INTERSOLV

DataDirect Connect ODBC for Text

INTERSOLV

About Connect ODBC for Text

Connect ODBC for Text (the "Text driver") supports ASCII text files. These files can be printed directly or edited with text editors or word processors because none of the data is stored in a binary format.

The driver file name is IVTXT*nn*.DLL, where *nn* is the revision level.

The Text driver executes SQL statements directly on the text files. The driver supports Insert statements and inserts the record at the end of the file. You can execute Update and Delete statements conditionally.

Common Text File Formats

Some common formats for text files are listed in the following table:

Format	Description
Comma-separated values	Commas separate column values, and each line is a separate record. Column values can vary in length. These files often have the .CSV extension.
Tab-separated values	Tabs separate column values, and each line is a separate record. Column values can vary in length.
Character-separated values	Any printable character except single or double quotation marks can separate column values, and each line is a separate record. Column values can vary in length.
Fixed	No character separates column values. Instead, values start at the same position and have the same length in each line. The values appear in fixed columns if you display the file. Each line is a separate record.
Stream	No character separates column values nor records. The table is one long stream of bytes.

Comma-, tab-, and character-separated files are called character-delimited files, since values are separated by a special character.

Configuring Data Sources

To configure a Text data source, do the following:

1   Start the ODBC Administrator to display a list of data sources.

2   If you are configuring an existing data source, select the data source name and click **Configure** to display the ODBC Text Driver Setup dialog box.

If you are configuring a new data source, click **Add** to display a list of installed drivers. Select TextFile and click **Finish** to display the ODBC Text Driver Setup dialog box.

3   Specify a data source name, a database directory, and optionally, a description. a default table type, a delimiter character, and whether to use column names in the first line. Click **Apply**.

4   Click the **Advanced** tab to configure optional data source settings, such as the action to be taken if an undefined table is used. Click **Apply**.

5   Click **Translate** to display the Select Translator dialog box, where you can select a translator to translate your data from one character set to another. INTERSOLV provides a translator named INTERSOLV OEM ANSI that translates your data from the IBM PC character set to the ANSI character set. Click **OK** to close this dialog box and perform the translation.

The translators that are listed in this dialog box are determined by the values listed in the ODBC Translators section of the system information.

6   Click **Define** to define the structure of your text files. Defining Table Structure discusses how to define the column names and data types in a table.

7   Click **OK** or **Cancel**. If you click **OK**, the you have specified become the defaults when you connect to the data source. You can change these defaults by using this procedure to reconfigure your data source. You can override these defaults by connecting to the data source using a connection string with alternate values.

Defining Table Structure

Since text files do not all have the same structure, the driver provides the option of defining the structure of an existing file. Although defining the structure is not mandatory (the driver can attempt to guess the names and types of the columns), this feature is extremely useful.

Define the structure of a file as follows:

1 Display the ODBC Text Driver Setup dialog box through the ODBC Administrator. Click the **Advanced tab**, then click **Define** to display the Define File dialog box.

2 Select the correct file and click **Open** to display the Define Table dialog box.

The Database Name and File fields display the name of the database directory that contains the file and the name of the file you selected in the Define File dialog box, respectively.

3 Use the Table Information pane to specify information about the overall structure of the file:

   a In the Table box, type a table name. The name may be up to 32 characters long and must be unique. This name is returned by SQLTables. By default, it is the file name without its extension.

   b Select the Column Names in First Line check box if the first line of the file contains column names.

   c Open the Table Type box and select a table type: comma, tab, fixed, character, or stream.

   d If the table type is character-separated, type the character that separates the values in the Delimiter Character box.

   e In the Decimal Symbol box, type a comma if you want to store the data using a comma as the separator for decimal numbers.

4 Use the Column Information pane to define the names and types of the columns in the table.

If you specified a comma-separated, tab-separated, or character-separated table type, you can click **Guess** to have the driver display in the Columns list box what it thinks the columns are. You can then modify the column definitions by selecting a column name, specifying values in the Name, Type, Mask, Precision, and Scale boxes (as appropriate), and clicking **Modify**.

or

If you do not want the driver to guess the columns, take the following steps to define each column and add its name to the Columns listbox:

   a In the Name box, type the name of the column.

   b Open the Type drop-down list and select the data type of the column. If the type is Date, you must select a date mask for the column or type one in. See Date Masks for more information.

   c In the Precision box, type the precision of the column.

   d In the Scale box, type the scale of the column.

