

IBM® DB2® OLAP Server™ 8.1



SQL Interface Guide



© Copyright International Business Machines Corporation 1998, 2002. All Rights Reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Copyright 1991-2002 Hyperion Solutions Corporation. All rights reserved. This material belongs to Hyperion Solutions Corporation and its licensors. You may only use it internally in the form in which it is received. You have no right to reproduce, reprint, modify, distribute, or transfer this material to any other person, except as expressly and unambiguously stated in a separate agreement between you and Hyperion Solutions Corporation.

Portions copyright 2002 eHelp Corporation. All rights reserved.

Portions copyright 1995-2002 MERANT Solutions, Inc. All rights reserved.

IBM, AIX, DB2, DRDA, DB2 OLAP Server, and DB2 Universal Database are trademarks of International Business Machines Corporation in the United States, other countries, or both. 1-2-3, Notes, and Lotus are registered trademarks of Lotus Development Corporation and/or IBM Corporation.

Hyperion, Essbase, and the Hyperion “H” logo are registered trademarks, and Hyperion Solutions is a trademark of Hyperion Solutions Corporation

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both. (For a complete list of Intel trademarks, see www.intel.com/tradmarx.htm.)

UNIX is a registered trademark of The Open Group in the United States and other countries. Other company, product or service names may be the trademarks or service marks of others.

Printed in the U.S.A.

Contents

Preface	7
Purpose	7
Audience	7
Document Contents	8
Sample Applications	8
Conventions.....	9
More Information	10
Chapter 1: Introducing SQL Interface	15
New Features.....	15
New for Release 6.5	15
New for Release 6.2	16
New for Release 6.1	16
New for Release 6.0	17
Understanding SQL Interface	17
Preparing for Data Loading.....	18
Chapter 2: Configuring a Data Source	19
Configuring a Data Source on Windows	19
Example: Creating a Data Source on Windows NT	20
Example: Creating a Data Source on Windows 2000	23
Configuring a Data Source on UNIX.....	27
Understanding the .odbc.ini File	27
Editing the .odbc.ini File for Changes.....	27
Example: Editing .odbc.ini for Oracle 8.....	27

Chapter 3: Preparing Multiple-Table Data for Loading	29
Understanding Multiple-Table Preparation	29
Understanding Two Methods for Multiple-Table Data Preparation	30
Choosing a Preparation Method.....	31
Understanding Security Requirements.....	32
Preparing Data from Multiple Tables	32
Chapter 4: Loading SQL Data	33
Understanding Data Loading	33
Using Substitution Variables	34
Understanding the Rules for Substitution Variables.....	34
Creating Substitution Variables	34
Creating a Data Load Rules File.....	36
Summary of Steps to Create a Data Load Rules File	36
Step 1: Opening the Data Prep Editor	37
Step 2: Selecting the SQL Data Source	38
Step 2a: Creating SQL Queries (optional)	40
Example: Selecting Fields and Records from a Table	40
Example: Querying Data from Two Tables.....	41
Example: Order By with No Where Clause.....	42
Example: Order By with a Where Clause.....	42
Example: Group By with No Where Clause (Incorrect).....	43
Example: Group By with a Where Clause (Correct)	44
Step 3: Defining the Data Load Rules	45
Changing Field Names.....	46
Changing Field Properties	48
Step 4: Associating the Rules File with an Outline	50
Step 5: Validating the Rules File	51
Step 6: Saving the Rules File	52
Loading the Data.....	53
Chapter 5: Using Non-MERANT ODBC Drivers	57
Configuring a Non-MERANT Driver	58
Creating a Configuration File for Non-MERANT Drivers	58
Including the Minimum Required Contents.....	59

Finding a Driver Name on Windows	62
Finding a Driver Name on UNIX	63
Example: Creating a Configuration File for the MS Access Driver on Windows ...	63
Example: Creating a Configuration File for the IBM DB2 Driver on AIX.....	64
Configuring Non-MERANT Data Sources	65
Configuring an MS Access Data Source	65
Configuring an IBM DB2 Data Source	69
Support for Non-MERANT Drivers.....	70
Index	71

Preface

This preface provides information about Hyperion® and this document:

- “Purpose” on page 7
- “Audience” on page 7
- “Document Contents” on page 8
- “Sample Applications” on page 8
- “Conventions” on page 9
- “More Information” on page 10

Purpose

Use this guide to help you load data from SQL, relational, and flat-file data sources into an Essbase® database. This guide explains the SQL Interface features and options, and contains the concepts, processes, procedures, formats, tasks, and examples that you need to use the software.

Audience

This guide is primarily for database administrators who install SQL Interface, configure Open Database Connectivity (ODBC) drivers, and load relational data into Essbase databases.

