASE, and Microsoft FoxPro** driver may require changes in the AUTOEXEC.BAT or CONFIG.SYS files.
- The **Btrieve** driver requires Novell Btrieve for Windows (WBTRCALL.DLL).

CONFIG.SYS and AUTOEXEC.BAT Changes

If you are working with the **Microsoft Access** ODBC driver and do not have Microsoft Access installed on your computer, add the line "FILES= 50" to your CONFIG.SYS file and add the line "SHARE.EXE /L:200" to AUTOEXEC.BAT. If you are working with very large databases, add "SHARE.EXE /L:500" to AUTOEXEC.BAT.

Before you use the **dBASE** or **Microsoft FoxPro** driver, you must add the line "SHARE.EXE /L:200" to your AUTOEXEC.BAT file. The SHARE command allows file sharing and locking in a multitasking environment. The /L option specifies the number of files that can be locked at one time.

Btrieve for Windows DLL

The ODBC **Btrieve** driver requires the standalone Btrieve for Windows dynamic-link library (DLL), WBTRCALL.DLL. This file must be in the directory that contains Windows system DLLs (usually the C:\WINDOWS\SYSTEM directory). To check if you already have a copy of this file on your hard drive:

- 1 In the Main group, choose the File Manager icon.
- 2 From the File menu, choose Search.
The Search dialog box appears.
- 3 In the Search For box, type WBTRCALL.DLL.
- 4 In the Start From box, type C:\.
- 5 Choose the OK button.

If this file exists, copy it to the directory that contains Windows system DLLs (usually the C:\WINDOWS\SYSTEM directory).

Important If you already have a copy of WBTRCALL.DLL in your C:\WINDOWS\SYSTEM directory, contact your system administrator before copying over it. This file might be used by other programs on your computer.

If you don't have a copy of this file, contact Novell, Inc. at 1-800-453-1267 or your local software dealer. For more information about Btrieve files, see your Btrieve documentation.

See Also

For All Users

[Setting Up the ODBC Driver](#)

Setting Up the ODBC Driver

See Also

To set up the ODBC driver

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

Note If running 16-bit applications on Microsoft Windows NT 3.5, start the ODBC Administrator by double-clicking the ODBC Administrator icon in the Microsoft ODBC group.

- 2 In the Data Sources dialog box, choose the Drivers button.
- 3 In the Drivers dialog box, choose the Add button.
- 4 In the Add Driver dialog box, enter the name of the drive and directory containing the ODBC driver in the text box. Or choose the Browse button to select a drive and directory name.
- 5 Choose the OK button.
- 6 In the Install Drivers dialog box, choose the desired Data from the Available ODBC Drivers list.
- 7 Choose the OK button to install the driver.

Note The ODBC driver may share some of the same dynamic link libraries (DLLs) with other drivers installed on your computer. If so, you will be asked to overwrite the ODBC driver, regardless of whether it has been installed. Choose the Yes button to install the driver.

Before using the driver, you must add a data source for it. If the added driver replaces an already installed version 1.0 driver, you will have to run a data source conversion on the existing data sources.

Note If you have problems with your application after installing a version 2.0 ODBC driver and converting version 1.0 data sources to version 2.0 data sources, check the changes made in the version 2.0 drivers (What's New) to verify that none of the changes is causing the problems.

To delete the ODBC driver

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

Note If running 16-bit applications on Microsoft Windows NT 3.5, start the ODBC Administrator by double-clicking the Microsoft ODBC Administrator icon in the Microsoft ODBC group.

- 2 In the Data Sources dialog box, choose the Drivers button.
- 3 In the Drivers dialog box, select the ODBC driver from the Installed ODBC Drivers list.
- 4 Choose the Delete button.
The ODBC setup program asks if you want to remove the driver and all the data sources that use the driver.
- 5 Choose the Yes button.

See Also

For All Users

[Hardware and Software Requirements](#)

Data Source Conversion

If a version 2.0 driver replaces an installed version 1.0 driver, existing data sources associated with the version 1.0 driver will need to be converted before they can be used with the new driver. (Data sources to be converted can be either those associated with ODBC version 1.0 drivers, or those associated with the ODBC version 2.0 driver ODBCJT16.DLL, which was used with Microsoft Access 2.0 data sources.) This process converts all data source names (DSNs) associated with the version 1.0 driver to version 2.0 DSNs.

For Windows 3.1 or later, the keywords in the ODBC.INI files are changed. The ODBCINST.INI and ODBCISAM.INI files are also changed. For the **Microsoft Access 2.0** driver, the name of the setup DLL in ODBCINST.INI is also changed.

If you choose to install only 32-bit components, the data sources will be converted to 32-bit data sources; otherwise, the data sources will be converted to 16-bit data sources.

Converting to Version 2.0 DSNs

Data source conversion is performed when you run driver setup. For each new driver installed for which a version 1.0 driver was previously installed, you will be asked whether you want to convert the existing data sources associated with that driver. If so, choose Yes, and the conversion will be performed automatically.

When the **Text** driver is used, all converted Text DSNs will be given the same default format as the last converted DSN.

Reverting to Version 1.0 DSNs

On Windows 3.1 or later, data sources can be converted back to version 1.0. During the original conversion process, ODBC.INI, ODBCINST.INI, and ODBCISAM.INI are not deleted, but are renamed with .BAK extensions. If there are any versions of these files with BAK extensions, they are renamed with BA1 extensions; BA1 files are renamed BA2; and so on. To convert back to version 1.0, rename ODBC.BAK, ODBCINST.BAK, and ODBCISAM.BAK with INI extensions; and rename BA1, BA2, etc., back one level.

Adding, Modifying, and Deleting a Data Source

See Also

Before you can access data with an ODBC driver, you must add a data source for it. You can change or delete a data source at any time.

Note When the dBASE or Microsoft FoxPro driver is used, changes in the data source (DSN) setup for either driver will affect the DSN setup of the other driver. Changes to the dBASE DSN setup will change the Microsoft FoxPro DSN setup, and vice versa.

To add a data source

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

Note If running 16-bit applications on Microsoft Windows NT 3.5, start the ODBC Administrator by double-clicking the Microsoft ODBC Administrator icon in the Microsoft ODBC program group.

- 2 In the Data Sources dialog box, choose the Add button.
- 3 In the Add Data Source dialog box, select the driver's name from the Installed ODBC Drivers list and choose OK.
- 4 In the ODBC Setup dialog box, enter information to set up the data source. (Note that the Description entry cannot exceed 79 characters.)

To modify a data source

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

Note If running 16-bit applications on Microsoft Windows NT 3.5, start the ODBC Administrator by double-clicking the Microsoft ODBC Administrator icon in the Microsoft ODBC program group.

- 2 In the Data Sources dialog box, select the data source from the Data Sources list.
- 3 Choose the Setup button.
- 4 In the ODBC Setup dialog box, enter information to set up the data source. (Note that the Description entry cannot exceed 79 characters.)

To delete a data source

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

Note If running 16-bit applications on Microsoft Windows NT 3.5, start the ODBC Administrator by double-clicking the Microsoft ODBC Administrator icon in the Microsoft ODBC program group.

- 2 In the Data Sources dialog box, select the data source from the Data Sources list.
- 3 Choose the Delete button, and then choose the Yes button to confirm the deletion.

To add, modify, or delete a data source dynamically

You can call the SQLConfigDataSource API function to add, modify, or delete a data source dynamically. This function uses keywords to set connect options that in the above procedures are set through the Setup dialog box. This function should be used when you want to add, modify, or delete a data source without displaying the Setup dialog box.

See Also

For All Users

[Connecting to a Data Source](#)

[Setting Up the ODBC Driver](#)

data source

When the **Microsoft Access** driver is used, the data source specifies the Microsoft Access database (.MDB) you want to use and the information needed to get to that database.

When the **Btrieve, dBASE, Microsoft Excel 3.0/4.0, Microsoft FoxPro, Paradox, or Text** driver is used, the data source specifies the default data directory in which the ODBC driver searches for data files you want to access, as well as other information.

When the **Microsoft Excel 5.0** driver is used, the data source specifies the default data workbook in which the ODBC driver searches for data files you want to access, as well as other information.

ODBC Setup Dialog Box

See Also

The ODBC Setup dialog box contains the following fields:

| Control | Description |
|-----------------------|---|
| Approximate Row Count | <p>(This control is displayed only for the dBASE or Microsoft FoxPro driver.)</p> <p>Determines whether table size statistics are approximated. Note that this option applies to all data sources that use the ODBC driver.</p> |
| Buffer Size | <p>(This control is displayed only for the Microsoft Access driver.)</p> <p>The size of the internal buffer, in kilobytes, that is used by Microsoft Access to transfer data to and from the disk. The default buffer size is 256K (displayed as 256). Note that this option applies to all data sources that use the ODBC driver.</p> |
| Collating Sequence | <p>(This control is displayed only for the Btrieve, dBASE, Microsoft FoxPro, or Paradox driver.)</p> <p>The sequence in which the fields are sorted.</p> <p>When the Btrieve, dBASE, or Microsoft FoxPro driver is used, the sequence can be: ASCII or International.</p> <p>When the Paradox driver is used, the sequence can be: ASCII, International, Swedish-Finnish, or Norwegian-Danish.</p> |
| Data Source Name | <p>A name that identifies the <u>data source</u>, such as Payroll or Personnel.</p> |
| Database | <p>(This control is displayed only for the Microsoft Access driver.)</p> <p>The full path of the Microsoft Access database you want to access. Use the <u>Select</u> button to select an existing database. Use the <u>Create</u> button to create and select a new Microsoft Access database.</p> <p>A Microsoft Access data source can be set up without selecting or creating a database. If no database is provided upon setup, the user will be prompted to choose a database file when connecting to the data source.</p> <p>Use the <u>Repair</u> button to repair a damaged database. Use the <u>Compact</u> button to compact a database.</p> |
| Define Format | <p>(This control is displayed only for the Text driver.)</p> <p>Displays the <u>Define Text Format dialog box</u> and enables you to specify the <u>schema</u> for individual tables in the data source directory.</p> |
| Description | <p>An optional description of the data in the data source; for example, "Hire date, salary history, and current review of all employees."</p> |
| Directory | <p>Displays the currently selected directory. Before you add the data source, you must either use the Select Directory button to select a directory, or select the Use Current Directory check box to use the application's current working directory. (This does not apply to the Microsoft Access driver.)</p> <p>For Microsoft Excel 3.0/4.0 files, the path display is labeled "Directory", while for Microsoft Excel 5.0 files, the path display is labeled "Workbook".</p> |
| Exclusive | <p>(This control is displayed only for the Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, or Paradox driver.)</p> |

| | |
|-------------------|---|
| | <p>If the Exclusive box is selected, the database will be opened in Exclusive mode and can be accessed by only one user at a time. If the Exclusive box is cleared, the database will be opened in Shared mode and can be accessed by more than one user at a time. Performance is enhanced when running in Exclusive mode.</p> |
| Extensions List | <p>(This control is displayed only for the Text driver.)</p> <p>Lists the filename extensions of the text files on the data source. To use all files in the directory, select the Default (*.*) check box. To use only those files with certain extensions, clear the Default (*.*) check box and add each extension you want to use.</p> <p>To add an extension, type the extension in the Extension box and click the Add button. The extension must use the format *.xxx. For example, to use .DAT files, type the extension *.DAT. To remove an extension, highlight the extension in the Extensions list and click the Remove button. Removing all extensions is like checking the Default (*.*) check box: all files will be displayed.</p> |
| Network Directory | <p>(This control is displayed only for the Paradox driver.)</p> <p>The full path of the directory containing the PARADOX.NET file. If the directory does not contain a PARADOX.NET file, the Paradox driver creates one. For information about the PARADOX.NET file, see the Paradox documentation.</p> <p>Before you can select a network directory, you must enter your Paradox user name in the User Name text box. Use the Select Network Directory button to select a network directory.</p> |
| Page Timeout | <p>(This control is displayed only for the Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, or Paradox driver.)</p> <p>Specifies the period of time, in tenths of a second, that a page (if not used) remains in the buffer before being removed. The default is 600 tenths of a second (60 seconds). Note that this option applies to all data sources that use the ODBC driver.</p> |
| Read Only | Designates the database as read-only. |
| Rows to Scan | <p>(This control is displayed only for the Microsoft Excel or Text driver.)</p> <p>The number of rows to scan to determine the data type of each column. The data type is determined given the maximum number of kinds of data found. If data is encountered that does not match the data type guessed for the column, the data type will be returned as a NULL value.</p> <p>For the Microsoft Excel driver, you may enter a number from 1 to 16 for the rows to scan; however, the value will always default to 1. (A number outside the limit will return an error.)</p> <p>For the Text driver, you may enter a number from 1 to 32767 for the number of rows to scan; however, the value will always default to 1. (A number outside the limit will return an error.)</p> |
| Select Directory | <p>(This control is displayed only for the Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver.)</p> <p>Displays a dialog box where you can <u>select a directory</u> containing the files you want to access.</p> <p>For Microsoft Excel 3.0/4.0 files, the path selection button is labeled "Select Directory", while for Microsoft Excel 5.0 files, the path selection button is labeled "Select Workbook".</p> |
| Select Indexes | (This control is displayed only for the dBASE or Microsoft |

| | |
|-----------------------|---|
| | FoxPro driver.) |
| Show Deleted Rows | Displays the <u>Select Indexes dialog box</u> where you can associate dBASE or Microsoft FoxPro files with index files. (This control is displayed only for the dBASE or Microsoft FoxPro driver.) Specifies whether or not rows that have been marked as deleted should be displayed. Note that this option applies to all dBASE or Microsoft FoxPro data sources that use the ODBC driver. |
| Sort Order | (This control is displayed only for the Microsoft Access driver.) Sets a default sort order for the database. Can be: General (English, French, German, Portuguese, Italian), Spanish, Dutch, Swedish/Finnish, Norwegian/Danish, Icelandic, Czech, Hungarian, Polish, Russian, Turkish, Arabic, Hebrew, Greek. |
| System Database | (This control is displayed only for the Microsoft Access driver.) The full path of the Microsoft Access <u>system database</u> to be used with the Microsoft Access database you want to access. Select the None option button if no system database will be used. |
| Use Current Directory | (This control is displayed only for the Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver.) When selected, makes the application's current working directory the data source directory and disables the Select Directory option. When cleared, enables you to select the data source directory using the Select Directory option. The Use Current Directory control is deleted for Microsoft Excel 5.0 . |
| User Name | The Paradox user name. (This control is displayed only for the Paradox driver.) |
| Version | Enables you to select the version of the files used. |
| Workbook | For Microsoft Excel 5.0 files, displays the currently selected workbook. Before you add the data source, you must use the Select Workbook button to select a directory. Worksheets within workbooks are treated as SYSTEM TABLES. Defined or named ranges within a worksheet are treated as TABLES. |

Select Database

Enter the full path of an existing **Microsoft Access** database you want to access.

