Data Server Utility Programs

Relativity for UNIX

Contents

Relativity Data Server Utility Programs

Several utility programs are supplied with the Relativity Data Client for UNIX to facilitate the creation of client and server data sources, the testing of the connection between the UNIX client and the server machine running the Relativity Data Server, and maintaining the drivers.

They include the following:

- *RelServerAdmin utility*. Use the Relativity Server Administrator utility program (`relserveradmin` or `relserveradmin64`) to manage data sources on the server and store server definitions on the UNIX client. (This utility is also included in the Relativity Data Server for UNIX product.)
- *RelClientAdmin utility*. Use the Relativity Client Data Source Administrator utility program (`relclientadmin` or `relclientadmin64`) to manage client data sources and server definitions on the UNIX client.
- *RelClientTest utility*. Use the Relativity Client Data Source Test utility program (`relclienttest` or `relclienttest64`) to test a client data source and its connection to a server data source, and to issue simple SQL statements to the Relativity Data Server.
- *RelDriverAdmin utility*. Use the Relativity Client Driver Administrator utility program (`reldriveradmin` or `reldriveradmin64`) to maintain the Relativity Data Client driver on a UNIX client. This utility is primarily used during the install and uninstall processes of the Relativity Data Client.

Command Line Interface

The utility programs that are supplied with the Relativity Data Client for UNIX employ a command-line interface. Technically, each element typed on the command line is a parameter; however, the following terminology is used in order to distinguish among the various types of parameters:

- `programname` refers to the name of the utility program; for example, `relserveradmin`, `relclientadmin64`, and so on.

  > **Note:** On the command line, all utility program names must be entered in lowercase characters. However, this document uses a mixed case convention when referring to the abbreviated names of the utilities. For example, the shortened name of the Relativity Server Administrator utility is RelServerAdmin utility.

- `command` refers to the administrative function to be performed by the utility; for example:

  `--add-data-source`

- `parameter` refers to additional information associated with a command; for example:

  `--server-name ServerName`

Each of the utility programs is invoked by typing its program name on the command line, followed by a command or one or more parameters. Each command and parameter supported by the utility program has two forms: a verbose form, which is introduced by two dashes, and a terse form, which is introduced by a single dash. Both forms may be used on the same command line.

Most, but not all, parameters have values that are associated with them. The meanings of these values are discussed below:

- `ServerName` identifies a Relativity Data Server on the network. `ServerName` is composed of two components: `NetworkName` and `ServicePort`.

  - `NetworkName` is the name of the UNIX or Windows server on the network. A `NetworkName` is composed of alphanumeric characters.

- `ServicePort` is the TCP/IP service port that a Relativity Data Server running on the server `NetworkName` is configured to use. A `ServicePort` is an unsigned decimal number. The default for `ServicePort` is 1583.

The format for `ServerName` is `NetworkName.ServicePort`. When `ServicePort` is omitted, the delimiting period should be omitted as well.
- `FileName` specifies a filename, in the format for the operating system on which the Relativity Data Server is running.
- `Text` indicates one or more words of text. If Text contains spaces, delimit Text with quotation marks. See the individual parameters under each command for the meaning of the Text value.
- `UnsignedInteger` specifies a zero or positive decimal number.
- `SignedInteger` specifies a decimal number, which may contain a leading dash to indicate a negative number.

Conventions

In this chapter, the following conventions for using command-line parameters are observed:

- Parameters or values that are optional are enclosed in square brackets, [ ].
- Parameters or values that are required, and where only one may appear, are enclosed in braces, { }, with the individual items in the list separated by vertical bars, | .
- Each parameter is described using its verbose form, with the terse form following in parentheses. Many, but not all, of the terse forms are merely the first letter of each word of the verbose form.

Displaying Help

If you supply the parameter `--help (-h)` in every command, the utility displays additional information and documentation on the command and its parameters. If you supply the --help parameter without a command, only a list of the supported commands is displayed.

The syntax of this command is as follows:

```
programname [command] --help
```

RelServerAdmin Utility

The Relativity Server Administrator utility program, `relserveradmin` or `relserveradmin64`, supports many administrative functions on the Relativity Data Server and can be used to maintain data sources on a Relativity Data Server. This utility can also be used to store the server definitions (server network names and service port numbers) of Relativity Data Servers on the local client machine, so that the *RelClientAdmin utility* can display the servers in its list.

The following sections describe how to use the commands and associated parameters for this utility, and the additional Micro Focus COBOL file manager configuration parameters you may need.

Using RelServerAdmin Utility Commands and Parameters

Each administrative function, or command, performed by the `RelServerAdmin` utility is addressed separately, in order to document clearly the additional parameters that the command supports.

The following table lists the commands supported by the `RelServerAdmin` utility.

Verbose	Terse	Description
--add-data-source	-ads	Adds a new server data source.
--add-server	-as	Adds a data server definition.
--list-character-sets	-lcs	Lists all of the character sets that a data server supports.
--list-data-sources	-lds	Lists all existing server data sources.
--list-servers	-ls	Lists all data server definitions.
--modify-data-source	-mds	Modifies an existing server data source.
--remove-data-source	-rds	Removes an existing server data source.
--remove-server	-rs	Removes a data server definition.
--show-data-source	-sds	Shows an existing server data source definition.
--show-server-status	-sss	Shows a data server's current status and lists all of its current connections.
--shutdown-server	-ss	Shuts down a data server.
--terminate-server-connection	-tsc	Terminates a current connection on a data server.

Adding a Data Source

A data source contains information necessary for the Relativity Data Server to open a data source requested by a Relativity Data Client. The `--add-data-source` command creates a new data source on the Relativity Data Server.