To use the information in this book, you need an understanding of relational databases and ODBC drivers, and knowledge of Essbase database outlines.

Document Contents

- “Introducing SQL Interface” on page 15 provides a high-level list of steps for using SQL Interface.
- “Configuring a Data Source” on page 19 provides information and instructions to select and configure the correct database driver to ensure a fast and smooth data load.
- “Preparing Multiple-Table Data for Loading” on page 29 provides instructions to prepare data for loading into an Essbase database.
- “Loading SQL Data” on page 33 provides information and instructions to construct a data load rules file to map the data source to the Essbase database.
- “Using Non-MERANT ODBC Drivers” on page 57 provides information you can use if you need to use ODBC drivers other than those supplied by Essbase.

Sample Applications

This guide provides examples based on an application called Sample, which is provided with Essbase, and a dBASE file called `budget.dbf`, which is provided with the SQL Interface software in the `/app/SQL_Samp` directory of your Essbase installation.

The individual who installs the server is responsible for making Sample available to end users. Sample contains three databases: Basic, Interntl, and Xchgrate. Contact your Essbase system administrator if any of the following problems occur when you connect to OLAP Server:

- You cannot find the Sample application.
- You do not have adequate access to the Sample application.
- You do not see any data in the Sample databases.

You can review a brief introduction to the Sample application in the *Essbase Installation Guide*, and a longer discussion in the *Essbase Database Administrator's Guide*, Volume I.

Conventions

Use this table to understand the conventions used in this guide:

Table i: Conventions Used in This Document

Item	Meaning
	Arrows indicate the beginning of a procedure consisting of sequential steps.
Brackets []	In examples, brackets indicate that the enclosed elements are optional.
Bold	Bold text indicates words or characters that you type exactly as they appear on the page. Bold in procedural steps highlights major interface elements.
CAPITAL LETTERS	Capital letters denote Essbase commands, functions (except MaxL), and configuration settings. (Example: CLEARBLOCK command)
Example text	Courier font indicates that the material shown is a code or syntax example.
Ctrl + 0	Keystroke combinations shown with the plus symbol (+) indicate that you should press the first key and hold it while you press the next key. Do not type the + symbol.
<i>Courier italics</i> <i>ARBORPATH</i> <i>n, x</i>	<ul style="list-style-type: none"> • Courier italic text indicates a variable field in command syntax. Substitute a value in place of the variable shown in Courier italics. • When you see the environment variable <i>ARBORPATH</i> in italics, substitute the value of <i>ARBORPATH</i> from your site. • Italic <i>n</i> stands for a variable number; italic <i>x</i> can stand for a variable number or an alphabet. These variables are sometimes found in formulas.
<i>Italics</i>	Italics may be used to indicate emphasis, or a variable whose value you should supply. Italics also mark the first occurrence of a word that is in the glossary.
Ellipses (...)	Ellipsis points indicate that text has been omitted from an example.

Table i: Conventions Used in This Document (Continued)

Item	Meaning
Mouse orientation	This document provides examples and procedures using a right-handed mouse. If you use a left-handed mouse, adjust the procedures accordingly.
Menu options	Options in menus are shown in the following format: <i>Menu name > Menu command > Extended menu command</i> For example: File > Desktop > Accounts
The suffix .DLL	A dynamic link library file has the suffix .dll. Your operating system might use the term <i>shared object</i> or <i>shared library files</i> instead.

More Information

If you do not find what you need in this book, refer to the *Essbase Information Map* to locate other documents provided with Essbase. The *Essbase Information Map* is in the root directory of your installation CD or in the \docs directory of your Essbase installation. The file is `essdocs.htm`.

If Hyperion Software is not your software provider, contact your software provider's Technical Support hotline.

- You can access the documentation from the DB2 OLAP Server Hyperion site at: www.ibm.com/db2/db2olap/library.html

Use this chapter to understand SQL Interface and prepare for data loading with it:

- If you are already familiar with SQL Interface, review the changes since Release 5 in “New Features” on page 15.
- If you need a conceptual overview of SQL Interface, see “Understanding SQL Interface” on page 17.
- If you need to identify the steps required to load data with SQL Interface, see “Preparing for Data Loading” on page 18.

New Features

Use this section to identify new features or changes since Release 5:

- “New for Release 6.5” on page 15
- “New for Release 6.1” on page 16
- “New for Release 6.0” on page 17

New for Release 6.5

Release 6.5 provides these improvements:

- SQL Interface is installed automatically with every OLAP Server installation.
- You can use substitution variables in SQL strings in a manner similar to substitution variable use in report scripts and calculation scripts. For details, see “Using Substitution Variables” on page 34.
- Release 6.2 supports MERANT DataDirect ODBC drivers Version 4.0 on Windows NT and UNIX, except for AIX, which uses Version 3.6.