Create Database

Enter the full path of the new **Microsoft Access** database you want to create.

System Database

Select the System Database option button to select the system database to be used. The ODBC Microsoft Access driver prompts the user for a name and password. The default name is Admin and the default password in Microsoft Access for the Admin user is an empty string.

To increase the security of your Microsoft Access database, create a new user to replace the Admin user and delete the Admin user, or change the objects to which the Admin user has access.

Select the None option button to log into the Microsoft Access database as the Admin user; no system database will be used.

For more information about system databases and the Admin user, see the Microsoft Access documentation.

Select Directory

When defining a data source directory (for all drivers except **Microsoft Access**), specify the directory where your most commonly used files are located. The ODBC driver uses this directory as the default directory. Copy other files into this directory if they are used frequently. Alternatively, you can qualify filenames in a SELECT statement with the directory name:

```
SELECT * FROM C:\MYDIR\EMP
```

Or, you can specify a new default directory by using the **SQLSetConnectOption** function with the **SQL_CURRENT_QUALIFIER** option.

Paths for Additional Directories (Btrieve)

If you want to open Btrieve data files in directories other than the default, list these directories by the XTRPATH variable; for example, XTRPATH = path [; path [; path]]. This variable can be set in either of the following ways:

- Use the SET command in the AUTOEXEC.BAT file. For example, SET XTRPATH = C:\BTREIVE\DATA; F:\DATA.
- Use the Btrieve environment (.ENV) file. For other applications to recognize the Btrieve environment file, the file's path and filename must be specified by setting the XTRENV variable in the AUTOEXEC.BAT file (using the SET command).

The path for additional Btrieve directories can also be set in the application's initialization file. With the Microsoft Access application, for instance, the XTRPATH variable can be set in the [Btrieve ISAM] section of the ODBCDDP.INI file. If specifying the path in the Btrieve environment (.ENV) file, the path and filename of the environment file can be set in the MSACC20.INI file, rather than in AUTOEXEC.BAT.

Directory/Workbook Path Displays (Microsoft Excel)

For Microsoft Excel 3.0/4.0 files, the path display is labeled "Directory", and the path selection button is labeled "Select Directory". For Microsoft Excel 5.0 files, the path display is labeled "Workbook", and the path selection button is labeled "Select Workbook". When defining a data source directory, specify the directory where your most commonly used Microsoft Excel files are located for Microsoft Excel 3/4, or the directory where the workbook file is located for Microsoft Excel 5. The Use Current Directory control is deleted for Microsoft Excel 5.0.

See Also

For All Users

[Adding, Modifying, and Deleting a Data Source](#)

Select Network Directory

Select the full path of the directory containing the PARADOX.NET file. If the directory does not contain a PARADOX.NET file, the Paradox driver creates one. For information about the PARADOX.NET file, see the Paradox documentation.

Define Text Format

When the **Text** driver is used, the Define Text Format dialog box enables you to define the format for columns in a selected file. This dialog box enables you to specify the schema for each data table. This information is written to a SCHEMA.INI file in the data source directory. A separate SCHEMA.INI is created for each text data source directory.

Note The same default file format applies to all new text data tables. All files created by the CREATE TABLE statement inherit those same default format values, which are set by selecting file format values in the Define Text Format dialog box with <default> chosen in the Tables list box.

| Section | Information |
|--------------------|--|
| Characters | ANSI or OEM. OEM specifies a non-ANSI character set. |
| Column Name Header | Indicates whether the columns of the first row of the selected table are to be used as column names. Either TRUE or FALSE. |
| Columns | Contains a list of the column names for each column in the selected table. The order of the columns reflects the order of the columns in the table. |
| Data Type | Can be Char, Date, Float, or Integer. Date data types can be in the following formats: "dd-mmm-yy", "mm-dd-yy", "mmm-dd-yy", "yyyy-mm-dd", or "yyyy-mmm-dd". "mm" denotes numbers for months; "mmm" denotes letters for months. |
| Delimiter | Specifies the custom delimiter character to be used to separate columns. Enabled when the "Custom Delimiter" format is selected. Double quotations marks (") cannot be used as the delimiter character. (Note that the delimiter cannot be specified in hexadecimal or decimal format.) |
| Format | Either delimited or fixed length. If delimited, indicates the type of delimiter used: comma (CSV), tab, or special character (custom). If Format is fixed-length and Column Name Header is TRUE, the first line must be comma-delimited. |
| Guess | Automatically generates the column's data type, name, and width values for the columns in the selected table by scanning the table's contents according to the Format list box selection. Enabled when the table format is delimited. Any previously defined columns in the Columns list are cleared and replaced with new entries. If Column Name Header is not checked, column names are generated automatically as "F1", "F2", and so on. Note that no default value is shown in the Data Type box. |
| Name | Displays the name of the selected column. May be used to specify a new column name for either an existing column or a new column. If Column Name Header is TRUE, the column |

| | |
|--------------|--|
| | name displayed is ignored. |
| Rows to Scan | <p>The number of rows that Setup or the driver will scan when setting the columns and column data types based upon existing data. You may enter a number from 1 to 32767 for the number of rows to scan; however, the value will always default to 1. (A number outside the limit will return an error.)</p> |
| Tables | <p>Contains a list of all files in the directory selected in the Text Setup dialog box that match the list of extensions specified. Selecting <default> sets the table attributes for all files in the specified directory that are not explicitly defined. When <default> is selected, the "Columns" group is disabled.</p> |
| Width | <p>The width of the column may be changed for Char columns. For other data types, the width is displayed but cannot be changed.</p> |

schema

The schema includes information about each table (text file) in a data source, including the table's format, the number of rows to scan to determine column types, whether the first row of the table contains column names, whether the source file is written using an OEM or ANSI codepage, and each column's name, data type, and width.

SCHEMA.INI File

When the **Text** driver is used, the SCHEMA.INI file in the data source directory contains format information for the source text files. This information is used to specify text file format at data source setup time. Each text data directory has a separate SCHEMA.INI file. File format information entered in the Define Text Format dialog box is written to the SCHEMA.INI file, which is automatically created for this purpose.

Note The same default file format applies to all new text data tables. All files created by the CREATE TABLE statement inherit those same default format values, which are set by selecting file format values in the Define Text Format dialog box with <default> chosen in the Tables list box.

File formats in SCHEMA.INI have the following sections:

| Section | Information |
|------------------|---|
| ColNameHeader | Indicates whether the first record of data will specify the column names. Either TRUE or FALSE. |
| Format | Can be FIXEDLENGTH, TABDELIMITED, CSVDELIMITED (by a comma), or DELIMITED() (by the special character specified in the parentheses). The special character is one character in length and can be in character, decimal, or hexadecimal format. If Format is FIXEDLENGTH and ColNameHeader is TRUE, the first line must be comma-delimited. |
| MaxScanRows | The number of rows to be scanned when setting a column's data type based upon existing data. You may enter a number from 1 to 32767 for the number of rows to scan; however, the value will always default to 1. (A number outside the limit will return an error.) |
| Character Set | OEM or ANSI. |
| DateTimeFormat | Indicates date and time format. Entered as a string. Should be specified if all datetime fields in the SCHEMA.INI file will be handled with the same format. If this entry is absent, the short date picture and time options in the control panel will be used. |
| CurrencySymbol | Indicates the currency symbol to be used for currency values in the text file. If this entry is absent, the value in the control panel will be used. |
| CurrencyFormat | Indicates the format for the currency symbol and value. Can be "0" (currency symbol prefix with no separation, such as \$1), "1" (currency symbol suffix with no separation, such as 1\$), "2" (currency symbol prefix with one-character separation, such as \$ 1), or "3" (currency symbol suffix with one-character separation, such as 1 \$). |
| CurrencyDigits | Indicates the number of digits used for the fractional part of a currency amount. |
| CurrencyNegative | Indicates placement of the minus sign in currency format. Can be \$1, -\$1, \$-1, \$1-, |

| | |
|---------------------|--|
| Format | 1\$, -1\$, 1-\$, 1\$-, -1 \$, -\$ 1, 1 \$-, \$ 1-, \$ -1, 1-\$, \$ 1, or 1 \$. |
| DecimalSymbol | Can be any single character used to separate the integer from the fractional part of a number. |
| NumberDigits | Indicates the number of decimal digits in the fractional portion of a number. |
| NumberLeading Zeros | Indicates whether a decimal value less than 1 and greater than -1 should contain leading zeros. Can be either "0" (no leading zeros) or "1". |
| Col1, Col2,... | Column names, types, and width. If ColNameHeader is TRUE, the column names listed here are ignored. |

Text File Format (Advanced)

See Also

The ODBC **Text** driver supports both delimited and fixed-width text files. A text file consists of an optional header line and zero or more text lines.

Although the header line uses the same format as the other lines in the text file, the ODBC text driver interprets the header line entries as column names, not data.

A delimited text line contains one or more data values separated by delimiters: commas, tabs, or a custom delimiter. The same delimiter must be used throughout the file. Null data values are denoted by two delimiters in a row with no data between them. Character strings in a delimited text line can be enclosed in double quotation marks (" ").

The width of each data entry in a fixed-width text line is specified in a schema. Null data values are denoted by blanks.

The following grammar, written for programmers, defines the format of a text file that can be read by the ODBC text driver. Non-italics represent characters that must be entered as shown, italics represent arguments that are defined elsewhere in the grammar, brackets ([]) represent optional items, braces ({}), vertical bars (|) separate these choices, and ellipses (...) represent items that can be repeated one or more times.

The format of a text file is:

text-file ::=

*[delimited-header-line] [delimited-text-line]... end-of-file |
[fixed-width-header-line] [fixed-width-text-line]... end-of-file*

delimited-header-line ::= delimited-text-line

delimited-text-line ::=

*blank-line |
delimited-data [delimiter delimited-data]... end-of-line*

fixed-width-header-line ::= fixed-width-text-line

fixed-width-text-line ::=

*blank-line |
fixed-width-data [fixed-width-data]... end-of-line*

end-of-file ::= <EOF>

blank-line ::= end-of-line

delimited-data ::= delimited-string | number | date | delimited-null

fixed-width-data ::= fixed-width-string | number | date | fixed-width-null

The width of each column in a fixed width text file is specified in the SCHEMA.INI file.

end-of-line ::= <CR> | <LF> | <CR><LF>

delimited-string ::= unquoted-string | quoted-string

unquoted-string ::= [character | digit] [character | digit | quote-character]...

quoted-string ::=

quote-character

*[character | digit | delimiter | end-of-line | embedded-quoted-string]...
quote-character*

*embedded-quoted-string ::=
quote-character quote-character
[character | digit | delimiter | end-of-line]
quote-character quote-character*

fixed-width-string ::= [character | digit | delimiter | quote-character] ...

*character ::= any character except:
delimiter
digit
end-of-file
end-of-line
quote-character*

digit ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

delimiter ::= , | <TAB> | custom-delimiter

*custom-delimiter ::= any character except:
end-of-file
end-of-line
quote-character*

The delimiter in a custom-delimited text file is specified in the SCHEMA.INI file.

quote-character ::= "

number ::= exact-number | approximate-number

*exact-number ::= [+ | -] {unsigned-integer[.unsigned-integer] |
unsigned-integer. |
.unsigned-integer*

approximate-number ::= exact-number{e | E}[+ | -]unsigned-integer

unsigned-integer ::= {digit}...