To add a new data source on the Relativity Data Server, use the following command:

```
relserveradmin[64] --add-data-source Text
   [--server-name ServerName]
   --catalog FileName
   [--data-source-desc Text]
   [--create]
   [--query-plan {Y|N|C}]
   [--query-plan-file FileName]
   [--character-set SignedInteger]
   [--cat-format {IDX4|IDX8|DEFAULT|BTRIEVE|CISAM|LEVELII|UnsignedInteger}]
   [--new-table-format {IDX4|IDX8|DEFAULT|BTRIEVE|CISAM|LEVELII|
UnsignedInteger}]
   [--data-compress UnsignedInteger]
   [--sparse-byte UnsignedInteger]
   [--trailing-spaces {Y|N}]
   [--leading-chars {Y|N}]
   [--compress-duplicate-keys {Y|N}]
```

where:

- The value of the `--add-data-source (-ads)` command specifies the name of the new data source.
- The value of the `--server-name (-s)` parameter indicates the Relativity Data Server on which to add the data source.
- The value of the `--catalog (-c)` parameter specifies the filename of the data source's catalog.
- The value of the `--data-source-desc (-desc)` parameter specifies the description to give to the new data source.
- The presence of the `--create (-cr)` parameter specifies that the new data source's catalog file should be created.
- The value of the `--query-plan (-qp)` parameter specifies whether the new data source should generate query plans. The value of Y will cause query plans always to be generated. The value of N will cause query plans never to be generated. The value of C will allow the generation of query plans to be controlled by the `--query-plan` parameter of the client data source.
- The value of the `--query-plan-file (-qpf)` parameter specifies the filename to receive generated query plans. This parameter is required only if the value of the `--query-plan` parameter is Y or C
- The value of the `--character-set (-cs)` parameter specifies the number of the character set to use with the new data source. If this parameter is absent, the default character set will be used.
- The value of the `--cat-format (-cf)` parameter specifies the type of Micro Focus COBOL file that should be used when creating a new catalog file. The valid values are DEFAULT, BTRIEVE, CISAM, LEVELII,IDX4, and IDX8. For more information, see *Catalog Format*.
- The value of the `--new-table-format (-ntf)` parameter specifies the type of Micro Focus COBOL file that should be used when creating a new file because of a CREATE TABLE SQL statement. The valid values are DEFAULT, BTRIEVE, CISAM, LEVELII, IDX4, and IDX8. For more information, see *New File Format*.
- The value of the `--data-compress (-dc)` parameter specifies the decimal value of the data compression algorithm to be configured for a new data file created as the result of a CREATE TABLE SQL statement. For more information, see *Data Compression*.
- The value of the `--sparse-byte (-sb)` parameter specifies the decimal value of the sparse byte to be configured for key compression in a new data file created as the result of a CREATE TABLE SQL statement. For more information, see *Key Compression - Sparse Byte*.
- The value of the `--trailing-spaces (-ts)` parameter specifies whether trailing spaces are to be compressed in keys in a new data file created as the result of a CREATE TABLE SQL statement. For more information, see *Key Compression - Trailing Spaces*.
- The value of the `--leading-chars (-lc)` parameter specifies whether leading characters are to be compressed in keys when creating a new file because of a CREATE TABLE SQL statement. For more information, see *Key Compression - Leading Characters*.

- The value of the `--compress-duplicate-keys (-cdk)` parameter specifies whether duplicate keys are to be compressed in a new data file created as the result of a CREATE TABLE SQL statement. For more information, see *Key Compression - Duplicates*.

Example

```
$ relserveradmin --add-data-source Verify
   --server-name MyDataServer
   --catalog demo/cobol/verify/verify.rcg

Data source 'Verify' added on MyDataServer.
```

Adding a Server Definition

A server definition is merely a `ServerName` that has been recorded using this command. The Relativity Server Administrator utility can maintain a set of server definitions for the current user.

> **Note:** The set of server definitions is for a future enhancement of the Relativity Server Administrator utility. This set of definitions and this command are of limited use at this time. However, this command's `--default` parameter is very helpful, particularly if you are dealing with only one Relativity Data Server. Setting a `ServerName` as the default will eliminate the need to supply the `--server-name` parameter in other commands.

To add a new server definition, use the following command:

```
relserveradmin[64] --add-server ServerName [--default]
```

where:

- The value of the `--add-server (-as)` command, `ServerName`, is the network name and service port of the Relativity Data Server to add to the list of server definitions.
- The presence of the `--default (-d)` parameter indicates that `ServerName` is to be the default for the `--server-name` parameter in other commands.

Example:

```
$ relserveradmin --add-server MyDataServer.1583 --default

Created Relativity data server definition for MyDataServer.1583.
```

Listing Character Sets

A character set is an interpretation of the bytes within a data file that represent characters, typically the items with a picture clause of X. Each data file has a character set associated with it, and this information is stored as a fixed attribute within a Relativity database. Relativity is capable of translating these characters from the character set of the data file to the character set of the ODBC application. This command supplies a list of the character sets for which the data server can supply translation.

To display a list of the character sets that a Relativity Data Server supports, use the following command:

```
relserveradmin[64] --list-character-sets [ServerName]
   [--server-name ServerName]
```

where:

- The value of the `--list-character-sets (-lcs)` command indicates the Relativity Data Server for which to list character sets.
- The value of the `--server-name (-s)` parameter may also be used to indicate the Relativity Data Server for which to list character sets.

Example:

The number to the left of each character set name is the identifier of the character set. Use this value in the `--character-set` parameter of the `--add-data-source` command and the `--modify-data-source` command.

```
$ relserveradmin --list-character-sets MyDataServer

Character sets on data server MyDataServer.
-28. Hebrew 7-bit Old Code
-25. MF EBCDIC (UNIX)
-24. MF EBCDIC (DOS)
-23. RM EBCDIC
-22. IBM-850
-21. SCO Latin 1
-20. STANDARD-2
-16. ISO 8859-16
-15. ISO 8859-15
-14. ISO 8859-14
-13. ISO 8859-13
-11. ISO 8859-11
-10. ISO 8859-10
-9. ISO 8859-9
-8. ISO 8859-8
-7. ISO 8859-7
-6. ISO 8859-6
-5. ISO 8859-5
-4. ISO 8859-4
-3. ISO 8859-3
-2. ISO 8859-2
-1. ISO 8859-1
37. EBCDIC US Canada
.
.
.
1250. Windows Latin 2
1251. Windows Cyrillic
1252. Windows Latin 1
1253. Windows Greek
1254. Windows Latin 5
1255. Windows Hebrew
1256. Windows Arabic
1257. Windows Baltic Rim
1258. Windows 1258
```

Note: Those character sets that represent Windows codepages have a positive value the same as their codepage number. Negative numbers represents all other character sets.

Listing Data Sources

A data source name is an identifier that ODBC applications use to access a Relativity database. This list of data source names represents the data sources on the Relativity Data Server to which a Relativity Data Client data source may be associated.

To display a list of the data source names on a Relativity Data Server, use the following command:

```
relserveradmin[64] --list-data-sources [ServerName]
   [--server-name ServerName]
```

where:

- The value of the `--list-data-sources` (`-lds`) command indicates the Relativity Data Server for which to list data sources.

- The value of the `--server-name (-s)` parameter may also be used to indicate the Relativity Data Server for which the data sources are to be listed.