New for Release 6.2

Release 6.2 includes these improvements:

- Sample SQL Interface data source files are now located in `/app/SQL_Samp` of your Essbase installation instead of in the same directory as the Sample application, `/app/Sample`.
- Release 6.2 supports MERANT DataDirect ODBC drivers Version 3.6 on Windows NT and UNIX.

If you do not have the ODBC components but you have already installed SQL Interface, you can install the ODBC components by navigating to the `ARBORPATH\odbc\merant40\mdac` directory (Windows) or `$ARBORPATH\setup` (UNIX) in your Essbase installation, and double-clicking `mdac_typ.exe`.

New for Release 6.1

Release 6.1 includes this improvement:

As of Release 6.1, the SQL Interface installation program on Windows no longer installs the ODBC Data Source components in the `/bin` directory. You will need these components to configure your data sources.

- To verify whether you have these components installed, and if not, how to install them, use this procedure:

1. Open your Windows control panel, and perform either of these steps:
 - Check for an ODBC Data Sources icon in Windows NT.
 - Double-click the Administration Services icon in the Control Panel, double-click the Data Sources icon and check for ODBC drivers.
2. If you do not have the ODBC components, you can download Microsoft Data Access Components (MDAC) from the following Web site:

http://www.microsoft.com/data/MDAC21info/manifest_intro.htm

New for Release 6.0

Release 6.0 includes these improvements:

- SQL Interface runs on HP-UX and Windows 2000, as well as on AIX, Solaris, and Windows NT.
- As of Release 6, Essbase SQL Drill-Through is no longer provided, so you may need to use a previous release of SQL Drill-Through.

To use a previous release of SQL Drill-Through with Release 6.x, do the following:

- a. Install Essbase Spreadsheet Add-in, Release 6.x.
- b. Install Essbase SQL Drill-Through, Release 5.0.2 Patch 9 (or later), in the same directory.

For information on the data sources qualified for use with Essbase on Windows NT and UNIX platforms, see the *Essbase Installation Guide*, file `essinst.pdf` in the `/docs/pdf` directory of your Essbase installation.

Note: You can also load data from spreadsheets and text files directly, using Application Manager. See the *Essbase Database Administrator's Guide*.

Understanding SQL Interface

Use SQL Interface to load data directly from a number of popular SQL, relational, and flat-file database servers. SQL Interface provides the connection you need to accomplish these tasks:

- You can execute SQL statements on data before loading. For example, this allows you to load summary data instead of the entire contents of your source database.
- Optionally, you can build Essbase dimensions dynamically as you load data.
- You can load data from flat files that are in ASCII text format, as well as connecting directly to your relational source database.

Note: You can load data from spreadsheets or text files using Application Manager without the use of SQL Interface. See the *Essbase Database Administrator's Guide*.

SQL Interface works with Application Manager to retrieve data:

1. You write a SELECT statement in SQL using Application Manager.
2. SQL Interface passes it to the SQL or relational database server. For non-SQL data sources, SQL Interface converts SQL statements to requests for data that the source understands.
3. SQL Interface interprets the records received from the SQL, relational, or flat-file database server using the rules defined in the data load rules file. (For more information on data load rules files, see Chapter 4, “Loading SQL Data.”)
4. SQL Interface loads the interpreted, summary-level data into the database.

Preparing for Data Loading

- To prepare to load SQL, relational, or flat-file data use this procedure:
1. SQL Interface is installed during your OLAP Server installation. See the *Essbase Installation Guide* for information about initial configuration tasks.
 2. Choose your ODBC driver and point it to its data source. See Chapter 2, “Configuring a Data Source.”
 3. If your data is in multiple tables, prepare the data for loading by doing either of these actions:
 - Create one table or view in your SQL database before using SQL Interface.
 - Join the tables during the data load by entering a SELECT statement in Application Manager.See “Understanding Multiple-Table Preparation” on page 29 for instructions.
 4. Check that you can connect to the data source by using the Data Prep Editor in Application Manager to open the SQL source file. See Chapter 4, “Loading SQL Data.”
 5. Create a data load rules file to tell SQL Interface how to interpret and load the SQL data into your Hyperion Essbase database. See Chapter 4, “Loading SQL Data.”

After these steps are complete, you can load your data. See Chapter 4, “Loading SQL Data” for instructions.

Configuring a Data Source

Before you can access data using SQL Interface, you must select a data source and the driver it requires. The procedure depends on your operating system:

- “Configuring a Data Source on Windows” on page 19
- “Example: Creating a Data Source on Windows NT” on page 20
- “Configuring a Data Source on UNIX” on page 27

Note: For detailed, driver-specific information on each of the MERANT drivers, see the *MERANT DataDirect Connect ODBC Reference* in the `/odbc/merant40/docs` directory of your Essbase installation (`$ARBORPATH/odbcdocs` for UNIX).