    The precision and scale values determine how numeric data is to be returned.

   e If you specified a fixed-length table type, you may specify the length and offset in the Length and Offset boxes. The length is the number of bytes the data takes up in storage; the offset is the number of bytes from the start of the table to the start of the field.

    or

    Click **Parse** to display the Parse Table dialog box and define the table columns.

    This dialog box displays the first line of the file. You must mark where *each* column begins and ends by enclosing the name in brackets. These brackets indicate the position and length of each field value in the record. Click **OK** to close the Parse Table dialog box.

f Click **Add.**

Note: At any time, you can select a column in the Columns list box and click **Remove** to delete its definition.

5 Click **OK** to define the table.

Date Masks

Date masks tell the driver how a date is stored in a text file. When a value is inserted into a text file, the date is formatted so that it matches the mask. When reading a text file, the driver converts the formatted date into a date data type.

The following table lists the symbols to use when specifying the date mask:

Symbol	Description
m	Output the month's number (1-12).
mm	Output a leading zero if the month number is less than 10.
mmm, Mmm, MMM	Output the three-letter abbreviation for the month depending on the case of the Ms (that is, jan, Jan, JAN).
mmmm, Mmmm, MMMM	Output the full month name depending on the case of the Ms (that is, january, January, JANUARY).
d	Output the day number (1-31).
dd	Output a leading zero if the day number is less than 10.
ddd, Ddd, DDD	Output the three-letter day abbreviation depending on the case of the D's (that is, mon, Mon, MON).
dddd, Dddd, DDDD	Output the day depending on the case of the D's (that is, monday, Monday, MONDAY).
yy	Output the last two digits of the year.
yyyy	Output the full four digits of the year.
J	Output the Julian value for the date. The Julian value is the number of days since 4712 BC.
\ - . : , (space)	Special characters used to separate the parts of a date.
\	Output the next character. For example, if the mask is mm/dd/yyyy \A\D, the value appears as 10/01/1993 AD in the text file.
"string", 'string'	Output the string in the text file.

The following table shows some example date values, masks, and how the date appears in the text file.

Date	Mask	Value
1993-10-01	yyyy-mm-dd	1993-10-01
	m/d/yy	10/1/93
	Ddd, Mmm dd, yyyy	Fri, Oct 01, 1993

Connecting to a Data Source Using a Connection String

If your application requires a connection string to connect to a data source, you must specify the data source name that tells the driver which section of the system information to use for the default connection information. Optionally, you may specify *attribute=value* pairs in the connection string to override the default values stored in the system information. These values are not written to the system information.

You can specify either long or short names in the connection string. The connection string has the form:

```
DSN=data_source_name[;attribute=value[;attribute=value]...]
```

An example of a connection string for text files is:

```
DSN=TEXT FILES;TT=CHARACTER;DC=&
```

The paragraphs that follow give the long and short names for each attribute, as well as a description. The defaults listed are initial defaults that apply when no value is specified in either the connection string or in the data source definition in the system information. If you specified a value for the attribute when configuring the data source, that value is your default.

AllowUpdateAndDelete (AUD): AllowUpdateAndDelete={0 | 1}. Specifies whether a data source allows Update and Delete statements. The default is 0. Because Update and Delete statements cause immediate changes to a table, only one connection at a time can operate on a table. When this option is set, tables are opened exclusively by the current connection. Each update and delete on a text file can cause significant changes to the file, and performance may be poor. Consider a more appropriate database form if performance is a significant factor.

ApplicationUsingThreads (AUT): ApplicationUsingThreads-{0 | 1}. Ensures that the driver works with multi-threaded applications. The default is 1, which makes the driver thread-safe. When using the driver with single-threaded applications, you may set this option to 0 to avoid additional processing required for ODBC thread safety standards.

CacheSize (CSZ): The number of 64 KB blocks the driver uses to cache database records. The higher the number, the better the performance. The maximum number of blocks you can set depends on the system memory available. If the cache size is greater than 0, when browsing backwards, you will not be able to see updates made by other users until you run the Select statement again. The default is 4 (4x64KB=256KB).

Database (DB): The directory in which the text files are stored.

DataFileExtension (DFE): String of three or fewer characters that specifies the file extension to use for data files. The default DataFileExtension value is TXT. The DataFileExtension value is used for all Create Table statements. Sending a Create Table using an extension other than the DataFileExtension value causes an error.