Example:

```
$ relserveradmin --list-data-sources MyDataServer

Data sources on data server MyDataServer.
Shirt-3
Verify
```

Listing Server Definitions

A server definition is simply a `ServerName` that has been recorded using the `--add-server` command. The RelServerAdmin utility can maintain a set of server definitions for the current user.

Note: The set of server definitions is for a future enhancement of the Relativity Server Administrator utility. This set of definitions and this command are of limited use at this time.

To display a list of the current set of server definitions defined on the UNIX client, use the following command:

```
relserveradmin[64] --list-servers
```

Example:

```
$ relserveradmin --list-servers

Relativity data servers defined on this client.
tcp:MyDataServer.1583
```

Modifying a Data Source

A data source contains information necessary for the Relativity Data Server to open a data source requested by a Relativity Data Client. This command alters an existing data source on the data server.

To modify a data source on a Relativity Data Server, use the following command. Note that parameter values that are not specified in the command retain their previous value in the data source.

```
relserveradmin[64] --modify-data-source Text
    [--server-name ServerName]
    [--new-data-source-name Text]
    [--catalog FileName]
    [--data-source-desc Text]
    [--create]
    [--query-plan {Y|N|C}]
    [--query-plan-file FileName
    [--character-set SignedInteger]
    [--cat-format {IDX4|IDX8|DEFAULT|BTRIEVE|CISAM|LEVELII|UnsignedInteger}]
    [--new-table-format {IDX4|IDX8|DEFAULT|BTRIEVE|CISAM|LEVELII|
UnsignedInteger}]
    [--data-compress UnsignedInteger]
    [--sparse-byte UnsignedInteger]
    [--trailing-spaces {Y|N}]
    [--leading-chars {Y|N}]
    [--compress-duplicate-keys {Y|N}]
```

where:

- The value of the `--modify-data-source (-mds)` command specifies the current name of the data source.
- The value of the `--server-name (-s)` parameter indicates the Relativity Data Server on which to modify the data source.

- The value of the `--new-data-source-name (-ndsn)` parameter specifies the new name of the data source.
- The value of the `--catalog (-c)` parameter specifies the filename of the data source's catalog.
- The value of the `--data-source-desc (-desc)` parameter specifies the description to give to the data source.
- The presence of the `--create (-cr)` parameter specifies that the data source's catalog file should be created.
- The value of the `--query-plan (-qp)` parameter specifies whether the data source should generate query plans. The value of Y will cause query plans always to be generated. The value of N will cause query plans never to be generated. The value of C will allow the generation of query plans to be controlled by the `--query-plan parameter` of the client data source.
- The value of the `--query-plan-file (-qpf)` parameter specifies the filename to receive generated query plans. This parameter is required only if the value of the --query-plan parameter is Y or C
- The value of the `--character-set (-cs)` parameter specifies the number of the character set to use with the data source.
- The value of the `--cat-format (-cf)` parameter specifies the type of Micro Focus COBOL file that should be used when creating a new catalog file. The valid values are DEFAULT, BTRIEVE, CISAM, LEVELII, IDX4, and IDX8. For more information, see *Catalog Format*.
- The value of the `--new-table-format (-ntf)` parameter specifies the type of Micro Focus COBOL file that should be used when creating a new data file as a result of the CREATE TABLE SQL statement. The valid values are DEFAULT, BTRIEVE, CISAM, LEVELII, IDX4, and IDX8. For more information, see *New File Format*.
- The value of the `--data-compress (-dc)` parameter specifies the decimal value of the data compression algorithm to be configured for a new data file created as the result of the CREATE TABLE SQL statement. For more information, see *Data Compression*.
- The value of the `--sparse-byte (-sb)` parameter specifies the decimal value of the sparse byte to be configured for key compression in a new data file created as a result of the CREATE TABLE SQL statement. For more information, see *Key Compression - Sparse Byte*.
- T he value of the `--trailing-spaces (-ts)` parameter specifies whether trailing spaces are to be compressed in keys in a new data file created as a result of the CREATE TABLE SQL statement. For more information, see *Key Compression - Trailing Spaces*.
- T he value of the `--leading-chars (-lc)` parameter specifies whether leading characters are to be compressed in keys in a new data file created as a result of the CREATE TABLE SQL statement. More information is available in *Key Compression - Leading Characters*.
- The value of the `--compress-duplicate-keys (-cdk)` parameter specifies whether duplicate keys are to be compressed in a new data file created as a result of the CREATE TABLE SQL statement. For more information, see *Key Compression - Duplicates*.

Example:

```
$ relserveradmin --modify-data-source Verify
    --server-name MyDataServer --new-data-source-name "New Verify"

Data source 'Verify' modified on server MyDataServer.
```

Removing a Data Source

To remove a data source on a Relativity Data Server, use the following command:

```
relserveradmin[64] --remove-data-source Text
    [--server-name ServerName]
```

where:

- The value of the `--remove-data-source (-rds)` command specifies the name of the data source to remove.

- The value of the `--server-name (-s)` parameter indicates the Relativity Data Server from which to remove the data source.

Example:

```
$ relserveradmin --remove-data-source "New Verify"
    --server-name MyDataServer

Removed data source 'New Verify' from MyDataServer.
```

Removing a Server Definition

A server definition is merely a `ServerName` that has been recorded using the `--add-server` command. The `RelServerAdmin` utility can maintain a set of server definitions for the current user.

Note: The set of server definitions is for a future enhancement of the Relativity Server Administrator utility. This set of definitions and this command are of limited use at this time.

To remove a server definition, use the following command:

```
relserveradmin[64] --remove-server ServerName
```

where:

- The value of the `--remove-server (-rs)` command, `ServerName`, is the network name and service port of the Relativity Data Server to remove from the list of server definitions.

Example:

```
$ relserveradmin --remove-server MyDataServer.1583

Relativity data server definition for MyDataServer.1583 removed.
```

Showing a Data Source

A data source contains information necessary for the Relativity Data Server to open a data source requested by a Relativity Data Client. This command shows the current contents of the data source.