Configuring a Data Source on Windows

On Windows, use the ODBC Administrator to configure the data source.

➤ To configure the ODBC Administrator, use this procedure :

1. Start ODBC Administrator, using either of these methods:
 - Open the Windows Control Panel.
 - For NT, double-click the **ODBC Data Sources** icon, if one exists. If it does not, use the other method in this list.
 - For Windows 2000, double-click the **Administrative Services** icon, then double-click the **Data Sources** icon.
 - At the Windows NT command prompt, type the appropriate path to the file `ODBCAD32.EXE`, which you typically can find in the Windows system directory. For example:

```
C : \WINNT\SYSTEM32\ODBCAD32
```

If you do not have the ODBC components, look for the installation program `mdac_typ.exe` in `\odbc\merant40\mdac` in your Essbase installation.

2. Choose a data source, or add one, and then enter the required information about the driver.

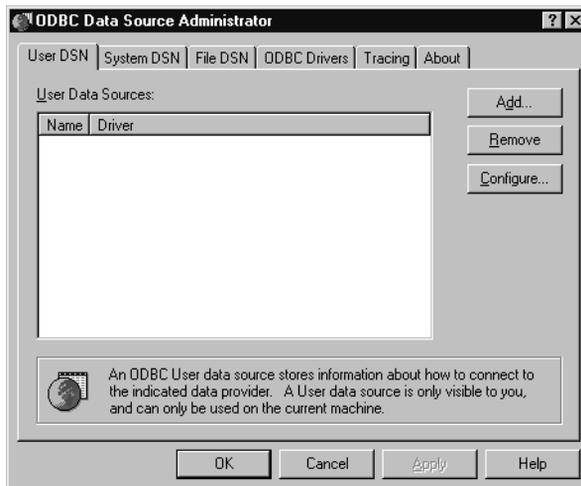
For detailed instructions, see the appropriate driver chapter in the *MERANT DataDirect Connect ODBC Reference* (for example, “Connect ODBC for dBASE,” or “Connect ODBC for Oracle”).

For an examples, see the example appropriate for your operating system:

- “Example: Creating a Data Source on Windows NT” on page 20
- “Example: Creating a Data Source on Windows 2000” on page 23

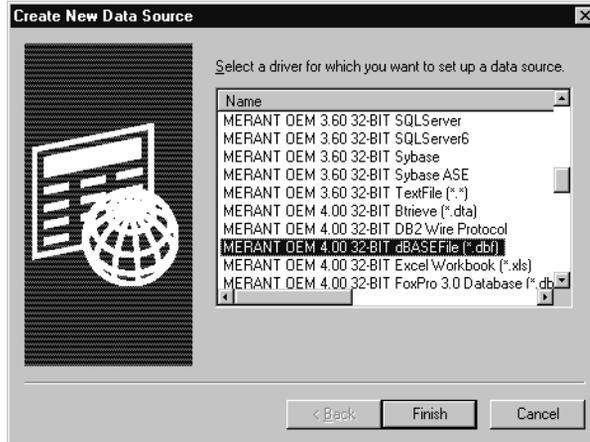
Example: Creating a Data Source on Windows NT

- To configure the dBASE data source, complete the following steps:
1. Start the ODBC Administrator, using the instructions in Step 1 in “Configuring a Data Source on Windows” on page 19.
 2. Click the User DSN tab in the **ODBC Data Source Administrator** dialog box.

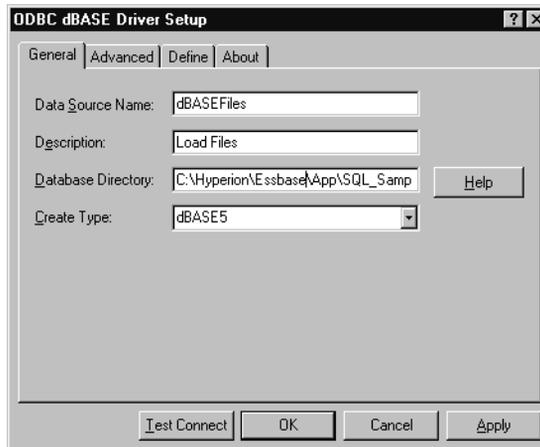


3. Click **Add** on the **User DSN** page.

The **Create New Data Source** dialog box is displayed.