*date ::=
mm date-separator dd date-separator yy |
mmm date-separator dd date-separator yy |
dd date-separator mmm date-separator yy |
yyyy date-separator mm date-separator dd |
yyyy date-separator mmm date-separator dd*

mm ::= digit [digit]

dd ::= digit [digit]

yy ::= digit digit

yyyy ::= digit digit digit digit

mmm ::= Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec

date-separator ::= - | / | .

delimited-null ::=

For delimited files, a NULL is represented by no data between two delimiters.

fixed-width-null ::= <SPACE>...

For fixed width files, a NULL is represented by spaces.

See Also

For All Users

[Adding, Modifying, and Deleting a Text Data Source](#)

Select Indexes

When the **dBASE** or **Microsoft FoxPro** driver is used, the Select Indexes dialog box contains the following fields.

Tables

Displays a list of the files in the currently selected data source.

Indexes

Displays the indexes assigned to the currently selected file in the Tables list.

List Files of Type

Displays and allows you to choose the types of files to display in the Indexes list.

To associate an index with a table

- 1 From the Tables list, select a file.
- 2 From the Indexes list, select an index.
- 3 Choose the OK button to save the table/index associations.

Note dBASE III indexes or Microsoft FoxPro 2.0 indexes must be assigned using this dialog box for the driver to recognize them.

Note The ODBC dBASE and Microsoft FoxPro drivers do not support Clipper .NTX index files.

Select Unique Record Identifier

Select the column(s) that uniquely identify a record for a table.

Connecting to a Data Source

See Also

When you connect to a data source, an application may prompt you to enter the name of a directory. If you are prompted, enter or select the directory containing the files you want to access.

Password Entry

Passwords must be entered at connection time, either by password in the connection string (UID and PWD for **Microsoft Access** or PWD for **Paradox**), or with the *szAuthStr* argument in the **SQLConnect** function.

When the **Microsoft Access** driver is used, if the data source uses a system database, an application may prompt you for your name and password. The default name is Admin and the default password in Microsoft Access for the Admin user is an empty string.

Heterogeneous Joins/Attached Tables

When the **Microsoft Access** driver is used, attached tables and heterogeneous joins are supported.

.DDF Files

When the **Btrieve** driver is used, make sure that .DDF files contain information for all tables, new and existing, in the data source directory you specify. If .DDF files do not exist when a data source is connected to, .DDF files will be created automatically at connection time. However, these new .DDF files will only contain information about tables created after the .DDF files themselves were created. They will not contain information about existing tables.

Opening Microsoft Excel Tables

The **Microsoft Excel** driver is not able to read Microsoft Excel files whose format is earlier than version 3.0. These files can be opened in a later release of Microsoft Excel and saved in version 3, 4, or 5 format that the driver is able to read.

Microsoft Excel 3 (BIFF3) and Microsoft Excel 4 (BIFF4) file formats must have a database range defined. The Microsoft Excel driver recognizes named ranges as tables. Databases specified by a named range will increase as new data is added. If any new data would overwrite existing data in a worksheet (outside the named range that represents the data), an error will be returned.

All Microsoft Excel tables (spreadsheets) that are created and opened for inserting are opened exclusive, and can be opened by only one user at a time. Otherwise, Microsoft Excel tables are opened shared. "Read Only" allows multiple users to view .xls files.

Password-Protected Files

The **Btrieve** driver cannot access data in password-protected Btrieve files.

When the **Paradox** driver is used, after a password-protected file has been opened by a user, other users are not allowed to open the same file.

Referential Integrity

Referential integrity is supported only for a **Microsoft Access** data source, and only if the referential integrity is set up for the data source by the **Microsoft Access** database management system.

See Also

For All Users

[Adding, Modifying, and Deleting a Data Source](#)

For Advanced Users

[Connection Strings \(Advanced\)](#)

Connection Strings (Advanced)

See Also

The connection string for the ODBC drivers uses the following keywords:

| <u>Keyword</u> | <u>Description</u> |
|----------------|---|
| DSN | Name of the data source |
| DBQ | Name of the directory |
| FIL | File type (MS Access for Microsoft Access; BTRIEVE; DBASE3 or DBASE4; EXCEL for Microsoft Excel; FOXPRO 2.0, 2.5, or 2.6 for Microsoft FoxPro; PARADOX; or TEXT) |

Driver-Specific Keywords

In addition to the keywords shown above, the **Microsoft Access** driver uses UID and PWD keywords and the **Paradox** driver uses the PWD keyword.

| <u>Keyword</u> | <u>Description</u> |
|----------------|--|
| UID | The user login ID (Microsoft Access only) |
| PWD | The user-specified password. |

Note When the **Microsoft Access** or **Paradox** driver is used, the PWD keyword should not include any of the special characters (see SQL_SPECIAL_CHARACTERS in [SQLGetInfo Return Values](#)).

See Also

For All Users

[Connecting to a Data Source](#)

Select Database (Advanced)

When the **Microsoft Access** driver is used, the Select button may be pressed in the Microsoft Access 2.0 Setup dialog box to select an existing database.

The Open Database dialog box contains the following fields:

| Control | Description |
|----------------|--|
| File Name | Name of the existing database file with an .MDB extension. |
| Exclusive | If the Exclusive box is selected, the database will be opened in Exclusive mode and can be accessed by only one user at a time. If the Exclusive box is cleared, the database will be opened in Shared mode and can be accessed by more than one user at a time. Performance is enhanced when running in Exclusive mode. |
| Read Only | Designates the database as read-only. |

Create Database (Advanced)

When the **Microsoft Access** driver is used, the Create button may be pressed in the Microsoft Access 2.0 Setup dialog box to create a new database.

The New Database dialog box contains the following fields:

| Control | Description |
|----------------|--|
| File Name | Name of the database file with an .MDB extension. |
| Exclusive | If the Exclusive box is selected, the database will be opened in Exclusive mode and can be accessed by only one user at a time. If the Exclusive box is cleared, the database will be opened in Shared mode and can be accessed by more than one user at a time. Performance is enhanced when running in Exclusive mode. |
| Sort Order | Sets a default sort order for the database. Can be: General (English, French, German, Portuguese, Italian), Spanish, Dutch, Swedish/Finnish, Norwegian/Danish, Icelandic, Czech, Hungarian, Polish, Russian, Turkish, Arabic, Hebrew, Greek. |

Database Repair (Advanced)

When the **Microsoft Access** driver is used, if a client failure occurs during the commit process (after **SQLTransact** has been issued, but before the function returns), the affected database can be corrupted and need to be repaired. Repair can be performed either upon setup through the Microsoft Access driver setup dialog box, or programmatically by using the **SQLConfigDataSource** function.

To repair the database through the Microsoft Access driver setup dialog box, click the Repair button. Select the file and path, and then click OK.

To repair the database programmatically, include "REPAIR_DB=<path name>" in the list of attributes (*IpszAttributes*) of the **SQLConfigDataSource** function. In this function, <path name> is the full path to the Microsoft Access database.

When repairing a password-protected file, a password and user id must be entered in the advanced options dialog box (or in the list of attributes for **SQLConfigDataSource**, if repairing programmatically). The password and user id should be removed after the repair.

These operations will run the standard Microsoft Access repair process on the database. Data in Microsoft Access databases could be affected by either of these actions.

Note that a database indicated as corrupted may in fact not be corrupted. The indication may be prompted by an improper shutdown that does not clear the appropriate bits. In this and other cases, a programmer may want to keep the corruption message box transparent to the end user. This can be accomplished by passing SQL_DRIVER_NOPROMPT with the SQLDriverConnect function.

For more information about database repair in Microsoft Access, refer to the Microsoft Access help file.

Database Compaction (Advanced)

When the **Microsoft Access** driver is used, data compaction can be performed on a database file either upon setup through the Microsoft Access driver setup dialog box, or programmatically by using the **SQLConfigDataSource** function.

To perform data compaction on a database file through the Microsoft Access driver setup dialog box, click the Compact button. From the Database to Compact From dialog box, select the file and path, and then click OK. From the Database to Compact Into dialog box, select the file and path, and a sort order, and then click OK. The name of the file to compact into can be the same as the file to be compacted.

To perform data compaction on a database file programmatically, include "COMPACT_DB=<source db> <dest db> <sort order>" in the list of attributes (*lpzAttributes*) of the **SQLConfigDataSource** function. In this function, <source db> is the full path to the Microsoft Access database to be compacted and <dest db> is the full path for the compacted Microsoft Access database.

When compacting a password-protected file, a password and user id must be entered in the advanced options dialog box (or in the list of attributes for **SQLConfigDataSource**, if compacting programmatically). The password and user id should be removed after the compaction.

These operations will run the standard Microsoft Access compaction process on the database.

For more information about database compaction in Microsoft Access, refer to the Microsoft Access help file.

SQL Statements (Advanced)

The Microsoft ODBC Desktop Database Drivers support most [SQL statements](#) and clauses in the ODBC minimum grammar. While the driver supports the grammar for reading data, it has limitations for writing data.

For information about SQL grammar limitations, and additional and driver-specific grammar supported, see the following topics:

For Advanced Users

-  [Additional Supported ODBC SQL Grammar](#)
-  [Driver-Specific SQL Grammar](#)
-  [Limitations to SQL Grammar](#)

Additional Supported ODBC SQL Grammar (Advanced)

See Also

The ODBC drivers completely support the Minimum ODBC grammar and the following clauses in the Core and Extended ODBC grammar:

| Core and Extended grammar | Comments |
|--|--|
| Approximate numeric literal (Core) | Supported. |
| ALTER TABLE statement (Core) | When the Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, or Paradox driver is used, the ALTER TABLE statement is supported with add and drop table constraint definition clauses. (ALTER TABLE statements are not supported for the Microsoft Excel or Text driver.) |
| Binary Literals (Extended) | Supported. |
| BETWEEN predicate (Core) | Supported. |
| Correlation names (Core) | Correlation names are fully supported, including within the table list. For example, in the following string, E1 is the correlation name for the table named Emp: <pre>SELECT * FROM Emp E1 WHERE E1.LastName = 'Smith'</pre> |
| EXISTS predicate (Core) | Supported. |
| IN (<i>valuelist</i>) (Core) | Supported. |
| Procedure Invocation (Extended) | When the Microsoft Access driver is used, procedures can be invoked from the driver by using the SQLExecDirect or SQLPrepare function with the following syntax: {CALL <i>procedure-name</i> [(<i>parameter</i> [, <i>parameter</i>]...)]}. Note that expressions are not supported as parameters to a called procedure. If a procedure name includes a dash, the name must be delimited with backquotes (`). A parameterized query can be called using the above statement. A parameterized query cannot be used in a SELECT statement: SELECT * FROM <QueryName> WHERE <Column> = ?. |
| SELECT statement (Extended) (including GROUP BY, HAVING, and UNION clauses) | Supported. |
| Subqueries (SUBSELECTs) (Core) | Supported. |

See Also

For Advanced Users

[Driver-specific ODBC SQL Grammar Supported \(Advanced\)](#)

[Limitations to ODBC SQL Grammar \(Advanced\)](#)

Driver-Specific SQL Grammar Supported (Advanced)

See Also

The Microsoft ODBC drivers support the following driver-specific SQL grammar:

| Driver-specific SQL grammar | Comments |
|------------------------------------|--|
| BETWEEN predicate | <p>The syntax:</p> <p><i>expression1</i> BETWEEN <i>expression2</i> AND <i>expression3</i></p> <p>returns True only if <i>expression1</i> is greater than or equal to <i>expression2</i> and <i>expression1</i> is less than or equal to <i>expression3</i>.</p> |
| COUNT(<i>expression</i>) | <p>Counts all non-NULL values for an expression across a predicate. This function behaves like other set functions, such as SUM, AVG, MIN, and MAX. For example:</p> <pre>SELECT COUNT (A+B) FROM Q</pre> <p>counts all the rows in Q where A+B does not equal NULL.</p> <p>When the Paradox driver is used, the COUNT function includes NULL as well as non-NULL values in the count.</p> |
| CREATE INDEX | <p>The syntax of the CREATE INDEX statement is: CREATE [UNIQUE] INDEX <i>index-name</i> ON <i>table-name</i> (<i>column-identifier</i> [ASC] [DESC][, <i>column-identifier</i> [ASC][DESC]...]) WITH <<i>index option list</i>> where <<i>index option list</i>> can be: PRIMARY DISALLOW NULL IGNORE NULL</p> <p>Only the Microsoft Access driver uses the DISALLOW NULL and IGNORE NULL index options. The Btrieve, dBASE, Microsoft FoxPro, and Paradox drivers accept the syntax, but ignore the presence of either option.</p> <p>When the Paradox driver is used, the <u>CREATE INDEX</u> statement creates Paradox primary key files and secondary files.</p> |
| Date arithmetic | <p>The driver supports adding and subtracting an integer from a DATE column. The integer specifies the number of days to add or subtract.</p> |
| Date literals | <p>The MM-DD-YYYY format is supported.</p> |
| DROP INDEX | <p>When the Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, or Paradox driver is used, the syntax of the DROP INDEX statement is "DROP INDEX a on b" where "a" is the name of the index and "b" is the name of the table (not DROP INDEX <i>index-name</i>).</p> <p>When the Paradox driver is used, the DROP INDEX statement deletes Paradox secondary index files and does not use the ODBC SQL grammar.</p> <p>The DROP INDEX statement is not supported for the Microsoft Excel or Text driver.</p> |
| Fixed-Width Text File | <p>When the Text driver is used, the last column</p> |

| | |
|---------------------------------|---|
| GROUP BY <i>expression-list</i> | <p>of a fixed-width text file can be variable length. GROUP BY supports an expression list as well as a column name. The select list of a SELECT statement that has a GROUP BY clause can only include expressions from the GROUP BY clause or set functions.</p> |
| ORDER BY <i>expression-list</i> | <p>No ordering is allowed on Set functions or an expression that contains a Set function. For example, in the following clauses the table is ordered by three key expressions: a+b, c+d, and e.</p> |
| ORDER BY with GROUP BY | <pre data-bbox="841 537 1128 600">SELECT * FROM emp ORDER BY a+b, c+d, e</pre> <p>ORDER BY can be performed on any expression in the GROUP BY <i>expression-list</i> or any column in the result set.</p> |
| Outer joins | <p>A SELECT statement can contain a list of OUTER JOIN clauses. The OUTER JOIN syntax is extended to support nested OUTER JOINS. For more information on OUTER JOINS, refer to the documentation for the applicable Database Management System.</p> |
| Scalar functions | <p><u>Scalar functions</u> are supported.</p> |
| Table names | <p>When the Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver is used, table names that occur in the FROM clause of SELECT or DELETE, after the INTO clause in INSERT, and after UPDATE, CREATE TABLE, and DROP TABLE can contain a valid path, primary name, and filename extension.</p> <p>Use of a table name elsewhere in an SQL statement does not support the use of paths or extensions but will accept only the primary name (for example, EMP FROM C:\ABC\EMP).</p> <p>Correlation names (aliases) can be used. For example:</p> <pre data-bbox="841 1356 1128 1442">SELECT * FROM C:\ABC\EMP T1 WHERE T1.COL1 = 'aaa'</pre> |