To display a data source on a Relativity Data Server, use the following command:

```
relserveradmin[64] --show-data-source Text
    [--server-name ServerName] [-env]
```

where:

- The value of the `--show-data-source (-sds)` command specifies the name of the data source to show.
- The value of the `--server-name (-s)` parameter indicates the Relativity Data Server for which to show the data source.
- The presence of the `--env (-e)` parameter specifies that the information from the data source be printed to standard output in the form of Bourne shell environment variable assignments. The purpose is to emit the information in a form that will be usable in a script to manipulate or copy an existing data source. The environment variable names and their corresponding parameter options are given in the following table:

Variable Name	Parameter
DS_CATALOG	--catalog
DS_DESC	--data-source-desc
DS_QP	--query-plan

Variable Name	Parameter
DS_QPF	--query-plan-file
DS_CS	--character-set
DS_CATF	--cat-format
DS_NEWF	--new-table-format
DS_DC	--data-compress
DS_SB	--sparse-byte
DS_TS	--trailing-spaces
DS_LC	--leading-chars
DS_DUP	--compress-duplicate-keys

Example:

```
$ relserveradmin --show-data-source Shirt-3
   --server-name MyDataServer

Data source 'Shirt-3' on MyDataServer:
DBQ=Shirt_3
Description=Sample Shirt-3 Data Source
Catalog=demo/cobol/shirt-3/shirt3.rcg
FileMgr=MF
```

Showing the Status of a Data Server

A Relativity Data Server's status consists of several pieces of information, including when it started, the number of clients that are connected to it, the maximum number of clients that can connect to it, the number of connections that it is servicing, and the actual connections themselves. The connection information consist of the ID of the UNIX or Windows process that is servicing the connection, the name of the client machine, the user name of client, and the name of ODBC application program. (Not all ODBC applications supply the application name.)

To show the status of a Relativity Data Server, and to display a list of connections it is currently servicing, use the following command.

```
relserveradmin[64] --show-server-status [ServerName]
   [--server-name ServerName]
```

where:

- The value of the --show-server-status (-sss) command indicates which Relativity Data Server's status to show.
- The value of the --server-name (-s) parameter may also be used to indicate which Relativity Data Server's status to show.

Example:

```
$ relserveradmin --show-server-status MyDataServer

Status of server MyDataServer.
Server Start Time: Thu Jun 24 13:58:43 2004.
Current number of connections: 2.
Process ID Client Machine (User Name) Application
     28446 MyDataClient (db3 owner)       Microsoft Access for Windows
     28450 MyDataClient (db3 owner)       Unknown app
```

Shutting Down a Data Server

The `--shutdown-server` command will terminate a Relativity Data Server. This command has the same effect as executing the ./STOP script on the Relativity Data Server for UNIX or stopping the service on the Relativity Data Server for Windows.

Note: If a Relativity Data Server for UNIX is configured to start automatically at machine boot, the `--shutdown-server` command will shut down the data server, but the UNIX operating system will automatically restart it.

To shut down or stop a running Relativity Data Server, use the following command:

```
relserveradmin[64] --shutdown-server [ServerName]
   [--server-name ServerName]
   {--immediately | --gracefully}
```

where:

- The value of the `--shutdown-server` (`-ss`) command may also be used to indicate the Relativity Data Server to shut down.
- The value of the `--server-name` (`-s`) parameter may also be used to indicate the Relativity Data Server to shut down.
- The presence of the `--immediately` parameter indicates that the Relativity Data Server is to shut down immediately, summarily disconnecting existing client connections.
- The presence of the `--gracefully` parameter indicates that the Relativity Data Server is to refuse new connections and allow existing connections to disconnect normally, before terminating itself.

Example:

```
$ relserveradmin --shutdown-server MyDataServer --gracefully

Server MyDataServer requested to shut down gracefully.
```

Terminating a Connection to a Data Server

Each client connection to a Relativity Data Server has a unique ID that can be used to identify the connection. This ID is supplied when terminating a client's connection.

Note: The `--terminate-server-connection` command will not force the client connection to terminate. It will merely request that the connection be terminated. This request will be honored when the current request from the client is complete, or if the Relativity Data Server is idle waiting for the next request from the client. This command will not interrupt a current client request.

To terminate a client's connection to a Relativity Data Server, first use the `--show-server-status` command to determine the process ID of the connection to terminate, and then use the following command:

```
relserveradmin[64] --terminate-server-connection UnsignedInteger
   [--server-name ServerName]
```

where:

- The value of the `--terminate-server-connection` (`-tsc`) command specifies the process ID of the connection to terminate.
- The value of the `--server-name` (`-s`) parameter indicates the Relativity Data Server on which to terminate a connection.

Example:

```
$ relserveradmin --terminate-server-connection 28446
   --server-name MyDataServer
```

```
Terminated connection for process ID 28446.
```

Micro Focus COBOL File Manager Configuration Parameters

This section explains the configuration parameters for access to Micro Focus COBOL files. See the documentation supplied with your Micro Focus development product for more information on specific formats.

Catalog Format

This configuration parameter controls the format of the Micro Focus COBOL index file that will be used when opening or creating a catalog. The available entries are DEFAULT, BTRIEVE, CISAM, LEVELII, IDX4, and IDX8. This configuration parameter cannot be used to change the format of an existing catalog. DEFAULT specifies the default Micro Focus COBOL file format for this operating system. BTRIEVE specifies the Btrieve file format. CISAM species the C-ISAM file format. LEVELII specifies the Micro Focus COBOL Level II indexed file format. IDX4 specifies the catalog as an optimized form of the format used by this system, for fast duplicate key handling. IDX8 specifies the catalog as a large file format file.

New File Format

This configuration parameter controls the format of the Micro Focus COBOL index file that will be used when creating new files because of a CREATE TABLE SQL statement. The available entries are DEFAULT, BTRIEVE, CISAM, LEVELII, IDX4, and IDX8. DEFAULT specifies the default Micro Focus COBOL file format for this operating system. BTRIEVE specifies the Btrieve file format. CISAM specifies the C-ISAM file format. LEVELII specifies the Micro Focus COBOL Level II indexed file format. IDX4 specifies the catalog as an optimized form of the format used by this system for fast duplicate key handling. IDX8 specifies the file as a large file format file.

Data Compression

This configuration parameter controls the type of data compression to be used when new files are created because of a CREATE TABLE SQL statement. This parameter is the same as the DATACOMPRESS File Handler configuration option. The value of 0 indicates that no compression of data records is to take place. The value of 1 indicates that compression algorithm 1 is to be used.

Key Compression - Trailing Spaces

This configuration parameter controls whether compression of trailing spaces is to occur on the keys of new files that are created because of a CREATE TABLE SQL statement. The value Y indicates that trailing spaces are to be compressed. The default is N, which indicates not to compress trailing spaces.