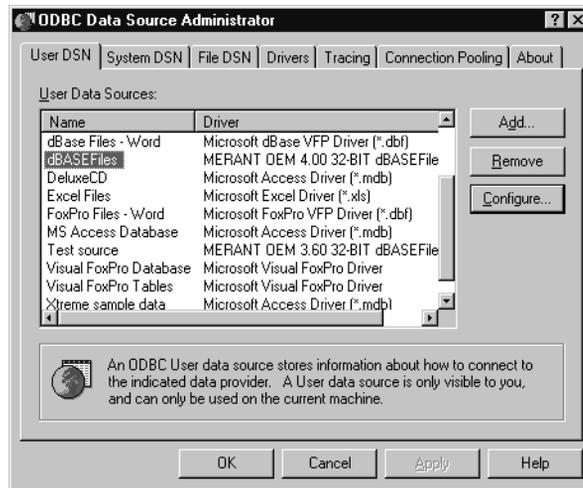
4. From the list box, select **MERANT OEM 4.0 32-bit dBASEFile (*.dbf)** and click the **Finish** button to display the **General** tab of the **ODBC dBASE Driver Setup** dialog box.



5. Type the following information on the **General** tab:
 - a. In the **Data Source Name** text box, type **dBASEFiles**.
 - b. In the **Description** text box, type **Load Files**.

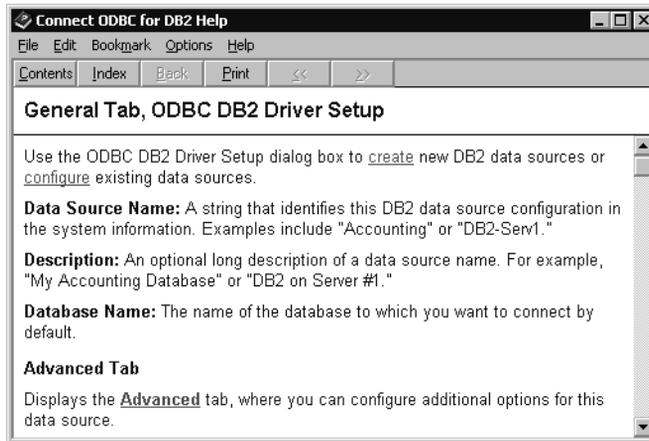
- c. In the **Database Directory** text box, type `c:\hyperion\essbase\app\SQL_Samp`. If you installed Essbase in the default directory, `c:\hyperion\essbase`, `budget.dbf` is stored in this subdirectory. If you installed it elsewhere, substitute your site's value for `c:\hyperion\essbase`.
- d. Click **OK**.

The **User DSN** tab of the **ODBC Data Source Administrator** dialog box is displayed with `dBASEFiles` listed in the **User Data Sources** list box:



- 6. Click **OK** to close the ODBC Data Source Administrator.

If you need help during the procedure, use the online help. For example, clicking the Help button on the **General** tab of the **ODBC dBASE Driver Setup** dialog box (in Step 4) brings up a Help screen:

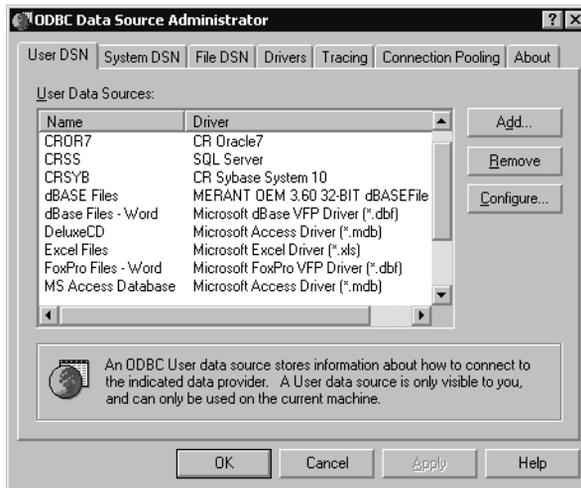


Different help topics are available from different dialog box Help buttons.

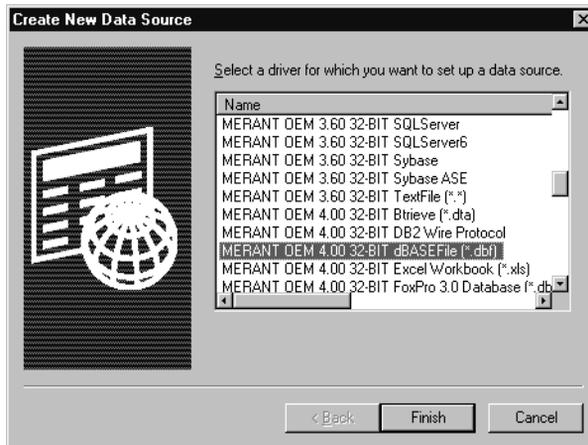
Example: Creating a Data Source on Windows 2000

- To configure the dBASE data source, complete the following steps:
 1. Start the ODBC Administrator, using the instructions in Step 1 in “Configuring a Data Source on Windows” on page 19.