See Also

For Advanced Users

[Additional Supported ODBC SQL Grammar \(Advanced\)](#)

[Limitations to ODBC SQL Grammar \(Advanced\)](#)

Limitations to SQL Grammar (Advanced)

See Also

The Microsoft ODBC drivers impose the following limitations on the SQL grammar:

| Grammar | Limitation |
|------------------------|---|
| Aggregate functions | An aggregate function and a non-aggregate column reference cannot both be used as arguments to a single SQL statement. |
| ALTER TABLE statement | When the Btrieve, dBASE, Microsoft FoxPro, or Paradox driver is used, once an index has been created and a new record added, the structure of the table cannot be changed by the ALTER TABLE statement unless the index is dropped and the contents of the table are deleted. |
| AND predicates | A maximum of 40 supported. |
| CALL statement | Expressions are not supported as parameters to a called procedure. (Microsoft Access only) |
| Column names | <p>Column names can contain any valid characters (for example, spaces). If column names contain any characters except letters, numbers, and underscores, the name must be delimited by enclosing it in back quotes (`).</p> <p>When the Microsoft Access or Microsoft Excel driver is used, column names are limited to 64 characters, and longer names generate an error. When the Paradox driver is used, the maximum column name is 25 characters. When the Text driver is used, the maximum column name is 64 characters, and longer names are truncated.</p> <p>When the Microsoft Access driver is used, the driver expects the Microsoft Access database to have column names in the first row of the table.</p> <p>When the dBASE or Microsoft FoxPro driver is used, characters with an ASCII value greater than 127 are converted to underscores.</p> <p>When the Microsoft Excel driver is used, the "\$" character is used in place of the "!" character, which is not legal.</p> <p>When the Text driver is used, the driver provides a default name if a column name is not specified. For example, the driver calls the first column F1, the second column F2, and so on.</p> |
| CONVERT function | <p>Type conversion failures result in the affected column being set to NULL.</p> <p>Neither the DATE nor TIMESTAMP data type can be converted to another data type (or itself) by the CONVERT function.</p> |
| CREATE INDEX statement | <p>Not supported for the Microsoft Excel or Text driver.</p> <p>An index can be defined on a maximum of 10</p> |

columns. If more than 10 columns are included in a CREATE INDEX statement, the index will not be recognized and the table will be treated as though no index were created.

The **dBASE** driver cannot create an index on a LOGICAL column.

When the **dBASE** or **Microsoft FoxPro** driver is used, response time on large files can be improved by building an .MDX (or .NDX) index on the column (field) specified in the WHERE clauses of a SELECT statement.

Existing .MDX indexes will automatically be applied for =, >, <, >=, <=, and BETWEEN operators in a WHERE clause, and LIKE predicates, as well as in join predicates.

When the **dBASE** or **Microsoft FoxPro** driver is used, the index created by a CREATE UNIQUE INDEX statement is actually non-unique, and duplicate values can be inserted into the indexed column.

When the **Paradox** driver is used, a unique index must be defined upon the first "n" columns in a table. A table cannot be updated by the Paradox driver if a unique index is not defined on the table. (Note that this is not true for an empty table, which can be updated even if a unique index is not defined on the table.)

CREATE TABLE statement

When the **Microsoft Access**, **Btrieve**, **Microsoft Excel**, or **Paradox** driver is used, and the length of a text or binary column is not specified (or is specified as 0), the column length will be set to 255.

When the **dBASE** or **Microsoft FoxPro** driver is used, and the length of a text or binary column is not specified (or is specified as 0), the column length will be set to 254.

A maximum of 255 columns is supported.

When the **Microsoft Excel5** driver is used, a worksheet cannot be created with the same name as a worksheet that was previously dropped. The DROP TABLE statement clears the worksheet, but does not delete the worksheet name.

Date Arithmetic

Date arithmetic is not supported for subtracting a DATE data type from a DATE data type.

DELETE statement

Not supported for the **Text** driver. Note that the INSERT statement is supported for the Text driver.

For the **Paradox** driver to delete a row from a table, the table must have a unique index (Paradox primary key).

DISTINCT keyword

Not supported for Long Text fields (**Microsoft Access**) or Memo fields (**dBASE** or **FoxPro**).

DROP TABLE

When the **Microsoft Excel5** driver is used,

| | |
|------------------|--|
| statement | the DROP TABLE statement clears the worksheet, but does not delete the worksheet name. The worksheet name still exists in the workbook, so another worksheet cannot be created with the same name. |
| FROM clause | The maximum number of tables in a FROM clause is 16. |
| HAVING clause | The maximum number of search conditions in a HAVING clause is 40. |
| Identifiers | If an identifier contains a space or a special symbol, the identifier must be enclosed in back quotes. A valid name is a string of no more than 64 characters, of which the first character must not be a space. Valid names cannot include control characters or the following special characters: ` # * ? [] . ! \$. Do not use reserved words listed in the SQL grammar in Appendix C of the <i>Microsoft ODBC Programmer's Reference</i> as identifiers (that is, table or column names), unless you surround the word in back quotes (`). |
| Index names | When the Btrieve driver is used, an index cannot have the same name as any of the column names in the table upon which the index is defined. Only 24 index "segments" are allowed per table. When the Paradox driver is used, a primary <u>index</u> must have the same name as the table upon which it is defined. Other unique or non-unique indexes must have the same name as the table upon which they are defined. |
| INSERT statement | Inserted data is truncated on the right without warning if it is too long to fit into the column. If a column reference is left out of an INSERT statement, a parameter size error is returned instead of a syntax error. Attempting to insert a value that is out of the range of a column's data type causes a NULL to be inserted into the column. A table is not updateable by the Paradox driver if a unique index is not defined on the table. (Note that this is not true for an empty table, which can be updated even if a unique index is not defined on the table.) When the Text driver is used, NULL values are represented by a blank padded string in fixed-length files, but are represented by no spaces in delimited files. For example, in the following row containing three fields, the second field is a NULL value: "Smith:,, 123 When the Text driver is used, all column values can be padded with leading spaces. The length of any row must be less than or equal to 65,543 bytes. |

| | |
|-----------------------|--|
| LIKE predicate | <p>If data in a column is longer than 255 characters, the LIKE comparison will be based only on the first 255 characters.</p> <p>A LIKE used in a procedure is supported only with constant patterns.</p> <p>Use of an escape clause in a LIKE predicate is not supported.</p> |
| NOT NULL | The NOT NULL constraint in the CREATE TABLE statement is not supported. |
| ORDER BY | If a SELECT statement contains a GROUP BY clause and an ORDER BY clause, the ORDER BY clause can contain only a column in the result set or an expression in the GROUP BY clause. |
| Parameterized Queries | A parameterized query can be called using the following syntax: <code>CALL query-name [(parameter[,parameter]...)]</code> . A parameterized query cannot be used in a SELECT statement: <code>SELECT * FROM query-name WHERE column = ?</code> . |
| Scalar Functions | Scalar functions are supported only by using the ODBC canonical format. |
| SELECT DISTINCT | The DISTINCT keyword does not apply to binary data. |
| SELECT statement | <p>An aggregate-function column cannot be mixed with a non-aggregate column in a SELECT statement.</p> <p>The select list of a SELECT statement that has a GROUP BY clause can only have expressions from the GROUP BY clause or set functions.</p> <p>The use of an asterisk (to select all columns) in a SELECT statement containing a GROUP BY clause is not supported. The names of the columns to be selected must be specified.</p> <p>The use of a vertical bar in a SELECT statement is not supported.</p> <p>When using a column alias in a SELECT statement, the word "as" must precede the alias. For example, "SELECT col1 as a from b". Without the "as", the statement will return an error.</p> <p>If an incorrect column name is entered into a SELECT statement, a SQLSTATE 07001 error, "Wrong Number of Parameters", is returned instead of a SQLSTATE S0022 error, "Column Not Found".</p> |
| Set Functions | The set functions (AVG, MAX, MIN, and SUM) do not support the DISTINCT keyword. |
| Sorting | The maximum length of a sort key in a GROUP BY clause, ORDER BY clause, SELECT DISTINCT statement, or outer join is 255 bytes; the maximum length of all sort keys in a sort row is 65,500 bytes. |
| Strings | The maximum length of an SQL statement |

| | |
|-------------------------|---|
| | string is 65,000 characters. |
| | When the Microsoft Access driver is used, only ANSI string constants (with single quotation marks, not double quotation marks) are supported. |
| Table names | <p>Table names can contain any valid characters (for example, spaces). If table names contain any characters except letters, numbers, and underscores, the name must be delimited by enclosing it in back quotes (`).</p> <p>When the Microsoft Excel driver is used, and a table name is not qualified by a database reference, the default database is implied. The '\$' character is used in place of the '!' character, which is not legal.</p> <p>The Microsoft Excel table name that references <filename> is supported for Microsoft Excel 3.0 and 4.0 files. The Microsoft Excel table name that references <workbook-name> is supported for Microsoft Excel 5.0 files.</p> <p>When the dBASE or Microsoft FoxPro driver is used, characters with an ASCII value greater than 127 are converted to underscores.</p> <p>When the Microsoft Access driver is used, the table name is limited to 64 characters. Special DOS keywords CON, AUX, LPT1, and LPT2 should not be used as table names.</p> |
| Table references | <p>A maximum of 16 table references can be included in any query statement. For 16-bit drivers, however, table references may be limited to a number less than 16, especially for highly complex queries, because of the compilation segment limit.</p> <p>For 32-bit drivers, the limit of 16 table references is always valid.</p> |
| UPDATE statement | For the Paradox driver to update a table, the table must have a unique index (Paradox primary key). |
| Views | Not supported by the Btrieve , dBASE , Microsoft Excel , Microsoft FoxPro , Paradox , or Text driver. |
| WHERE clause | <p>The maximum number of clauses in a WHERE clause is 40.</p> <p>LONGVARBINARY and LONGVARCHAR columns can be searched for using literals up to 255 characters in length, but not using parameters.</p> |
| WHERE CURRENT OF clause | Not supported. |

See Also

For Advanced Users

[Additional Supported ODBC SQL Grammar \(Advanced\)](#)

[Driver-Specific ODBC SQL Grammar Supported \(Advanced\)](#)

dBASE Indexes (Advanced)

The ODBC **dBASE** driver automatically opens and updates dBASE IV index files. You must use the ODBC Control Panel option (or ODBC Administrator program if you're running Windows 3.1 or later) to associate dBASE III .NDX files with dBASE files.

The following limitations apply to the creation of dBASE indexes:

- All column names must be valid.
- All columns must be in the same ascending or descending order.
- The length of any single text column must be less than 100 bytes.
- If more than one column exists, all of the columns must be text columns and the sum of the column sizes must be less than 100 bytes.
- Memo fields cannot be indexed.
- An index must not be specified for the current set of fields (that is, duplicate indexes are not allowed).
- The index name must match the dBASE index naming convention. dBASE III requires that each index be in a separate file, each having an .NDX extension. In dBASE IV, indexes are created as tag names that are stored in a single .MDX file. The .MDX file has the same base name as the database file (for example, EMP.MDX is the index file for the EMP.DBF database).

Microsoft FoxPro Indexes (Advanced)

The ODBC Microsoft **FoxPro** driver automatically opens and updates Microsoft FoxPro 2.5/2.6 index files. You must use the ODBC Control Panel option (or ODBC Administrator program if you're running Windows 3.1 or later) to associate Microsoft FoxPro 2.0 .IDX files with Microsoft FoxPro files.