In other SQL statements such as Select or Insert, users can specify an extension other than the DataFileExtension value. The DataFileExtension value is used when no extension is specified.

DataSourceName (DSN): A string that identifies a Text data source configuration in the system information. Examples include "Accounting" or "Text Files."

DecimalSymbol (DS): DecimalSymbol={. | ,}. Specifies the decimal separator used when data is stored. The international decimal symbol (.) must be used in DML statements and parameter buffers.

Delimiter (DC): The character used as a delimiter for character-separated files. It can be any printable character except a single or double quote. The initial default is a comma (,).

ExtraExtensions (EE): A list of additional filename extensions to be recognized as text tables. When an application requests a list of tables, only files that have been defined are returned. To have the driver also return names of undefined files, specify a comma-separated list of file extensions. To specify files with no extension, use the keyword None.

FileOpenCache (FOC): The maximum number of unused file opens to cache. For example, when FileOpenCache=4, and a user opens and closes four files, the files are not actually closed. The driver keeps them open so that if another query uses one of these files, the driver does not have to perform another open, which is expensive. The advantage of using file open caching is increased performance. The disadvantage is that a user who tries to open the file exclusively may get a locking conflict even

though no one appears to have the file open. The initial default is 0.

FirstLineNames (FLN): FirstLineNames={0 | 1}. A value that determines whether the driver looks for column names in the first line of the file. If FirstLineNames=1, the driver looks for column names in the first line of the file. If FirstLineNames=0 (the initial default), the first line is interpreted as the first record in the file.

IntlSort (IS): IntlSort={0 | 1}. This attribute determines the order that records are retrieved when you issue a Select statement with an Order By clause. If IntlSort=0 (the initial default), the driver uses the ASCII sort order. This order sorts items alphabetically with uppercase letters preceding lowercase letters. For example, "A, b, C" would be sorted as "A, C, b."

If IntlSort=1, the driver uses the international sort order as defined by your operating system. This order is always alphabetic, regardless of case; the letters from the previous example would be sorted as "A, b, C." See your operating system documentation concerning the sorting of accented characters.

ScanRows (SR): The number of rows in a text file that the driver scans to determine the column types in the file. If the value is 0, all rows in the file are scanned. The initial default is 25.

TableType (TT): TableType={Comma | Tab | Character | Fixed | Stream} The Text driver supports five types: comma-separated, tab-separated, character-separated, fixed-length, and stream. Setting this value tells the driver the default type, which is used when creating a new table and opening an undefined table.

UndefinedTable (UT): The Text driver can perform two operations when it encounters a file that has not been defined. UndefinedTable=Prompt tells the driver to display a dialog box that allows the user to describe the file's format. UndefinedTable=Guess tells the driver to guess the file's format. This is the initial default.

UseLongQualifiers (ULQ): UseLongQualifiers={0 | 1}. Specifies whether the driver uses long path names as table qualifiers. The default is 0, do not use long path names (the default length of path names is 128 characters). If UseLongQualifiers=1, the driver uses long path names (up to 255 characters).

Note: The ScanRows, TableType, Delimiter, FirstLineNames, DecimalSymbol, and Charset attributes apply to tables that have *not* been defined. These attributes also determine the characteristics of new tables created with the Create Table statement.

Data Types

The text file data types are mapped to the standard ODBC data types as follows:

Text	ODBC
Numeric	SQL_NUMERIC
Date	SQL_TYPE_DATE
Varchar	SQL_VARCHAR

SQL Statements for the Text Driver

Select Statements

You use the SQL Select statement to specify the columns and records to be read. The driver supports all Select statement clauses as described in SQL for Flat-File Drivers in the DataDirect ODBC Drivers General Help.

Alter Table Statement

The Text driver supports the Alter Table statement to add one or more columns to the table, or to delete (drop) a single column.

The Alter Table statement has the form:

```
ALTER TABLE table_name {ADD column_name data_type | ADD (column_name data_type [, column_name
data_type] . . . ) |
DROP [COLUMN] column_name}
```

table_name is the name of the table for which you are adding or dropping columns.

column_name assigns a name to the column you are adding or specifies the column you are dropping.

data_type specifies the native data type of each column you add. See <ins>Text Data Types</ins> for more information.