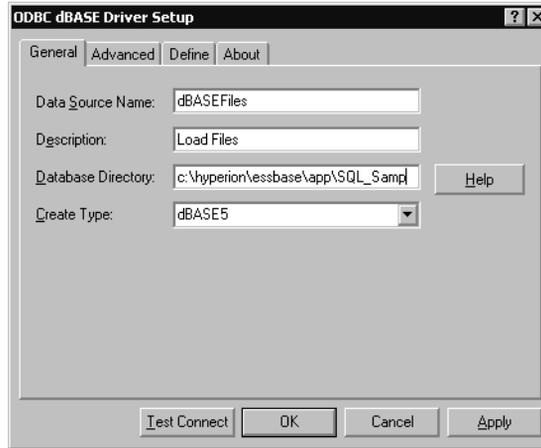
- Click the **User DSN** tab in the **ODBC Data Source Administrator** dialog box:



- Click **Add** on the **User DSN** page.
The **Create New Data Source** dialog box is displayed.



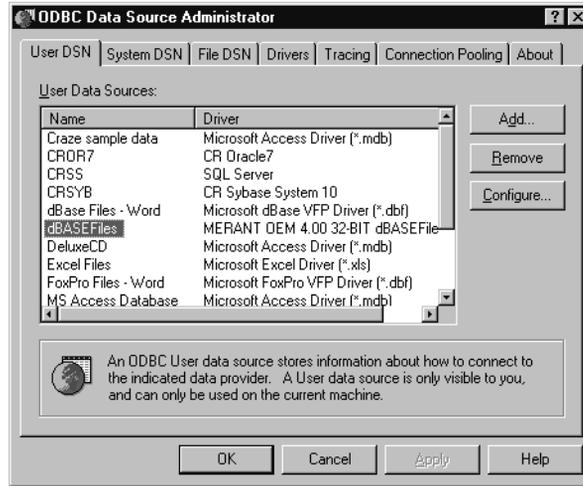
4. From the list box, select **MERANT OEM 4.0 32-BIT dBASEFile (*.dbf)** and click the **Finish** button to display the **General** tab of the **ODBC dBASE Driver Setup** dialog box:



5. Type the following information on the **General** tab:
 - a. In the **Data Source Name** text box, type `dBASEFiles`.
 - b. In the **Description** text box, type `Load Files`.
 - c. In the **Database Directory** text box, type `c:\hyperion\essbase\app\SQL_Samp`. If you installed Essbase in the default directory, `c:\hyperion\essbase`, `budget.dbf` is stored in this subdirectory. If you installed it elsewhere, substitute your site's value for `c:\hyperion\essbase`.

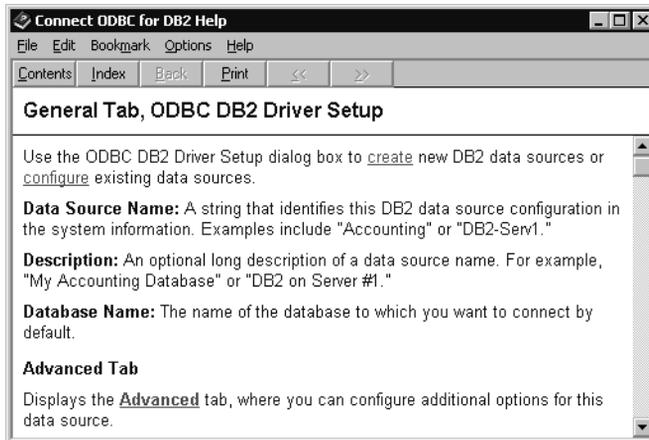
- d. Click **OK**.

The **User DSN** tab of the **ODBC Data Source Administrator** dialog box is displayed with **dBASEFiles** listed in the **User Data Sources** list box:



- 6. Click **OK** to close the ODBC Data Source Administrator.

If you need help during the procedure, use the online help. For example, clicking the **Help** button on the **General** tab of the **ODBC dBASE Driver Setup** dialog box (in Step 4) brings up a Help screen:



Different help topics are available from different dialog box **Help** buttons.

Configuring a Data Source on UNIX

The ODBC Administrator is not available on UNIX platforms. SQL Interface generates the `.odbc.ini` file upon installation, and you do not need to do any further configuration to use the `budget.dbf` file.

If you add data sources or drivers, you may need to edit the `.odbc.ini` file:

- “Understanding the `.odbc.ini` File” on page 2-27
- “Editing the `.odbc.ini` File for Changes” on page 2-27
- “Example: Editing `.odbc.ini` for Oracle 8” on page 2-27

Understanding the `.odbc.ini` File

The `.odbc.ini` file contains the ODBC connection and configuration information, such as data source name and driver product name. Specific information varies among platforms. The file and format are defined by the Microsoft ODBC specification.