The following limitations apply to the creation of Microsoft FoxPro indexes:

- All column names must be valid.
- All columns must be in the same ascending or descending order.
- The length of any single text column must be less than 100 bytes.
- If more than one column exists, all columns must be text columns and the sum of the column sizes must be less than 100 bytes.
- Memo fields cannot be indexed.
- An index must not be specified for the current set of fields (that is, duplicate indexes are not allowed).
- The index name must match the Microsoft FoxPro index naming convention. Microsoft FoxPro 2.0 requires that each index be in a separate file, each having an .IDX extension. In Microsoft FoxPro 2.5/2.6, indexes are created as tag names that are stored in a single .CDX file. The .CDX file has the same base name as the database file (for example, EMP.CDX is the index file for the EMP.DBF database).

CREATE INDEX (Advanced)

The syntax of the CREATE INDEX statement for the ODBC **Paradox** driver is:

```
CREATE [UNIQUE] INDEX index-name
  ON table-name
  (column-identifier [ASC]
  [, column-identifier [ASC]...])
```

The ODBC Paradox driver does not support the **DESC** keyword in the ODBC SQL grammar for the CREATE INDEX statement. The *table-name* argument can specify the full path of the table.

If the keyword **UNIQUE** is specified, the ODBC Paradox driver will create a unique index. The first unique index is created as a primary index. This is a Paradox primary key file named *table-name.PX*. Primary indexes are subject to the following restrictions:

- The primary index must be created before any rows are added to the table.
- A primary index must be defined upon the first "n" columns in a table.
- Only one primary index is allowed per table.
- A table cannot be updated by the Paradox driver if a primary index is not defined on the table.

(Note that this is not true for an empty table, which can be updated even if a unique index is not defined on the table.)

- The *index-name* argument for a primary index must be the same as the base name of the table, as required by Paradox.

If the keyword **UNIQUE** is omitted, the ODBC Paradox driver will create a non-unique index. This consists of two Paradox secondary index files named *table-name.Xnn* and *table-name.Ynn*, where *nn* is the number of the column in the table. Non-unique indexes are subject to the following restrictions:

- Before a non-unique index can be created for a table, a primary index must exist for that table.
- The *index-name* argument for any index other than a primary index (unique or non-unique) must be the same as the column name.
- Only one column can be specified for a non-unique index.

For example, to use the sales order number and line number columns as the unique index on the SO_LINES table, use the statement:

```
CREATE UNIQUE INDEX SO_LINES
  ON SO_LINES (SONum, LineNum)
```

To use the part number column as a non-unique index on the SO_LINES table, use the statement:

```
CREATE INDEX PartNum
  ON SO_LINES (PartNum)
```

Note that when two CREATE INDEX statements are performed, the first statement will always create a primary index with the same name as the table and the second statement will always create a non-unique index with the same name as the column. These indexes will be named this way even if different names are entered in the CREATE INDEX statements and even if the index is labeled UNIQUE in the second CREATE INDEX statement.

Data Types (Advanced)

See Also

Select the driver data types you want to display:

- **Microsoft Access**
- **Btrieve**
- **dBASE**
- **Microsoft Excel**
- **Microsoft FoxPro**
- **Paradox**
- **Text**

Microsoft Access Data Types (Advanced)

[See Also](#)

The following table shows the Microsoft Access data types, data types used to create tables, and ODBC SQL data types.

| Microsoft Access data type | Data type (CREATETABLE) | ODBC SQL data type |
|---------------------------------|-------------------------|--------------------|
| Binary | Binary | SQL_BINARY |
| Bit | Bit | SQL_BIT |
| Char | Char | SQL_VARCHAR |
| Counter | Counter | SQL_INTEGER |
| Currency | Currency | SQL_NUMERIC |
| Date/Time | Date/Time | SQL_TIMESTAMP |
| Long Binary | Long Binary | SQL_LONGVARBINARY |
| Long Text | Long Text | SQL_LONGVARCHAR |
| Memo | Long Text | SQL_LONGVARCHAR |
| Number (FieldSize=Single) | Single | SQL_REAL |
| Number (FieldSize=Double) | Double | SQL_DOUBLE |
| Number (FieldSize=Byte) | Unsigned Byte | SQL_TINYINT |
| Number (FieldSize=Integer) | Short | SQL_SMALLINT |
| Number (FieldSize=Long Integer) | Long | SQL_INTEGER |
| OLE | LongBinary | SQL_LONGVARBINARY |
| Text | VarChar | SQL_VARCHAR |
| VarBinary | VarBinary | SQL_VARBINARY |

Note `SQLGetTypeInfo` returns ODBC SQL data types. All conversions in Appendix D of the *Microsoft ODBC Programmer's Reference* are supported for the SQL data types listed above.

The following table shows limitations on Microsoft Access data types.

| Data type | Description |
|--------------------------------|---|
| BINARY, VARBINARY, and VARCHAR | Creating a BINARY, VARBINARY, or VARCHAR column of zero or unspecified length actually returns a 255-byte column. |
| BIT | BIT columns will display as 0 or 1 when a SELECT is performed, but a query making a direct comparison of a BIT data type to the number "1" will not succeed. If a direct comparison must be made, the comparison should be made to TRUE or <>0. |
| CHAR, LONGVARCHAR, and VARCHAR | A character string literal can be any ANSI character (1-255 decimal). Use two consecutive single quotation marks (") to represent one single quotation mark ('). A zero-length string should not be inserted into a Microsoft Access CHAR, LONGVARCHAR, or VARCHAR column. |

More limitations on data types can be found in [Data Type Limitations](#).

Btrieve Data Types (Advanced)

[See Also](#)

The following table shows how Btrieve data types are mapped to ODBC SQL data types. Note that not all ODBC SQL data types are supported.

| Btrieve data type | ODBC data type |
|--------------------------|-----------------------|
| DATE | SQL_DATE |
| FLOAT4 | SQL_REAL |
| FLOAT8 | SQL_DOUBLE |
| INTEGER1 | SQL_TINYINT |
| INTEGER2 | SQL_SMALLINT |
| INTEGER4 | SQL_INTEGER |
| LOGICAL1 | SQL_BIT |
| MONEY | SQL_NUMERIC |
| NOTE | SQL_LONGVARCHAR |
| STRING | SQL_VARCHAR |

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

The following table shows limitations on Btrieve data types.

| Data type | Description |
|------------------|---|
| INTEGER1 | The driver does not allow indexes on INTEGER1 fields. |
| LOGICAL1 | LOGICAL1 columns will display as 0 or 1 when a SELECT is performed, but a query making a direct comparison of a LOGICAL1 data type to the number "1" will not succeed. If a direct comparison must be made, the comparison should be made to TRUE or <>0. |
| NOTE | The maximum length of a NOTE column is 4000 bytes. The driver does not allow indexes on NOTE fields. |
| STRING | Creating a STRING column of zero or unspecified length actually returns a 255-byte column. |

More limitations on data types can be found in [Data Type Limitations](#).

dBASE Data Types (Advanced)

See Also

The following table shows how dBASE data types are mapped to ODBC SQL data types. Note that not all ODBC SQL data types are supported.

| <u>dBASE data type</u> | <u>ODBC data type</u> |
|------------------------|-----------------------|
| Char | SQL_VARCHAR |
| Date | SQL_DATE |
| Logical | SQL_BIT |
| Memo | SQL_LONGVARCHAR |
| Numeric (BCD) | SQL_DOUBLE |

Precision in dBASE III allows numbers with up to two-digit exponents and in dBASE IV numbers with up to three-digit exponents. Because numbers are stored as text, they are converted to numbers. If the number to convert does not fit in a field, unexplained results may occur.

While dBASE allows a precision and a scale to be specified with a Numeric data type, it is not supported by the ODBC dBASE driver. The ODBC dBASE driver always returns a precision of 15 and a scale of 0 for a Numeric data type.

A column created with the Numeric data type using the ODBC dBASE driver maps to the SQL_DOUBLE ODBC data type. Thus the data in this column is subject to rounding. This behavior is not the same as that of the Numeric data type in dBASE (type N), which is Binary Coded Decimal (BCD).

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

The following table shows limitations on dBASE data types.

| <u>Data type</u> | <u>Description</u> |
|------------------|---|
| CHAR | Creating a CHAR column of zero or unspecified length actually returns a 254-byte column. |
| Encrypted data | The dBASE driver does not support encrypted dBASE tables. |
| LOGICAL | LOGICAL columns will display as 0 or 1 when a SELECT is performed, but a query making a direct comparison of a LOGICAL data type to the number "1" will not succeed. If a direct comparison must be made, the comparison should be made to TRUE or <>0. The dBASE driver cannot create an index on a LOGICAL column. |
| MEMO | The maximum length of a MEMO column is 65,500 bytes. |

More limitations on data types can be found in [Data Type Limitations](#).

Microsoft Excel Data Types (Advanced)

[See Also](#)

The following table shows how Microsoft Excel driver data types are mapped to ODBC SQL data types. The Microsoft Excel driver assigns these data types to columns in Microsoft Excel tables based on the data in the column.

| Microsoft Excel data type | ODBC SQL data type |
|---------------------------|--------------------|
| DateTime | SQL_TIMESTAMP |
| Logical | SQL_BIT |
| Number | SQL_DOUBLE |
| Text | SQL_VARCHAR |

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

The following table shows limitations on Microsoft Excel data types.

| Data type | Description |
|----------------|--|
| Encrypted data | The Microsoft Excel driver cannot read encrypted data. |
| Error Strings | The Microsoft Excel driver cannot return a character string for the Microsoft Excel error values (#N/A!, #VALUE!, #REF!, #DIV/0!, #NUM!, #NAME?, and #NULL!), but returns a NULL instead. |
| LOGICAL | <p>LOGICAL columns will display as 0 or 1 when a SELECT is performed, but a query making a direct comparison of a LOGICAL data type to the number "1" will not succeed. If a direct comparison must be made, the comparison should be made to TRUE or <>0.</p> <p>When a LOGICAL value is returned in a column other than a LOGICAL column, the Microsoft Excel driver cannot convert that LOGICAL value to a TRUE or FALSE character string, but converts the value to NULL instead.</p> |
| TEXT | <p>When the rows of a column contain more than one Microsoft Excel data type, the ODBC Microsoft Excel driver assigns the SQL_VARCHAR data type to the column. There is one exception to this: if the column contains only two or three of the date/time data types (DATE, TIME, and DATETIME), the ODBC Microsoft Excel driver assigns the SQL_TIMESTAMP data type to the column.</p> <p>Creating a TEXT column of zero or unspecified length actually returns a 255-byte column.</p> <p>A character string literal can be any ANSI character (1-255 decimal). Use two consecutive single quotation marks (") to represent one single quotation mark (').</p> |

More limitations on data types can be found in [Data Type Limitations](#).

Microsoft FoxPro Data Types (Advanced)

[See Also](#)

The following table shows how Microsoft FoxPro data types are mapped to ODBC SQL data types. Note that not all ODBC SQL data types are supported.

| Microsoft FoxPro data type | ODBC data type |
|-----------------------------------|-----------------------|
| Char | SQL_VARCHAR |
| Date | SQL_DATE |
| General | SQL_LONGVARBINARY |
| Logical | SQL_BIT |
| Memo | SQL_LONGVARCHAR |
| Numeric (BCD) | SQL_DOUBLE |

Precision in Microsoft FoxPro 2.0 allows numbers with up to two-digit exponents and in Microsoft FoxPro 2.5 numbers with up to three-digit exponents. Because numbers are stored as text, they are converted to numbers. If the number to convert does not fit in a field, unexplained results may occur.

While Microsoft FoxPro allows a precision and a scale to be specified with a Numeric data type, it is not supported by the ODBC Microsoft FoxPro driver. The ODBC Microsoft FoxPro driver always returns a precision of 15 and a scale of 0 for a Numeric data type.

A column created with the Numeric data type using the ODBC Microsoft FoxPro driver maps to the SQL_DOUBLE ODBC data type. Thus the data in this column is subject to rounding. This behavior is not the same as that of the Numeric data type in Microsoft FoxPro (type N), which is Binary Coded Decimal (BCD).

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

The following table shows limitations on Microsoft FoxPro data types.

| Data type | Description |
|------------------|---|
| CHAR | Creating a CHAR column of zero or unspecified length actually returns a 254-byte column. |
| GENERAL | The General data type is supported only for Microsoft FoxPro version 2.5 files. |
| LOGICAL | LOGICAL columns will display as 0 or 1 when a SELECT is performed, but a query making a direct comparison of a LOGICAL data type to the number "1" will not succeed. If a direct comparison must be made, the comparison should be made to TRUE or <>0. |
| MEMO | The maximum length of a MEMO column is 65,500 bytes. |

More limitations on data types can be found in [Data Type Limitations](#).

Paradox Data Types (Advanced)

[See Also](#)

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

| <u>Paradox data type</u> | <u>ODBC SQL data type</u> |
|--------------------------|---------------------------|
| Alphanumeric | SQL_VARCHAR |
| Date | SQL_DATE |
| Image* | SQL_LONGVARBINARY |
| Number | SQL_DOUBLE |
| Short | SQL_SMALLINT |

* Only valid for Paradox versions 4x.