Key Compression - Leading Characters

This configuration parameter controls whether compression of leading characters is to occur on the keys of new files that are created because of a CREATE TABLE SQL statement. The value Y indicates that leading characters that are the same as in the previous key are compressed. The default is N, which indicates not to compress leading characters.

Key Compression - Duplicates

This configuration parameter controls whether compression of duplicate keys is to occur on the keys of new files that are created because of a CREATE TABLE SQL statement. The value Y indicates that the repetitions of duplicate keys are compressed. The default is N, which indicates not to compress duplicate keys.

Key Compression - Sparse Byte

This configuration parameter specifies the decimal number of the sparse character for keys of new files that are created because of creating tables. If this parameter is specified, keys in the file that contain only this value are suppressed. The default is not to suppress keys made up of the sparse character.

RelClientAdmin Utility

The Relativity Client Data Source Administrator utility program, `relclientadmin` or `relclientadmin64`, is used to maintain data sources on a UNIX client. This utility can also be used to store the server network name and service port number of Relativity Data Servers on the UNIX client.

The following sections describe the commands and their associated parameters for this utility.

Using the RelClientAdmin Utility Commands and Parameters

Each administrative function, or command, performed by the `RelClientAdmin` utility is addressed separately, in order to document clearly the additional parameters that the command supports.

The following table lists the commands supported by the `RelClientAdmin` utility.

Verbose	Terse	Description
--add-data-source	-ads	Adds a new client data source.
--add-server	-as	Adds a data server definition.
--list-data-sources	-lds	Lists all existing client data sources.
--list-server-character-sets	-lscs	Lists all of the character sets that a data server supports.
--list-server-data-sources	-lsds	Lists all existing data sources on a data server.
--list-servers	-ls	Lists all data server definitions.
--modify-data-source	-mds	Modifies an existing client data source.
--remove-data-source	-rds	Removes an existing client data source.
--remove-server	-rs	Removes a data server definition.
--show-data-source	-sds	Shows an existing client data source definition.

Adding a Data Source

A data source contains information necessary for the Relativity Data Client to open a data source requested by an ODBC application. This command creates a new data source on the client.

To add a data source to a UNIX client, use the following command:

```
relclientadmin[64] --add-data-source Text
    [--driver-name Text]
    [--server-name ServerName]
    --server-data-source-name Text
    [--type {U|S}]
    [--data-source-desc Text]
```

```
[--query-plan {Y|N}]
[--character-set SignedInteger]
```

where:

- The value of the `--add-data-source (-ads)` command specifies the name of the new data source.
- The value of the `--driver-name (-drv)` parameter indicates the name of the ODBC Driver with which to create the new data source. The default is "Relativity Client".
- The value of the `--server-name (-s)` parameter indicates the Relativity Data Server to which to associate the data source.
- The value of the `--server-data-source-name (-sdsn)` parameter indicates the server data source on the data server to which to associate the data source. Use the `--list-server-data-sources` command to obtain a list of data sources on the data server.
- The value of the `--type (-t)` parameter specifies the type of the data source. The value U indicates a user data source. The value of S indicates a system data source.
- The value of the `--data-source-desc (-desc)` parameter specifies the description to give to the new data source.
- The value of the `--query-plan (-qp)` parameter specifies whether the new data source should generate query plans. The value of Y will cause query plans always to be generated. The value of N will cause query plans never to be generated. In order for this parameter to be effective, the server data source must have a `--query-plan` parameter of C, which allows the client data source to control the creation of query plans.
- The value of the `--character-set (-cs)` parameter specifies the number of the character set to use with the new data source. If this parameter is absent, the default character set will be used. Use the `--list-server-character-sets` command to obtain a list of character sets on the data server.

Example:
```
$ relclientadmin --add-data-source Verify
   --server-name MyDataServer
   --server-data-source-name Verify

Data source 'Verify' added.
```

Adding a Server Definition

A server definition is merely a `ServerName` that has been recorded using this command. The Relativity Client Data Source Administrator utility can maintain a set of server definitions for the current user.

> **Note:** The set of server definitions is for a future enhancement of the Relativity Client Administrator utility. This set of definitions and this command are of limited use at this time. However, this command's `--default` parameter is very helpful, particularly, if you are dealing with only one Relativity Data Server. Setting a `ServerName` as the default will eliminate the need to supply the `--server-name` parameter in other commands.

To add a new server definition, use the following command:
```
relclientadmin[64] --add-server ServerName [--default]
```

where:

- The value of the `--add-server (-as)` command, `ServerName`, is the network name and service port of the Relativity Data Server to add to the list of server definitions.
- The presence of the `--default (-d)` parameter indicates that this server is to be the default for the `--server-name` parameter in other commands.

Example:

```
$ relclientadmin --add-server MyDataServer.1583

Created Relativity data server definition for MyDataServer.1583.
```

Listing Data Sources

A data source name is an identifier that ODBC applications use to access a Relativity database. This list of data source names represents the data sources on the UNIX client that an ODBC application may open.

To display a list of the data sources on a UNIX client, use the following command:

```
relclientadmin[64] --list-data-sources
    [--type {U|S|B}]
```

where:

- The value of the `--type` (`-t`) parameter specifies the type of data sources to list. The value U indicates a user data source. The value of S indicates a system data source. The value B indicates both user and system data sources. The default is to list both.

Example:

```
$ relclientadmin --list-data-sources

Data sources on data client.
_____User Data Sources_____
Shirt-3
_____System Data Sources_____
Verify
```

Listing Server Character Sets

A character set is an interpretation of the bytes within a data file that represent characters, typically the items with a picture clause of X. Each data file has a character set associated with it, and this information is stored as a fixed attribute within a Relativity database. Relativity is capable of translating these characters from the character set of the data file to the character set of the ODBC application. This command supplies a list of the character sets for which the data server can supply translation.

To display a list of the character sets that a Relativity Data Server supports, use the following command:

```
relclientadmin[64] --list-server-character-sets [ServerName]
    [--server-name ServerName]
```

where:

- The value of the `--list-server-character-sets` (`-lscs`) command indicates the Relativity Data Server for which to list character sets.
- The value of the `--server-name` (`-s`) parameter may also be used to indicate the Relativity Data Server for which to list character sets.