For example, to add two columns to the emp table:

```
ALTER TABLE emp (ADD startdate date, dept varchar (10))
```

You cannot add and drop columns in a single statement, and you can drop only one column at a time. For example, to drop a column:

```
ALTER TABLE emp DROP startdate
```

ODBC Conformance Level

The API functions supported are listed in *Supported ODBC Functions*, found in the General Help on DataDirect ODBC Drivers. The driver also supports backward and random fetching in SQLExtendedFetch and SQLFetchScroll.

The driver supports the minimum SQL grammar.

Number of Connections and Statements Supported

Text files support multiple connections and multiple statements per connection.

ODBC Text Driver Setup Dialog Box

Use the ODBC Text Driver Setup dialog box to <u>create</u> new data sources or <u>configure</u> existing data sources.

Data Source Name: A string that identifies this Text data source configuration in the system information. Examples include "Accounting" or "Text Files."

Description: An optional long description of a data source name. For example, "My Accounting Files" or "My Text Files in the Accounting Directory."

Database Directory: The directory in which the text files are stored. If none is specified, the current working directory is used.

Default Table Type: The type of text file: comma-separated, tab-separated, character-separated, fixed-length, or <u>stream</u>. This value tells the driver the default type, which is used when creating a new table and opening an undefined table.

Delimiter Character: The character used as a delimiter for character-separated files. It can be any printable character. The default is a comma (,).

Column Names in First Line: Select this check box to tell the driver to look for column names in the first line of the file.

Note: The Default Table Type, Delimiter Character, and Column Names in First Line settings apply only to tables *not* previously defined. These fields also determine the attributes of new tables created with the Create Table statement.

Advanced Tab

Displays the <u>Advanced tab</u>, where you can configure optional data source settings, such as the action to be taken if an undefined table is used.

OK

Cancel

Apply

Advanced Tab, ODBC Text Driver Setup Dialog Box

Use the Advanced tab on the ODBC Text Driver Setup dialog box to specify optional settings when you create new Text data sources or configure existing data sources.

Rows to Scan: The number of rows in a text file that the driver scans to determine the data types in the file. If the value is 0, all rows in the file are scanned. The default is 25.

Note: The Rows to Scan setting applies only to tables *not* previously defined. This field also determines the attributes of new tables created with the Create Table statement.

Action for Undefined Tables: Two radio buttons that indicate what action the driver should take when it encounters a file that has not been defined. Set the Prompt for Definition radio button, if you want the driver to prompt the user when it encounters a file whose format is not defined. Otherwise, set the Guess Definition radio button to have the driver guess the file's format.

Return Additional Tables: Select this check box to tell the driver to return files with a given extension in addition to the files specified in the Data File Extension field. In Extension List, specify a comma-separated list of these extensions. To have files with no extensions returned, specify None. For example, if some of your files have the extensions .TXT and .CSV and others have no extension, specify TXT,CSV,NONE.

By default, when an application requests a list of tables, only files that have been defined are returned.

File Open Cache: A numeric value to specify the maximum number of unused file opens to cache. For example, the value 4 specifies that when a user opens and closes four tables, the tables are not actually closed. The driver keeps them open so that if another query uses one of these tables, the driver does not have to perform another open, which is expensive. The advantage of file open caching is increased performance. The disadvantage is that a user who specifies file locking on open may get a locking conflict even though no one appears to have the file open. The default is 0, which means no file open caching.

Cache Size: The amount of memory, in 64 KB blocks, that the driver uses to cache database records. The higher the number, the better the performance. The maximum number you can set depends on the system memory available. The value (in kilobytes) must be a multiple of 64. The default is 4 (4x64KB=256KB). If the cache size is greater than 0, when browsing backwards, you will not be able to see updates made by other users until you run the Select statement again.

Data File Extension: Specifies the file extension to use for data files. The default Data File Extension setting is .TXT. The Data File Extension setting cannot be greater than three characters. The Data File Extension setting is used for all Create Table statements. Sending a Create Table using an extension other than the Data File Extension setting causes an error.

In other SQL statements such as Select or Insert, users can specify an extension other than the Data File Extension setting. The Data File Extension setting is used when no extension is specified.

Decimal Symbol: A setting that specifies the decimal separator used when data is stored (may be a comma or a period). The international decimal symbol (.) must be used in DML statements and parameter buffers.

International Sort: A setting to indicate the order in which records are retrieved when you issue a Select statement with an Order By clause. Clear this box to use ASCII sort order (the default setting). This order sorts items alphabetically with uppercase letters preceding lowercase letters. For example, "A, b, C" would be sorted as "A, C, b."