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

The following table shows limitations on Paradox data types.

| <u>Data type</u> | <u>Description</u> |
|------------------|---|
| Alphanumeric | Creating an Alphanumeric column of zero or unspecified length actually returns a 255-byte column. |

More limitations on data types can be found in [Data Type Limitations](#).

Text Data Types (Advanced)

See Also

The following table shows how text data types are mapped to ODBC SQL data types. Note that not all ODBC SQL data types are supported by the ODBC Text driver.

| <u>Text data type</u> | <u>ODBC data type</u> |
|-----------------------|-----------------------|
| CHAR | SQL_VARCHAR |
| DATETIME | SQL_TIMESTAMP |
| FLOAT | SQL_DOUBLE |
| INTEGER | SQL_INTEGER |
| LONGCHAR | SQL_LONGVARCHAR |

Note `SQLGetTypeInfo` returns ODBC data types. All conversions in Appendix D of the *Microsoft ODBC Programmer's Reference* are supported for the SQL data types listed above.

The following table shows limitations on Text data types.

| <u>Data type</u> | <u>Description</u> |
|------------------|--|
| CHAR | Creating a CHAR column of zero or unspecified length actually returns a 255-bit column. |
| DATETIME | MM-DD-YY (for example, 01-17-92) MMM-DD-YY (for example, Jan-17-92) DD-MMM-YY (for example, 17-Jan-92) YYYY-MM-DD (for example, 1992-01-17) YYYY-MMM-DD (for example, 1992-Jan-17) |
| FLOAT | The maximum width includes the sign and decimal point. In SCHEMA.INI, the width is denoted as follows: 14.083 is Float Width 6 -14.083 is Float Width 7 +14.083 is Float Width 7 14083. is Float Width 6 ODBC always returns 8 for float columns. Float columns can also be in scientific notation, for example: -3.04E+2 is Float Width 8 25E4 is Float Width 4 Note Decimal and scientific notation cannot be mixed in a column. |
| INTEGER | Valid values for INTEGER columns are 32767 to -32766. In SCHEMA.INI, the width is denoted as follows: 14083 is Integer Width 5 0 is Integer Width 1 ODBC always returns 4 for integer columns. |

More limitations on data types can be found in [Data Type Limitations](#).

Data Type Limitations (Advanced)

See Also

The Microsoft ODBC drivers impose the following limitations on the data types:

| Data type limitation | Description |
|--|---|
| All data types | Type conversion failures result in the affected column being set to NULL. |
| BIT | TRUE is represented as -1, not 1, so comparisons may not operate properly. A NULL is inserted as a 0. BIT columns will display as 0 or 1 when a SELECT is performed, but a query making a direct comparison of a BIT data type to the number "1" will not succeed. If a direct comparison must be made, the comparison should be made to TRUE or <>0. A value inserted into a column of type BIT (whether character or numeric) that isn't a 0 or a '0' is treated as TRUE. Creating a zero-length BINARY column actually returns a 255-byte BINARY column. |
| DATE | The DATE data type cannot be converted to another data type (or itself) by the CONVERT function. |
| DECIMAL (Exact Numeric) Floating-Point Data Types | Not supported. The number of decimal places in a floating-point number may be limited by the number format set in the International section of the Windows Control Panel. |
| TIMESTAMP | The TIMESTAMP data type cannot be converted to itself by the CONVERT function. |
| TINYINT | Tinyint values are always unsigned. |

See Also

For Advanced Users

[SQL Statements \(Advanced\)](#)

Error Messages (Advanced)

When an error occurs, the Microsoft ODBC Desktop Database drivers return the native error number, the SQLSTATE (an ODBC error code), and an error message.

Native Error

For errors that occur in the driver ISAM layer, the driver returns the native error returned to it by the ODBC File Library (that is, the ODBC driver ISAM).

SQLSTATE

For errors that occur in the data source, the ODBC driver maps the returned native error to the appropriate SQLSTATE. For errors that are detected by the driver or the Driver Manager, the ODBC driver or Driver Manager generates the appropriate SQLSTATE.

Error Message

For errors that occur in the data source, the ODBC driver returns an error message returned to it by the ODBC File Library. For errors that occur in the ODBC driver or the Driver Manager, the driver returns an error message based on the text associated with the SQLSTATE.

Error messages have the following format:

[vendor][ODBC-component][data-source]message-text

where the prefixes in brackets ([]) identify the location of the error. When the error occurs in the Driver Manager, *data-source* is not given. When the error occurs in the data source, the *[vendor]* and *[ODBC-component]* prefixes identify the vendor and name of the ODBC component that received the error from the data source.

The following table shows the error messages returned by the Driver Manager and driver ISAM:

| Error message | Error location |
|---|--|
| <i>[Microsoft][ODBC DLL]message-text</i> | Driver Manager (16-bit ODBC.DLL or 32-bit ODBC32.DLL) |
| <i>[Microsoft][ODBC driver-name][ODBC File Library]message-text</i> | Driver ISAM (see Driver ISAMs Table) |

SQLGetInfo Return Values (Programming)

The following table lists the C-language #defines for the *flInfoType* argument and the corresponding values returned by **SQLGetInfo**. This information can be retrieved by passing the listed C-language #defines to **SQLGetInfo** in the *flInfoType* argument. Where **SQLGetInfo** returns a 32-bit bitmask, a vertical bar (|) represents a bitwise OR. For more information about the values returned by **SQLGetInfo**, see the *Microsoft ODBC SDK Programmer's Reference, Version 2.0*.

| <i>flInfoType</i> value (#define) | Returned value |
|-----------------------------------|---|
| SQL_ACCESSIBLE_PROCEDURES | "Y" (Microsoft Access) "N" (Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, Text) |
| SQL_ACCESSIBLE_TABLES | "Y" |
| SQL_ACTIVE_CONNECTIONS | 64 |
| SQL_ACTIVE_STATEMENTS | 0 |
| SQL_ALTER_TABLE | SQL_AT_ADD_COLUMN SQL_AT_DROP_COLUMN (Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, Paradox) 0 (Microsoft Excel, Text) |
| SQL_BOOKMARK_PERSISTENCE | SQL_BP_SCROLL SQL_BP_UPDATE |
| SQL_COLUMN_ALIAS | "Y" |
| SQL_CONCAT_NULL_BEHAVIOR | SQL_CB_NON_NULL |
| SQL_CONVERT_BIGINT | 0 |
| SQL_CONVERT_BINARY | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_BIT | 0 |
| SQL_CONVERT_CHAR | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_DATE | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_DECIMAL | 0 |
| SQL_CONVERT_DOUBLE | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL |

| | |
|---------------------------|--------------------|
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_FLOAT | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_FUNCTIONS | SQL_FN_CVT_CONVERT |
| SQL_CONVERT_INTEGER | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_LONGVARBINARY | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_LONGVARCHAR | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_NUMERIC | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_REAL | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_SMALLINT | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |
| | SQL_CVT_SMALLINT |
| | SQL_CVT_VARCHAR |
| SQL_CONVERT_TIME | SQL_CVT_DOUBLE |
| | SQL_CVT_FLOAT |
| | SQL_CVT_INTEGER |
| | SQL_CVT_NUMERIC |
| | SQL_CVT_REAL |

| | |
|------------------------------|--|
| | SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_TIMESTAMP | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_TINYINT | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_VARBINARY | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CONVERT_VARCHAR | SQL_CVT_DOUBLE SQL_CVT_FLOAT SQL_CVT_INTEGER SQL_CVT_NUMERIC SQL_CVT_REAL SQL_CVT_SMALLINT SQL_CVT_VARCHAR |
| SQL_CORRELATION_NAME | SQL_CN_DIFFERENT |
| SQL_CURSOR_COMMIT_BEHAVIOR | SQL_CB_CLOSE |
| SQL_CURSOR_ROLLBACK_BEHAVIOR | SQL_CB_CLOSE |
| SQL_DATA_SOURCE_NAME | The DSN from ODBC.INI, or "" if DRIVER keyword is used in ODBC.INI |
| SQL_DATA_SOURCE_READ_ONLY | "N" (Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, Paradox) "Y" (Microsoft Excel, Text) |
| SQL_DATABASE_NAME | Filename |
| SQL_DBMS_NAME | "ACCESS" (Microsoft Access) "BTRIEVE" (Btrieve) "DBASE" (dBASE) "EXCEL" (Microsoft Excel) "FOXPRO" (Microsoft FoxPro) "PARADOX" (Paradox) "TEXT" (Text) |
| SQL_DBMS_VER | "1.0", "1.1", or "2.0" (Microsoft Access) "5.1" or "6.0" (Btrieve) "3.0" or "4.0" (dBASE) "3.0", "4.0", or "5.0" (Microsoft Excel) "2.0", "2.5", or "2.6" (Microsoft |

| | |
|----------------------------|---|
| | FoxPro) |
| | "3.x" or "4.x" (Paradox) |
| | "1.0" (Text) |
| SQL_DEFAULT_TXN_ISOLATION | SQL_TXN_READ_COMMITTED [Microsoft Access] |
| SQL_DRIVER_NAME | "ODBCJT16.DLL" for 16-bit, or "ODBCJT32.DLL" for 32-bit |
| SQL_DRIVER_ODBC_VER | "2.01" |
| SQL_DRIVER_VER | " 2.01.nnnn" (nnnn specifies the build date.) |
| SQL_EXPRESSIONS_IN_ORDERBY | "Y" |
| SQL_FETCH_DIRECTION | SQL_FD_FETCH_NEXT SQL_FD_FETCH_FIRST SQL_FD_FETCH_LAST SQL_FD_FETCH_PRIOR SQL_FD_FETCH_ABSOLUTE SQL_FD_FETCH_RELATIVE SQL_FD_FETCH_BOOKMARK |
| SQL_FILE_USAGE | SQL_FILE_QUALIFIER (Microsoft Access, Btrieve, Microsoft Excel5.0) |
| | SQL_FILE_TABLE (dBASE, Microsoft Excel3.0/4.0, Microsoft FoxPro, Paradox, Text) |
| SQL_GETDATA_EXTENSIONS | SQL_GD_ANY_ORDER SQL_GD_ANY_COLUMN SQL_GD_BLOCK SQL_GD_BOUND |
| SQL_GROUP_BY | SQL_GB_GROUP_BY_ CONTAINS_SELECT |
| SQL_IDENTIFIER_CASE | SQL_IC_MIXED |
| SQL_IDENTIFIER_QUOTE_CHAR | "" (backquote) |
| SQL_KEYWORDS | ALPHANUMERIC, AUTOINCREMENT, BINARY, BOOLEAN, BYTE, CURRENCY, DATABASE, DATABASENAME, DATETIME, DISTINCTROW, DOUBLEFLOAT, FLOAT4, FLOAT8, GENERAL, IEEEDOUBLE, IEEE SINGLE, INT, INTEGER1, INTEGER2, INTEGER4, LOGICAL, LOGICAL1, LONG, LONGBINARY, LONGCHAR, LONGTEXT, MEMO, MONEY, NOTE, NUMBER, OLEOBJECT, OWNERACCESS, PARAMETERS, PERCENT, PIVOT, REAL, SHORT, SINGLE, SINGLEFLOAT, STDEV, STDEVP, STRING, TABLEID, TEXT, TOP, TRANSFORM, UNSIGNEDBYTE, VAR, VARBINARY, VARP, YESNO |
| SQL_LIKE_ESCAPE_CLAUSE | "Y" |
| SQL_LOCK_TYPES | SQL_LOCK_NO_CHANGE |
| SQL_MAX_BINARY_LITERAL_LEN | 255 |

| | |
|-----------------------------|---|
| SQL_MAX_CHAR_LITERAL_LEN | 255 (Microsoft Access) 255 (Btrieve) 254 (dBASE) 255 (Microsoft Excel) 254 (Microsoft FoxPro) 255 (Paradox) 255 (Text) |
| SQL_MAX_COLUMN_NAME_LEN | 64 (Microsoft Access) 20 (Btrieve) 10 (dBASE) 64 (Excel 5.0) / 30 (Excel 3.0/4.0) (Microsoft Excel) 10 (Microsoft FoxPro) 25 (Paradox) 64 (Text) |
| SQL_MAX_COLUMNS_IN_GROUP_BY | 10 |
| SQL_MAX_COLUMNS_IN_INDEX | 32 (Microsoft Access) 0 (Limit Unknown or Not Applicable) (Btrieve) 0 (Limit Unknown or Not Applicable) (dBase) 0 (Microsoft Excel) 0 (Limit Unknown or Not Applicable) (Microsoft FoxPro) 0 (Limit Unknown or Not Applicable) (Paradox) 0 (Text) |
| SQL_MAX_COLUMNS_IN_ORDER_BY | 10 |
| SQL_MAX_COLUMNS_IN_SELECT | 255 |
| SQL_MAX_COLUMNS_IN_TABLE | 255 (Microsoft Access, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, Text) 254 (Btrieve) |
| SQL_MAX_CURSOR_NAME_LEN | 64 |
| SQL_MAX_INDEX_SIZE | 255 (Microsoft Access) 4096 (Btrieve) 220 (dBASE) 0 (Microsoft Excel) 512 (Microsoft FoxPro) 1350 (Paradox) 0 (Text) |
| SQL_MAX_OWNER_NAME_LEN | 0 |
| SQL_MAX_PROCEDURE_NAME_LEN | 64 (Microsoft Access) 0 (Btrieve) 0 (dBASE) 0 (Microsoft Excel) 0 (Microsoft FoxPro) 0 (Paradox) 0 (Text) |
| SQL_MAX_QUALIFIER_NAME_LEN | 66 |
| SQL_MAX_ROW_SIZE | 2096 (Microsoft Access) 4096 (Btrieve) 4000 (dBASE) |