Example:

The number to the left of each character set name is the character set's identifier. Use this value in the `--character-set` parameter of the `--add-data-source` and `--modify-data-source` commands.

```
$ relclientadmin --list-server-character-sets MyDataServer

Character sets on data server MyDataServer.
-28. Hebrew 7-bit Old Code
-25. MF EBCDIC (UNIX)
```

```
-24. MF EBCDIC (DOS)
-23. RM EBCDIC
-22. IBM-850
-21. SCO Latin 1
-20. STANDARD-2
-16. ISO 8859-16
-15. ISO 8859-15
-14. ISO 8859-14
-13. ISO 8859-13
-11. ISO 8859-11
-10. ISO 8859-10
-9. ISO 8859-9
-8. ISO 8859-8
-7. ISO 8859-7
-6. ISO 8859-6
-5. ISO 8859-5
-4. ISO 8859-4
-3. ISO 8859-3
-2. ISO 8859-2
-1. ISO 8859-1
37. EBCDIC US Canada
.
.
.
1250. Windows Latin 2
1251. Windows Cyrillic
1252. Windows Latin 1
1253. Windows Greek
1254. Windows Latin 5
1255. Windows Hebrew
1256. Windows Arabic
1257. Windows Baltic Rim
1258. Windows 1258
```

Note: Those character sets that represent Windows codepages have a positive value the same as their codepage number. Negative numbers represents all other character sets.

Listing Server Data Sources

A data source name is an identifier that ODBC applications use to access a Relativity database. This list of data source names represents the data sources on the Relativity Data Server to which a Relativity Data Client data source may be associated.

To display a list of the data source names on a Relativity Data Server, use the following command:

```
relclientadmin[64] --list-server-data-sources [ServerName]
   [--server-name ServerName]
```

where:

- The value of the `--list-server-data-sources` (`-lsds`) command indicates the Relativity Data Server for which to list the data sources.
- The value of the `--server-name` (`-s`) parameter may also be used to indicate the Relativity Data Server for which to list the data sources.

Example:

```
$ relclientadmin --list-data-sources MyDataServer

Data sources on data server MyDataServer.
Shirt-3
Verify
```

Listing Server Definitions

A server definition is merely a `ServerName` that has been recorded using the `--add-server` command of the Relativity Client Data Source Administrator utility. The Relativity Client Data Source Administrator utility can maintain a set of server definitions for the current user.

> **Note:** The set of server definitions is for a future enhancement of the Relativity Client Data Source Administrator utility. This set of definitions and this command are of limited use at this time.

To display a list of the current set of server definitions defined on the UNIX client, use the following command:

```
relclientadmin[64] --list-servers
```

Example:

```
$ relclientadmin --list-servers

Relativity data servers defined on this client.
tcp:MyDataServer.1583
```

Modifying a Data Source

A data source contains information necessary for the Relativity Data Client to open a data source requested by an ODBC application. This command alters an existing data source on the client.

To modify a data source on a UNIX client, use the following command. Note that parameter values that are not specified in the command retain their previous value in the data source.

```
relclientadmin[64] --modify-data-source Text
   [--new-data-source-name Text]
   [--driver-name Text]
   [--server-name ServerName]
   [--server-data-source-name Text]
   [--type {U|S}]
   [--data-source-desc Text]
   [--query-plan {Y|N}]
   [--character-set SignedInteger]
```

where:

- The value of the `--modify-data-source (-mds)` command specifies the current name of the data source.
- The value of the `--new-data-source-name (-ndsn)` parameter specifies the new name of the data source.
- The value of the `--driver-name (-drv)` parameter indicates the name of the ODBC Driver with which to create the modify data source. The default is the driver that was used to create the data source.
- The value of the `--server-name (-s)` parameter indicates the Relativity Data Server to which to associate the data source.
- The value of the `--server-data-source-name (-sdsn)` parameter indicates the server data source on the data server to which to associate the data source. Use the `--list-server-data-sources` command to obtain a list of data sources on the data server.
- The value of the `--type (-t)` parameter specifies the type of the data source indicated by the `--modify-data-source` parameter. The value U indicates a user data source. The value of S indicates a system data source. The default is a system data source.
- The value of the `--data-source-desc (-desc)` parameter specifies the description to give to the data source.

- The value of the `--query-plan (-qp)` parameter specifies whether the data source should generate query plans. The value of Y will cause query plans always to be generated. The value of N will cause query plans never to be generated. In order for this parameter to be effective, the server data source must have a `--query-plan` parameter value of C, which allows the client data source to control the creation of query plans.
- The value of the `--character-set (-cs)` parameter specifies the number of the character set to use with the data source. If this parameter is absent, the default character set will be used. Use the `--list-server-character-sets` command to obtain a list of character sets on the data server. For more information, see *Listing Server Character Sets*.

Example:

```
$ relclientadmin --modify-data-source Verify --query-plan Y

Data source 'Verify' modified.
```

Removing a Data Source

A data source contains information necessary for the Relativity Data Client to open a data source requested by an ODBC application. This command removes an existing data source on the client.

To remove a data source from a UNIX client, use the following command:

```
relclientadmin[64] --remove-data-source Text
   [--driver-name Text]
   [--type {U|S}]
```

where:

- The value of the `--remove-data-source (-rds)` command specifies the name of the data source to remove.
- The value of the `--driver-name (-drv)` parameter indicates the name of the ODBC Driver with which to remove the new data source. The default is the driver that was used to create the data source.
- The value of the `--type (-t)` parameter specifies the type of the data source indicated by the `--remove-data-source` parameter. The value U indicates a user data source. The value of S indicates a system data source. The default is a system data source.

Example:

```
$ relclientadmin --remove-data-source Verify

Removed data source 'Verify'.
```

Removing a Server Definition

A server definition is merely a `ServerName` that has been recorded using the `--add-server` command of the Relativity Client Data Source Administrator utility . The Relativity Client Data Source Administrator utility can maintain a set of server definitions for the current user.

🖉 **Note:** The set of server definitions is for a future enhancement of the Relativity Client Data Source Administrator. This set of definitions and this command are of limited use at this time.

To remove a server definition, use the following command:

```
relclientadmin[64] --remove-server ServerName
```

where:

- The value of the `--remove-server (-rs)` command specifies the network name and service port of the server definition to remove.

Example:

```
$ relclientadmin --remove-server MyDataServer.1583

Relativity data server definition for MyDataServer.1583 removed.
```

Showing a Data Source

A data source contains information necessary for the Relativity Data Client to open a data source requested by an ODBC application. This command shows the current contents of the data source.