Select this box to use international sort order as defined by your operating system. This order is always alphabetic, regardless of case; the letters from the previous example would be sorted as "A, b, C." See your operating system documentation concerning the sorting of accented characters.

Use Long Qualifiers: Set this check box to use long path names as table qualifiers. When you set this check box, path names can be up to 255 characters. The default length for path names is 128 characters.

Allow Update and Delete: Specifies whether a data source allows Update and Delete statements. The

default is 0. Because Update and Delete statements cause immediate changes to a table, only one connection at a time can operate on a table. When this option is set, tables are opened exclusively by the current connection. Each update and delete on a text file can cause significant changes to the file, and performance may be poor. Consider a more appropriate database form if performance is a significant factor.

Application Using Threads: A setting that ensures that the driver works with multi-threaded applications. You can clear this check box when using the driver with single-threaded applications. Turning off this setting avoids additional processing required for ODBC thread safety standards.

Define Button

Displays the Define File dialog box, where you can select a file to define in the Define Table dialog box.

Translate Button

Displays the Select Translator dialog box, where you can translate your data from one character set to another. Choose the INTERSOLV OEM ANSI translator to translate your data from the IBM PC character set to the ANSI character set.

OK

Cancel

Apply

Define Table Dialog Box

Use the Define Table dialog box to define the structure of a text file by specifying values as follows:

Table Information pane

Database Name and **File:** The name of the database directory and file you selected on the Define File dialog box.

Table: Specify a name. Values may be up to 32 characters long and must be unique. This name is returned by SQLTables. By default, it is the file name without its extension.

Column Names in First Line: Check this box if the first line of the file contains column names.

Table Type: Select a value from the drop-down list to specify the table type. You can select from: Comma, Tab, Fixed-length, Character, or Stream.

Delimiter Character: If the table type is Character, specify the delimiter used in character-separated files.

Decimal Symbol: Type a comma to store the data using a comma as the separator for decimal numbers.

Column Information pane

Columns: A list box that contains the names of the columns in the table.

Name: Contains the name of the column selected in the Columns list box, or you can enter a column name.

Type: A drop-down list where you specify the data type of the field: Varchar, Numeric, or Date. If the field type is Date, you must select a mask for the field or type one in.

Mask: If the field type is Date, you must select a date mask. See Date Masks for more information.

Precision: Contains the precision of the column. The precision of numeric data types is defined as the maximum number of digits used by the data type of the column. For character types, this is the length in characters of the data; for binary data types, precision is defined as the length in bytes of the data. For time, timestamp, and all interval data types, precision is the number of characters in the character representation of this data.

Scale: Contains the scale of the column. The scale of decimal and numeric data types is defined as the maximum number of digits to the right of the decimal point. For approximate floating point number columns, the scale is undefined, since the number of digits to the right of the decimal point is not fixed. For datetime or interval data that contains a seconds component, the scale is defined as the number of digits to the right of the decimal point in the seconds component of the data.

Length: Available if you specified a fixed-length table. Specifies the number of bytes the data takes up in storage.

Offset: Available if you specified a fixed-length table. Specifies the number of bytes from the start of the table to the start of the field.

Guess Button

Available for *comma-separated*, *tab-separated*, or *character-separated* table types. Displays the drive's guess at the field definitions described above.

Parse Button

Available for *fixed-length* and *stream* tables. Displays the Parse Table dialog box, where you can define the table columns.

Add Button

Adds a column definition and displays the name in the Columns listbox.

Modify Button

Modifies the definition of the column selected in the Columns listbox with the values you specify.

Remove Button

Removes the column selected in the Columns listbox.

OK

Saves the table definition and closes the Define Table dialog box.

Cancel

Closes the Define Table dialog box without defining a table.

Parse Table Dialog Box

This dialog box displays the first line of the file. You must mark where *each* field begins and ends by enclosing it in brackets. These brackets indicate the position and length of each field value in the record. Click **OK** to close the Parse Table dialog box and save your specifications.

The *stream* table type has no delimiter characters between columns nor between records. The table is one long, continuous stream of data.

Apply Button

Writes the settings you have specified to the system information. These settings remain in effect until you change them in this dialog box. Clicking **Cancel** does not affect settings that have been applied.

OK Button

Writes the settings you have specified to the system information and closes the dialog box.

Cancel Button

Closes the dialog box without saving settings that have not been applied.