| | |
|--------------------------------|--|
| | 65535 (Microsoft Excel) |
| | 65500 (Microsoft FoxPro 2.5 and above) |
| | 4000 (Microsoft FoxPro 2.0) |
| | 1350 (Paradox) |
| | 65535 (Text) |
| SQL_MAX_ROW_SIZE_INCLUDES_LONG | "N" (Microsoft Access) "Y" (Btrieve) "N" (dBASE) "Y" (Microsoft Excel) "N" (Microsoft FoxPro) "N" (Paradox) "Y" (Text) |
| SQL_MAX_STATEMENT_LEN | 65000 |
| SQL_MAX_TABLE_NAME_LEN | 64 (Microsoft Access) 20 (Btrieve) 12 (dBASE) 31 (Microsoft Excel 5.0) 12 (Microsoft Excel 3.0/4.0) 12 (Microsoft FoxPro) 12 (Paradox) 12 (Text) |
| SQL_MAX_TABLES_IN_SELECT | 16 |
| SQL_MAX_USER_NAME_LEN | 0 |
| SQL_MULT_RESULT_SETS | "N" |
| SQL_MULTIPLE_ACTIVE_TXN | "Y" |
| SQL_NEED_LONG_DATA_LEN | "N" |
| SQL_NON_NULLABLE_COLUMNS | SQL_NNC_NULL |
| SQL_NULL_COLLATION | SQL_NC_LOW |
| SQL_NUMERIC_FUNCTIONS | SQL_FN_NUM_ABS SQL_FN_NUM_ATAN SQL_FN_NUM_CEILING SQL_FN_NUM_COS SQL_FN_NUM_EXP SQL_FN_NUM_FLOOR SQL_FN_NUM_LOG SQL_FN_NUM_MOD SQL_FN_NUM_POWER SQL_FN_NUM_RAND SQL_FN_NUM_SIGN SQL_FN_NUM_SIN SQL_FN_NUM_SQRT SQL_FN_NUM_TAN |
| SQL_ODBC_API_CONFORMANCE | SQL_OAC_LEVEL1 |
| SQL_ODBC_SAG_CLI_CONFORMANCE | SQL_OSCC_NOT_COMPLIANT |
| SQL_ODBC_SQL_CONFORMANCE | SQL_OSC_MINIMUM |
| SQL_ODBC_SQL_OPT_IEF | "N" |
| SQL_ODBC_VER | From Driver Manager |
| SQL_ORDER_BY_COLUMNS_IN_SELECT | "N" |
| SQL_OUTER_JOINS | "Y" |
| SQL_OWNER_TERM | "" |
| SQL_OWNER_USAGE | 0 |
| SQL_POS_OPERATIONS | (SQL_POS_POSITION |

| | |
|------------------------------|---|
| | SQL_POS_REFRESH |
| | SQL_POS_UPDATE |
| | SQL_POS_DELETE |
| | SQL_POS_ADD |
| | (Microsoft Access, Btrieve, dBASE, Microsoft FoxPro, Paradox) |
| | SQL_POS_POSITION |
| | SQL_POS_REFRESH |
| | (Microsoft Excel, Text) |
| SQL_POSITIONED_STATEMENTS | 0 |
| SQL_PROCEDURE_TERM | "QUERY" (Microsoft Access) |
| | "" |
| | (Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, Text) |
| SQL_PROCEDURES | "Y" (Microsoft Access) |
| | "N" (Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, Text) |
| SQL_QUALIFIER_LOCATION | SQL_QL_START |
| SQL_QUALIFIER_NAME_SEPARATOR | ". " (Microsoft Access, Microsoft Excel 5) |
| | "\" (Btrieve, dBASE, Microsoft Excel 4, Microsoft FoxPro, Paradox, Text) |
| SQL_QUALIFIER_TERM | "Database" (Microsoft Access) "Directory" (Btrieve) "Directory" (dBASE) "Workbook" (Microsoft Excel 5)/ "Directory" (Microsoft Excel3/4) "Directory" (Microsoft FoxPro) "Directory" (Paradox) "Directory" (Text) |
| SQL_QUALIFIER_USAGE | SQL_QU_DML_STATEMENTS SQL_QU_TABLE_DEFINITION SQL_QU_INDEX_DEFINITION SQL_QU_PROCEDURE_ INVOCATION (Microsoft Access) |
| | SQL_QU_DML_STATEMENTS SQL_QU_TABLE_DEFINITION SQL_QU_INDEX_DEFINITION (Btrieve, dBASE, Microsoft FoxPro, Paradox) |
| | SQL_QU_DML_STATEMENTS SQL_QU_TABLE_DEFINITION (Microsoft Excel, Text) |
| SQL_QUOTED_IDENTIFIER_CASE | SQL_IC_MIXED |
| SQL_ROW_UPDATES | "N" |
| SQL_SCROLL_CONCURRENCY | SQL_SCCO_READ_ONLY SQL_SCCO_LOCK SQL_SCCO_OPT_VALUES (Microsoft Access, Btrieve, dBASE, |

| | |
|-----------------------------|---------------------------------------|
| | Microsoft FoxPro, Paradox) |
| | SQL_SCCO_READ_ONLY |
| | (Microsoft Excel, Text) |
| SQL_SCROLL_OPTIONS | SQL_SO_FORWARD_ONLY |
| | SQL_SO_STATIC |
| | SQL_SO_KEYSET_DRIVEN |
| SQL_SEARCH_PATTERN_ESCAPE | "\" |
| SQL_SERVER_NAME | "ACCESS" (Microsoft Access) |
| | "BTRIEVE" (Btrieve) |
| | "DBASE" (dBASE) |
| | "EXCEL" (Microsoft Excel) |
| | "FOXPRO" (Microsoft FoxPro) |
| | "PARADOX" (Paradox) |
| | "TEXT" (Text) |
| SQL_SPECIAL_CHARACTERS | "~`@#%\$%^&* _-+=\}\{\"";:;?/><.,![]" |
| SQL_STATIC_SENSITIVITY | SQL_SS_ADDITIONS |
| | SQL_SS_DELETIONS |
| | SQL_SS_UPDATES |
| SQL_STRING_FUNCTIONS | SQL_FN_STR_ASCII |
| | SQL_FN_STR_CHAR |
| | SQL_FN_STR_CONCAT |
| | SQL_FN_STR_LCASE |
| | SQL_FN_STR_LEFT |
| | SQL_FN_STR_LENGTH |
| | SQL_FN_STR_LOCATE |
| | SQL_FN_STR_LTRIM |
| | SQL_FN_STR_RIGHT |
| | SQL_FN_STR_RTRIM |
| | SQL_FN_STR_SPACE |
| | SQL_FN_STR_SUBSTRING |
| | SQL_FN_STR_UCASE |
| SQL_SUBQUERIES | SQL_SQ_COMPARISON |
| | SQL_SQ_EXISTS |
| | SQL_SQ_IN |
| | SQL_SQ_QUANTIFIED |
| | SQL_SQ_CORRELATED_ |
| | SUBQUERIES |
| SQL_SYSTEM_FUNCTIONS | 0 |
| SQL_TABLE_TERM | "TABLE" |
| SQL_TIMEDATE_ADD_INTERVALS | 0 |
| SQL_TIMEDATE_DIFF_INTERVALS | 0 |
| SQL_TIMEDATE_FUNCTIONS | SQL_FN_TD_CURDATE |
| | SQL_FN_TD_CURTIME |
| | SQL_FN_TD_DAYOFMONTH |
| | SQL_FN_TD_DAYOFWEEK |
| | SQL_FN_TD_DAYOFYEAR |
| | SQL_FN_TD_HOUR |
| | SQL_FN_TD_MINUTE |
| | SQL_FN_TD_MONTH |
| | SQL_FN_TD_NOW |
| | SQL_FN_TD_SECOND |
| | SQL_FN_TD_WEEK |
| | SQL_FN_TD_YEAR |
| SQL_TXN_CAPABLE | SQL_TC_ALL |
| | (Microsoft Access) |
| | 0 |

SQL_TXN_ISOLATION_OPTION
SQL_UNION

SQL_USER_NAME

(Btrieve, dBASE, Microsoft Excel,
Microsoft FoxPro, Paradox, Text)
SQL_TXN_READ_COMMITTED
SQL_U_UNION_ALL |
SQL_U_UNION
Not supported

See Also

For Advanced Users

[SQL Statements \(Advanced\)](#)

For Programmers

[Scalar Functions \(Programming\)](#)

Scalar Functions (Programming)

The Microsoft ODBC Desktop Database drivers support the following scalar functions:

| | | |
|----------------|--------|-----------|
| CONCAT | LCASE | RIGHT |
| CONVERT | LEFT | RTRIM |
| CURDATE | LENGTH | SUBSTRING |
| CURTIME | LOCATE | UCASE |
| DATABASE | LTRIM | USER |
| DAYOFMON TH | MOD | YEAR |
| DAYOFWEE K | MONTH | |

For information about the arguments and return values of scalar functions, see Appendix G of the *Microsoft ODBC SDK Programmer's Reference*.

ODBC API Functions (Programming)

See Also

The Microsoft ODBC drivers support all Core and Level 1 functions and the following Level 2 functions:

- **SQLDataSources**
- **SQLDrivers**
- **SQLExtendedFetch**
- **SQLMoreResults**
- **SQLNativeSQL**
- **SQLNumParams**
- **SQLProcedures**
- **SQLProcedureColumns**
- **SQLSetPos**
- **SQLSetScrollOptions**

The following table describes how the ODBC Desktop Database Drivers implement specific functions.

| Function | Description |
|-------------------------|--|
| SQLColAttributes | <p>LONGVARBINARY and LONGVARCHAR columns are reported as SQL_UNSEARCHABLE.</p> <p>For LONGVARBINARY data, SQL_COLUMN_DISPLAY_SIZE is the maximum length of the column, not the maximum length of the column times 2.</p> <p>The SQL_COLUMN_MONEY statement option of SQLColAttributes returns FALSE for a column of Paradox data type CURRENCY, which maps to the ODBC SQL data type SQL_DOUBLE.</p> |
| SQLColumns | <p>A column may be shown as nullable by SQLColumns even if the column participates in a primary key or unique index, so cannot accept a NULL value.</p> |
| SQLConnect | <p>When the Microsoft Access or Paradox driver is used, the PWD keyword should not include any of the special characters (see SQL_SPECIAL_CHARACTERS in <u>SQLGetInfo Return Values</u>).</p> |
| SQLDriverConnect | <p>SQLDriverConnect enables you to <u>connect to a driver</u> without creating a data source (DSN).</p> <p>The following keywords are supported in the <u>connection string</u> for all drivers: DSN, DBQ, and FIL.</p> <p>For the Microsoft Access driver, the UID and PWD keywords are also supported. For the Paradox driver, the PWD keyword is also supported.</p> <p>When the Microsoft Access or Paradox driver is used, the PWD keyword should not include any of the special characters (see SQL_SPECIAL_CHARACTERS in <u>SQLGetInfo Return Values</u>).</p> |

| | |
|----------------------------|---|
| SQLGetData | <p>When the Paradox driver is used, after a password-protected file has been opened by a user, other users are not allowed to open the same file.</p> <p>This function can retrieve data from any column, whether or not there are bound columns after it and regardless of the order in which the columns are retrieved.</p> |
| SQLGetInfo | <p>SQLGetInfo supports a driver-specific information type, <code>SQL_FILE_USAGE</code>. The returned value is a 16-bit integer that indicates how the driver directly treats files in a data source:</p> <p>0 (<code>SQL_FILE_NOT_SUPPORTED</code>) = The driver is not a single-tier driver.</p> <p>1 (<code>SQL_FILE_TABLE</code>) = A single-tier driver treats files in a data source as tables.</p> <p>3 (<code>SQL_FILE_QUALIFIER</code>) = A single-tier driver treats files in a data source as a qualifier.</p> |
| SQLGetTypeInfo | <p>The ODBC driver returns 3 for the Microsoft Access or Btrieve driver, because each file is a complete database. The ODBC driver returns 1 for the dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver, because each file is a table.</p> <p>The name of the type (<code>TYPE_NAME</code>) returned in the table produced by SQLGetTypeInfo will be the name most commonly used by the data source.</p> |
| SQLMoreResults | <p>When the Text driver is used, SQLGetTypeInfo returns a <code>CASE_SENSITIVE</code> value of <code>FALSE</code> for the text data types (<code>CHAR</code> and <code>LONGCHAR</code>), when the data types actually are case-sensitive.</p> |
| SQLPrepare | <p>This function always returns <code>SQL_NO_DATA_FOUND</code>.</p> <p>A statement prepared with an incorrect number of columns returns an error at execution time, not upon statement preparation.</p> |
| SQLProcedureColumns | <p>Supported for Microsoft Access DSNs only.</p> <p>Pattern matching is not supported for <code>szProcName</code>.</p> <p><code>SQL_PARAM_INPUT</code> or <code>SQL_RESULT_COL</code> is returned for <code>COLUMN_TYPE</code>.</p> |
| SQLProcedures | <p><code>SQL_PT_PROCEDURE</code> is returned in <code>PROCEDURE_TYPE</code> for all procedures.</p> |