Editing the `.odbc.ini` File for Changes

Because the `.odbc.ini` file maps data source names to driver products, you must edit the `.odbc.ini` file if you add a new driver product or data source or change driver products or data sources.

Example: Editing `.odbc.ini` for Oracle 8

Here is the entry you make in the `.odbc.ini` file for the Oracle 8 driver (assuming you installed to `home/hyperion/essbase`):

```
[ODBC Data Sources]
Oracle8=MERANT 4.0 Oracle 8 Driver

[Oracle8]
Driver=/home/hyperion/essbase/dlls/ARor815.so
Description=Oracle8
ServerName=name
```

Substitute for *name* the Oracle server name appropriate for your site, and substitute 3.6 for 4.0 if you are on AIX.

For more information about the `.odbc.ini` file, see the *MERANT DataDirect Connect ODBC Reference* in the `$ARBORPATH/odbcdocs` directory.

Preparing Multiple-Table Data for Loading

This section tells you how to prepare relational data in multiple tables for loading into Essbase:

- “Understanding Multiple-Table Preparation” on page 29
- “Preparing Data from Multiple Tables” on page 32

If all your data is in one table, skip to Chapter 4, “Loading SQL Data.”

Understanding Multiple-Table Preparation

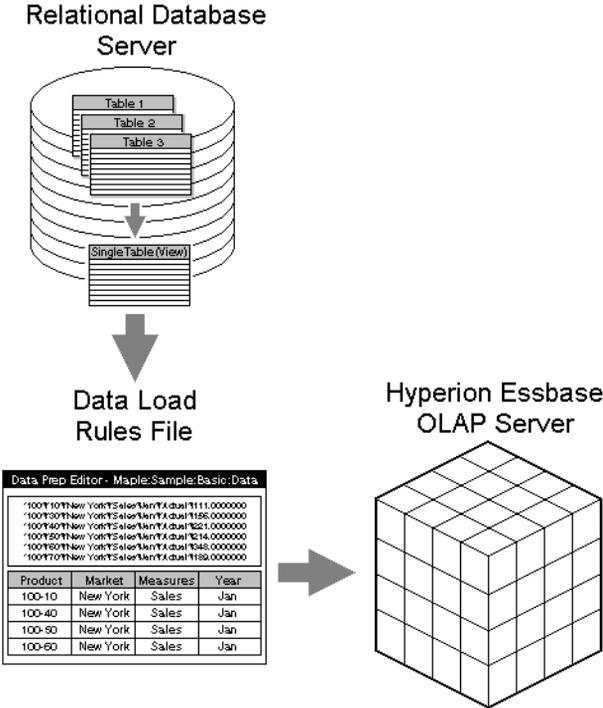
Use this section to understand the two methods available for preparing multi-table data:

- “Understanding Two Methods for Multiple-Table Data Preparation” on page 30
- “Understanding Security Requirements” on page 32
- “Choosing a Preparation Method” on page 31

Understanding Two Methods for Multiple-Table Data Preparation

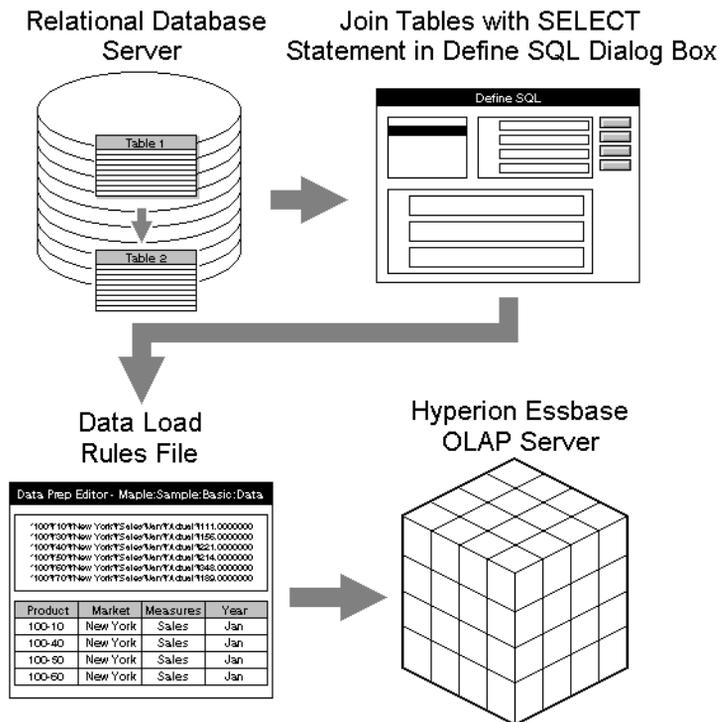
If data is in multiple tables, create a single table or a view of the data in either of the following ways:

- Create one table or view in the SQL database before using SQL Interface:



See “Choosing a Preparation Method” on page 31 for instructions.