To display a data source on a UNIX client, use the following command:

```
relclientadmin[64] –show-data-source Text
    [--type {U|S}]
    [-env]
```

where:

- The value of the `--show-data-source` (`-sds`) command specifies the name of the data source.
- The value of the `--type` (`-t`) parameter specifies the type of the data source indicated by the `--show-data-source` parameter. The value U indicates a user data source. The value of S indicates a system data source. The default is a system data source.
- The presence of the `--env` (`-e`) parameter specifies that the information from the data source be printed to standard output in the form of Bourne shell environment variable assignments. The purpose is to emit the information in a form that will be usable in a script to manipulate or copy an existing data source. The environment variable names and their corresponding parameters are given in the following table:

Variable Name	Parameter
DS_SERVER	--server-name
DS_SERVER_DS	--server-data-source-name
DS_DESC	--data-source-desc
DS_QP	--query-plan
DS_TYPE	--type
DS_CS	--character-set

Example:

```
$ relclientadmin --show-data-source Verify

Data source 'Verify':
Driver=Relativity Client
Description=Relativity C/S Data Source.
ServerName=MyDataServer.1583
ServerDSN=Verify
QryPlan=0
ArrayFetchOn=1
ArrayBufferSize=8
```

RelClientTest Utility

The Relativity Client Data Source Test utility program, `relclienttest` or `relclienttest64`, is used to test a client data source on the UNIX client and its connection to a server data source. This utility can also be used to issue limited SQL statements to the Relativity Data Server.

The following section describes the command and associated parameters for this utility.

Using the RelClientTest Utility Command and Parameters

The testing function, or command, performed by the `RelClientTest` utility is addressed separately, in order to document clearly the additional parameters that the command supports.

The following table lists the command supported by the `RelClientTest` utility.

Verbose	Terse	Description
--sql	-s	Executes an SQL SELECT statement on a client data source.

Testing Client Data Sources

To test a data source on a UNIX client, use the following command:

```
relclienttest[64] --sql Text
    --data-source-name Text
    [--user-name Text]
    [--password Text]
```

where:

- The value of the `--sql (-s)` command is the SQL statement to execute. Only SELECT statements are supported.
- The value of the `--data-source-name (-dsn or -d)` parameter is the name of the client data source.
- The value of the `--user-name (-u)` parameter is the user name with which to connect to the data source. If the `--user-name parameter` is not present, a connection will be attempted without a user name.
- The value of the `--password (-p)` parameter is the password with which to connect to the data source. If the `--password` parameter is not present, a connection will be attempted without a password.
- The value of the `--rows (-r)` parameter is the maximum number of rows to print from the SELECT statement specified with the `--sql` parameter. The default value for this parameter is 500.

Example:

```
$ relclienttest --sql "select * from backorder"
    --data-source-name Verify

ProductNumber|ProductSize|Color|PricePerUnit|Price4OrMore|
    BackOrderQuantity|DateStockExpected|
AP1927367D|20W |17|32|29.99|22|19940803|
AP1927466D|24W |03|12|10|0|19940705|
AP2823987D|ML  |03|19.99|17|20|19940721|
```

```
AP2824597D|ML  |17|28|25.99|30|19940701|
AP2824621D|MS  |45|20|16.99|120|19940824|
AP2824712D|MM  |03|12|10|26|19940724|
PF5430319B|S   |37|18|15.3|120|19940615|
PF5430442B|S   |37|16|13.6|100|19940615|
PF5430467B|M   |73|16|13.6|50|19940601|
PF5431036D|L   |78|22.99|18.79|2|19940630|
PF5432000D|L   |26|24|20.4|102|19940715|
PG5430418D|XLXT|34|20|17|40|19940630|
PG5430418D|XLXT|37|20|17|10|19940614|
PG5430418D|XLXT|73|20|17|130|19940701|
14 rows returned.
```

RelDriverAdmin Utility

The Relativity Client Driver Administrator utility program, `reldriveradmin` or `reldriveradmin64`, can be used to maintain the Relativity Data Client driver on a UNIX client. The primary use of this utility occurs during the install and uninstall processes of the Relativity Data Client.

The following sections describe the commands and their associated parameters for this utility.

Using the RelDriverAdmin Utility Commands and Parameters

Each administrative function, or command, performed by the `RelDriverAdmin` utility is addressed separately, in order to document clearly the additional parameters that the command supports.

The following table lists the commands supported by the `RelDriverAdmin` utility.

Verbose	Terse	Description
--install-driver	-id	Installs the Relativity Data Client's driver.
--list-drivers	-ld	Lists all drivers installed on the UNIX client.
--uninstall-driver	-ud	Uninstalls the Relativity Data Client's driver.

Installing the Relativity Data Client Driver

"Installing a driver" is the ODBC terminology for this function. This command does not actually install the library files for the Relativity Data Client; the installation script performs that function. Instead, this command registers the Relativity Data Client with the ODBC Driver Manager, so that data sources may be created with the Relativity Data Client.

To install the Relativity Data Client driver, use the following command:

```
reldriveradmin[64] --install-driver [Text]
    --directory Text
    [--driver-library Text]
    [--setup-library Text]
```

where:

- The value of the `--install-driver (-id)` command specifies the name of the driver. The default value is Relativity Client.
- The value of the `--directory (-dir)` parameter specifies the name of the directory in which the driver library files are located.
- The value of the `--driver-library (-dl)` parameter specifies the filename of the main driver library. The default value is `relclient.so`.
- The value of the `--setup-library (-sl)` parameter specifies the filename of the driver data source setup library. The default value is `relclnsu.so`.

Example:

```
$ reldriveradmin --install-driver "Relativity Client"
   --directory /usr/local/mf/lib

Driver 'Relativity Client' installed.
```

Listing the ODBC Drivers

This command prints a list of all the drivers installed in the Driver Manager, including the Relativity Data Client.

To list the installed ODBC drivers, use the following command:

```
reldriveradmin[64] --list-drivers
```

Example

```
$ reldriveradmin --list-drivers

Drivers on client.
Relativity Client
```

Uninstalling the Relativity Data Client Driver

This command does not actually uninstall the library files for the Relativity Data Client; the uninstall script performs that function. Instead, this command deregisters the Relativity Data Client with the ODBC Driver Manager.

To uninstall the Relativity Data Client driver, use the following command:

```
reldriveradmin[64] --uninstall-driver [Text]
```

where:

- The value of the `--uninstall-driver` (`-ud`) command specifies the name of the driver. The default value is Relativity Client.

Example

```
$ reldriveradmin --uninstall-driver "Relativity Client"

Removed driver 'Relativity Client'.
```

UNIX ODBC Driver Managers

This section discusses some of the implementation issues and differences between the ODBC Driver Managers available on UNIX.