| | |
|----------------------------|---|
| SQLSetConnectOption | <p>This function supports the SQL_ACCESS_MODE, SQL_CURRENT_QUALIFIER, SQL_OPT_TRACE, and SQL_OPT_TRACEFILE connection options.</p> <p>When the Microsoft Access driver is used, SQL_TXN_ISOLATION is always SQL_TXN_READ_COMMITTED. SQL_TRANSLATE_DLL and SQL_TRANSLATION_OPTION are not supported.</p> <p>SQL_LOGIN_TIMEOUT is not supported.</p> <p>When the Microsoft Access driver is used, the SQL_AUTOCOMMIT option may be set to either ON or OFF, because the Microsoft Access driver supports transactions.</p> <p>The Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, and Text drivers only support SQL_AUTOCOMMIT being set to ON (the default state), because they do not support transactions.</p> |
| SQLSetCursorName | <p>Because the driver does not support a positioned update or delete by the WHERE CURRENT OF <i>cursorname</i> syntax, SQLSetCursorName is supported, but cannot be used for positioned updates.</p> |
| SQLSetPos | <p>SQL_LOCK_NO_CHANGE is supported for <i>fLock</i>.</p> <p>SQL_LOCK_EXCLUSIVE and SQL_LOCK_UNLOCK are not supported.</p> <p>SQLSetPos supports updatable joins. (For more information on updatable joins, see the documentation for your database management system.)</p> |
| SQLSetScrollOptions | <p>Forward and static cursors are supported for SQL_CONCUR_READ_ONLY.</p> <p>Only keyset-driven cursors are supported for SQL_CONCUR_LOCK.</p> <p>Dynamic cursors and mixed cursors are not supported.</p> |
| SQLSetStmtOption | <p>The only valid keyset size (SQL_KEYSET_SIZE) is 0, because mixed and dynamic cursors are not supported.</p> <p>SQL_ROW_NUMBER returns 0 to indicate that a row number cannot be determined.</p> <p>SQL_SIMULATE_CURSOR is not</p> |

| | |
|--------------------------|--|
| SQLSpecialColumns | <p>supported.</p> <p>A unique index will be returned (if one exists) for the SQL_BEST_ROWID flag in <i>fColType</i>. No result set will be returned for the SQL_ROWVER flag. All row IDs have a scope of SQL_SCOPE_SESSION.</p> |
| SQLStatistics | <p>Filtering is based on uniqueness (<i>fUnique</i>). The <i>fAccuracy</i> parameter is ignored.</p> <p>NON_UNIQUE, SEQ_IN_INDEX, COLUMN_NAME, and COLLATION are index-dependent.</p> <p>NULL is always returned in the INDEX_QUALIFIER and PAGES columns.</p> <p>Only SQL_TABLE_STAT or SQL_INDEX_OTHER will be returned for TYPE.</p> <p>CARDINALITY will be returned for the Microsoft Access driver. CARDINALITY will not be returned for the Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver.</p> |
| SQLTables | <p>When the Microsoft Access driver is used, "SYSTEM TABLE" is supported for <i>szTableType</i> for system tables, "SYNONYM" is supported for attached tables, and "VIEW" is supported for row returning queries.</p> <p>When the Btrieve, dBASE, Microsoft Excel 3.0/4.0, Microsoft FoxPro, Paradox, or Text driver is used, "TABLE" is the only table type supported.</p> <p>When the Microsoft Excel 5.0 driver is used, sheet names (tables with a "\$" on the end) are returned as "SYSTEM TABLE", and "TABLE" is returned for all other sheets.</p> <p>When the Text driver is used, the list of files returned by SQLTables is determined by the file extensions in the Extensions List box in the ODBC Text Setup dialog box.</p> |
| SQLTransact | <p>When the Microsoft Access driver is used, COMMIT and ROLLBACK via the SQLTransact function are supported.</p> <p>When the Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver is used, this function supports COMMIT, but not ROLLBACK.</p> <p>When the Microsoft Access driver is</p> |

used, if a failure occurs during the commit process, the affected database can be repaired using the Repair Database option in the Microsoft Access driver setup, or through the use of the REPAIR_DB keyword in the SQLConfigDataSource function.

See Also

For Advanced Users

[Error Messages \(Advanced\)](#)

For Programmers

[Implementation Issues \(Programming\)](#)

SQLConfigDataSource Implementation (Programming)

The **SQLConfigDataSource** function that is used to add, modify, or delete a data source dynamically uses the following arguments.

| Keyword | Description |
|----------------------|---|
| CHARACTERSET | For the Text driver, OEM or ANSI. |
| COLNAMEHEADER | For the Text driver, indicates whether the first record of data will specify the column names. Either TRUE or FALSE. |
| COMPACT_DB | For the Microsoft Access driver, performs <u>data compaction</u> on a database file. |
| CREATE_DB | For the Microsoft Access driver, creates a database file. |
| DBQ | For the Microsoft Access, Btrieve, or Paradox driver, the name of the database file. For the Microsoft Excel 5 driver, the name of the workbook file. |
| DEFAULTDIR | The path specification to the database file (for the Microsoft Access driver) or directory (for the Btrieve, dBASE, Microsoft Excel, Microsoft FoxPro, Paradox, or Text driver). |
| DESCRIPTION | A description of the data in the data source. |
| DRIVER | The path specification to the driver DLL. |
| DRIVERID | An integer ID for the driver. 25 (Microsoft Access) 20 (Btrieve) 21 (dBASEIII) 277 (dBASEIV) 534 (Microsoft Excel3) 278 (Microsoft Excel4) 22 (Microsoft Excel5) 24 (Microsoft FoxPro2.0) 280 (Microsoft FoxPro2.5) 536 (Microsoft FoxPro2.6) 26 (Paradox3) 282 (Paradox4) 27 (Text) |
| EXCLUSIVE | Determines whether the database will be opened in exclusive mode (accessed by only one user at a time) or shared mode (accessed by more than one user at a time). Can be true (exclusive mode) or false (shared mode). (This does not apply to the Microsoft Excel or Text driver.) |
| EXTENSIONS | Lists the filename extensions of the Text files on the data source. |
| FIL | File type |

| | |
|--------------------|--|
| | (MS Access for Microsoft Access; BTRIEVE; DBASE3 or DBASE4; EXCEL 3.0, 4.0, or 5.0 for Microsoft Excel; FOXPRO 2.0, 2.5, or 2.6 for Microsoft FoxPro; PARADOX 3.x or 4.x). (This does not apply to the Text driver.) |
| FILETYPE | File type for the Text driver (Text). |
| FORMAT | For the Text driver, can be FIXEDLENGTH, TABDELIMITED, CSVDELIMITED (by a comma), or DELIMITED() (by the special character specified in the parentheses). The special character is one character in length and can be in character, decimal, or hexadecimal format. |
| JETINIPATH | Path to the initialization file. |
| MAXSCANROWS | For the Microsoft Excel or Text driver, the number of rows to be scanned when setting a column's data type based upon existing data. For the Microsoft Excel driver, you may enter a number from 1 to 16 for the rows to scan; however, the value will always default to 1. (A number outside the limit will return an error.) For the Text driver, you may enter a number from 1 to 32767 for the number of rows to scan; however, the value will always default to 1. (A number outside the limit will return an error.) |
| PWD | For the Microsoft Access or Paradox driver, the password. |
| READONLY | TRUE to make file read-only; FALSE to make file not read-only. |
| REPAIR_DB | For the Microsoft Access driver, <u>repairs a database</u> damaged by a failure that occurs during the commit process. |
| SYSTEMDB | For the Microsoft Access driver, the path specification to system database file. |
| UID | For the Microsoft Access driver, the user ID name used for login. |

SQLDriverConnect Implementation (Programming)

The following table shows the minimum keywords required to connect to each driver, and provides an example of keyword/value pairs used with **SQLDriverConnect**. For a full list of DRIVERID values, see [SQLConfigDataSource Implementation](#).

Note If DBQ or DefaultDir is not specified for the **Btrieve**, **dBASE**, **Excel 3/4**, **FoxPro**, **Paradox**, or **Text** driver, the driver will connect to the current directory.

| Driver | Keywords Required | Examples |
|--------------------|--------------------------|--|
| Microsoft Access | Driver, DBQ | Driver={Microsoft Access Driver (*.mdb)}; DBQ=c:\temp\sample.mdb |
| Btrieve | Driver | Driver={Microsoft Btrieve Driver (file.dd)}; DBQ=c:\temp |
| dBASE | Driver, DriverID | Driver={Microsoft dBASE Driver (*.dbf)}; DBQ=c:\temp; DriverID=277 |
| Microsoft Excel3/4 | Driver, DriverID | Driver={Microsoft Excel Driver (*.xls)}; DBQ=c:\temp; DriverID=278 |
| Microsoft Excel5 | Driver, DriverID, DBQ | Driver={Microsoft Excel Driver (*.xls)}; DBQ=c:\temp\sample.xls;DriverID=22 |
| Microsoft FoxPro | Driver, DriverID | Driver={Microsoft FoxPro Driver (*.dbf)}; DBQ=c:\temp;DriverID=536 |
| Paradox | Driver, DriverID | Driver={Microsoft Paradox Driver (*.db)}; DBQ=c:\temp;DriverID=26 |
| Text | Driver | Driver={Microsoft Text Driver (*.txt;*.csv)}; DefaultDir=c:\temp |

Implementation Issues (Programming)

The following information might affect the use of the Microsoft ODBC drivers.

| Issue | Implementation |
|---|---|
| Arithmetic Errors | The ODBC driver evaluates the WHERE clause in a SELECT statement as it fetches each row. If a row contains a value that causes an arithmetic error, such as divide-by-zero or numeric overflow, the driver returns all rows, but returns errors for columns with arithmetic errors. When inserting or updating, however, the ODBC driver stops inserting or updating data when the first arithmetic error is encountered. |
| Creating and Opening Tables | When the Text driver is used, a new table is created using the format specified in ODBCINST.INI. If not specified, tables are created in CSVDELIMITED format. By default, INTEGER columns default to 11 characters and FLOAT columns default to 22 characters. DATE columns use the YYYY-MM-DD format. CHAR and LONGCHAR columns are the width specified in the CREATE statement. |
| Multiple <i>hstmt</i> s | When the ODBC Paradox driver is used, if you want to execute more than one <i>hstmt</i> on a table, the table must have a unique index (Paradox primary key). |
| Read-Only Status | When the Microsoft Excel or Text driver is used, data source tables are read-only. Even though tables have read-only status, however, applications can perform insertions and updates for Microsoft Excel tables, or insertions for Text tables. |
| Setup DLL | The ODBC Administrator calls the function ConfigDSN when users configure data sources. |
| Single-User Mode versus Multi-User Mode | The ODBC drivers can use a data source in <u>single- or multi-user</u> mode. The user specifies which mode to use in the <u>ODBC Setup dialog box</u> . |
| Translation DLLs | Not supported |

Single-User Mode versus Multi-User Mode

The Microsoft ODBC drivers allow only one user at a time to be connected to a single-user data source. When it uses a file in the data source, such as when it processes a SELECT or UPDATE statement, it exclusively locks the file.

The ODBC driver allows any number of users to be connected to a multi-user data source at the same time. It exclusively locks records before updating or deleting them. Other users cannot update or delete a locked record. Whether they can read the record depends on the version of the file.

For the **dBASE** driver, other users cannot read locked dBASE III records; they can read locked dBASE

IV records ("dirty reads" are permitted).

For the **Microsoft FoxPro** driver, other users cannot read locked Microsoft FoxPro 2.0 records; they can read locked Microsoft FoxPro 2.5 or 2.6 records ("dirty reads" are permitted).

Note Although the ODBC driver locks records in multi-user mode, this does not imply that serializable transactions are available. Only the **Microsoft Access** driver supports transactions.

API

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

character set

A character set is a set of 256 letters, numbers, and symbols specific to a country or language. Each character set is defined by a table called a code page. An OEM (Original Equipment Manufacturer) character set is any character set except the ANSI character set. The ANSI character set (code page 1007) is the character set used by Microsoft Windows.

conformance level

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

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

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

data source

A data source includes the data a user wants to access and the information needed to get to that data.

Examples of data sources are:

- A SQL Server database, the server on which it resides, and the network used to access that server.
- A directory containing a set of dBASE files you want to access.

DBMS

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

DDL

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

DLL

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

DML

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

ODBC

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

ODBC Driver Manager

A dynamic-link library (DLL) that provides access to ODBC drivers.

ODBC driver

A dynamic-link library (DLL) that an ODBC-enabled application, such as Microsoft Excel, can use to gain access to a particular data source. Each database management system (DBMS), such as Microsoft SQL Server, requires a different driver.

SQL

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

SQL statement

A command written in Structured Query Language (SQL); also known as a query. An SQL statement specifies an operation to perform, such as SELECT, DELETE, or CREATE TABLE; the tables and columns on which to perform that operation; and any constraints to that operation.

translation option

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

translator

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