Driver Managers on UNIX

An ODBC Driver Manager is not a standard part of any UNIX operating system. Two groups, however, have written open source implementations of a Driver Manager for UNIX, both of which have been tested with the Relativity Data Client for UNIX. These two open source implementations are:

- **iODBC**. iODBC was written by Ke Jin and enhanced by OpenLink Software, Inc. It was released as Open Source under GNU's Lesser General Public License (LGPL) in 1999. More information is available from the web site at *http://www.iodbc.org*.
- **unixODBC**. unixODBC was written by Peter Harvey and enhanced by Easysoft Limited. It was released under GNU's Lesser General Public License (LGPL) and General Public License (GPL). Further information is available from the web site at *http://www.unixodbc.com*.

> **Note:** You can read more about the GNU General Public License at *http://www.gnu.org/copyleft/ gpl.html* and about the GNU Lesser General Public License at *http://www.gnu.org/copyleft/lesser.html*. These licenses essentially grant the right to distribute the software protected by the license, as long as the source of the original software is also distributed, or made available for the duration of three years.

Differences

In general, these two Driver Managers, iODBC and unixODBC, implement the exact same Application Program Interface (API). However, there are some differences between the two implementations, which are discussed here and in the following topics.

> **Note:** Micro Focus does not recommend one implementation over the other.

Because of UNIX compilation mechanisms, an ODBC application that is linked with a specific Driver Manager library should load that same Driver Manager library when the application is executed. It is possible for an ODBC application to be able to load either Driver Manager, but most applications do not use this technique.

It is possible, using UNIX links, to allow an ODBC-enabled application to use a Driver Manager other than the one to which it was linked. However, because an application should use a Driver Manager with which it was tested, and given that the two Driver Manager libraries have different names, creating links is not recommended.

unixODBC

The unixODBC Driver Manager uses a set of files to store information about the data sources and drivers that it manages. Furthermore, it supports both user data sources (those that are visible only to the current user) and system data sources (those that are visible to all users). This section discusses how the unixODBC Driver Manager locates these control files.

Data Source Search Sequence

unixODBC stores information about data sources in a file named `odbc.ini`. There is a pair of these files: one for user data sources and one for system data sources. This section describes how unixODBC locates these files.

When unixODBC searches for a data source, it uses the following search sequence:

1. unixODBC looks for the environment variable, ODBCINI. If it is present, unixODBC retrieves its value and attempts to open a file by that name. If the file is present, unixODBC reads all of the data sources within that file into memory.
2. If the environment variable ODBCINI is not present, unixODBC looks up the home directory for the current user name. (Note that unixODBC is not using the HOME environment variable to do this.) unixODBC attempts to open a file with the name, `.odbc.ini`, in the home directory. If that file is present, unixODBC reads all of the data sources within that file into memory.
3. The system data source file is opened, and the data sources that are within it are appended to the data sources already in memory. The Driver Manager contains the location of the system data source file. If you compile unixODBC yourself, this location will be the directory specified to the configure script by the `--prefix` parameter, plus `/etc/odbc.ini`. If the `--prefix` parameter is not specified, the location will be `/usr/local/etc/odbc.ini`.
4. The requested data source is located from within the list of data sources stored in memory.

Driver Search Sequence

unixODBC stores information about drivers in a file named `odbcinst.ini`. This file is used to store the location of the Relativity Data Client driver shared library, among other things.

unixODBC expects this file to be in the same directory where the system data source file is located. If you compile unixODBC yourself, this location will be the directory specified to the configure script by the `--prefix` parameter, plus `/etc/odbcinst.ini`. If the `--prefix` parameter is not specified, the location will be `/usr/local/etc/odbcinst.ini`.

iODBC

The iODBC Driver Manager uses a set of files to store information about the data sources and the drivers that it manages. Furthermore, it supports both user data sources (those that are visible only to the current user) and system data sources (those that are visible to all users). iODBC also supports both the user and system driver control files. This section discusses how the iODBC Driver Manager locates the control files.

Data Source Search Sequence

iODBC stores information about data sources in a file named `odbc.ini`. There is a pair of these files: one for user data sources and one for system data sources. This section describes how iODBC locates these files.

When iODBC searches for a data source, it uses the following search sequence:

1. iODBC looks for the environment variable, ODBCINI. If it is present, iODBC retrieves its value and attempts to open a file by that name. If it succeeds, iODBC searches for the data source in this file.
2. If the environment variable ODBCINI is not present, iODBC retrieves the value of the HOME environment variable and appends `/.odbc.ini` to it and attempts to open a file by that name. If it succeeds, iODBC searches for the data source in this file.
3. If the data source has not been located, and searching the system data sources is permitted, iODBC looks for the environment variable, SYSODBCINI. If it is present, iODBC retrieves its value and attempts to open a file by that name. If it succeeds, iODBC searches for the data source in this file.
4. If the environment variable SYSODBCINI is not present, and searching the system data sources is permitted, iODBC opens the configured system data source file and searches for the data source. The Driver Manager contains the location of the system data source file. If you compile iODBC yourself, this location will be the directory specified to the configure script by the `--with-iodbc-inidir`

parameter. If the `--with-iodbc-inidir` parameter is not specified, the location will be `/etc/odbc.ini`

Driver Search Sequence

iODBC stores information about drivers in a file named `odbcinst.ini`. This file is used to store the location of the Relativity Data Client driver shared library, among other things.

When iODBC searches for this file, it uses the following search sequence:

1. iODBC looks for the environment variable, ODBCINSTINI. If it is present, iODBC retrieves its value and attempts to open a file by that name. If it succeeds, iODBC searches for the driver in this file.
2. If the environment variable ODBCINSTINI is not present, iODBC retrieves the value of the HOME environment variable and appends `/.odbcinst.ini` to it and attempts to open a file by that name. If it succeeds, iODBC searches for the driver in this file.
3. If the driver has not been located, and searching the system `odbcinst.ini` file is permitted, iODBC looks for the environment variable, SYSODBCINSTINI. If it is present, iODBC retrieves its value and attempts to open a file by that name. If it succeeds, iODBC searches for the driver in this file.
4. If the environment variable SYSODBCINSTINI is not present, and if searching the system `odbcinst.ini` file is permitted, iODBC opens the configured system `odbcinst.ini` file and searches for the driver. The Driver Manager contains the location of the system `odbcinst.ini` file. If you compile iODBC yourself, this location will be the directory specified to the configure script by the `--with-iodbc-inidir` parameter. If the `--with-iodbc-inidir` parameter is not specified, the location will be `/etc/odbcinst.ini`.