- Join the tables by entering a SELECT statement in Application Manager during the data load:



See “Preparing Data from Multiple Tables” on page 32 for instructions.

The SQL data source file must always be accessible to OLAP Server. Make sure that the SQL data source file resides on the computer where OLAP Server is installed, or on the network.

Choosing a Preparation Method

A SQL database server reads from a single table and maintains a single view more efficiently than it processes a multiple-table SELECT statement. Therefore, creating a single table or view before you use SQL Interface greatly reduces the amount of processing time the SQL server needs.

Understanding Security Requirements

Data load preparation requires specific permissions:

- For any data load preparation method, you need **SELECT** access privileges to all tables containing the relevant data.
- For creating a single table or view in the relational database source, you need **CREATE** access privileges in the SQL database to create a single table or view. Ask the SQL Database Administrator to provide you with the appropriate **CREATE** privileges.

If you cannot obtain **CREATE** privileges, use Application Manager to join tables during the data load. See “Preparing Data from Multiple Tables” on page 32 for instructions.

Preparing Data from Multiple Tables

- Use this process to prepare data from multiple tables:
 1. Obtain the required privileges:
 - **SELECT** access to all tables where relevant data is stored
 - **CREATE** access to create a single table or view to join multiple-table data before using SQL Interface
 2. Use one of these methods to prepare data from multiple tables:
 - Create a single physical table or virtual table (view) based on data from multiple tables in the source database.
 - Create a **SELECT** statement in Application Manager that joins the tables:
 - i. Identify the tables and columns (or fields) containing the data that you want to load into Essbase.
 - ii. Select **File > Open SQL** in the **Application Manager Data Prep Editor** to display the **Define SQL** dialog box. For more information, see “Step 1: Opening the Data Prep Editor” on page 37.
 - iii. Write a **SELECT** statement that joins the tables. For more information, see “Step 2: Selecting the SQL Data Source” on page 38 and “Step 2a: Creating SQL Queries (optional)” on page 40.

Note: Essbase passes the **SELECT** statement to SQL Server without verifying the syntax.

This chapter describes how to load SQL data into OLAP Server using Application Manager:

- “Understanding Data Loading” on page 33
- “Using Substitution Variables” on page 34
- “Creating a Data Load Rules File” on page 36
- “Loading the Data” on page 53

Examples in this section are based on samples provided with your installation:

- The Sample application.
- The Basic database.
- The dBASE data file `budget.dbf`, which is installed with SQL Interface in the `/app/SQL_Samp` directory of your Essbase installation. This file contains budget data for the West and the East regions of a fictitious company.

For more information about the sample applications, see “Sample Applications” on page 8.

Understanding Data Loading

After you have configured the data source and prepared the multiple-table data for loading, load the data using this process:

1. If you plan to use substitution variables, create them. See “Using Substitution Variables” on page 34 for instructions.
2. Create a rules file. See “Creating a Data Load Rules File” on page 36.
3. Load data into the Essbase database. See “Loading the Data” on page 53.

Using Substitution Variables

Similar to report scripts and calculation scripts, you can use substitution variables in SQL strings. You can use substitution variables that apply across the entire OLAP Server, or specify them for a particular application or database.

Understand the rules for substitution variable use in SQL Interface before you create and use them:

- “Understanding the Rules for Substitution Variables” on page 34
- “Creating Substitution Variables” on page 34

Understanding the Rules for Substitution Variables

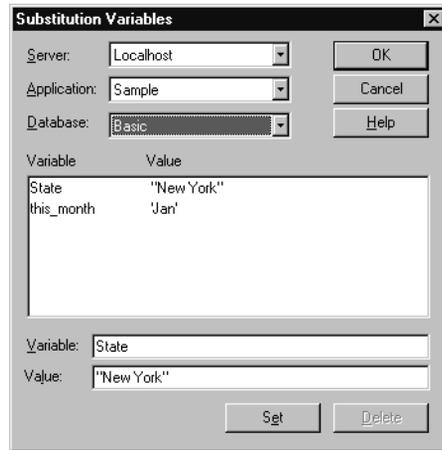
Remember these rules when you create substitution variables:

- The value of the substitution variable should be valid and appropriate SQL. Essbase does not validate the value.
- Be especially careful when you use quotation marks or single quotes. Different source databases have different conventions about when to use them. The examples in this document are based on dBASE because the sample data for SQL Interface, `budget.dbf`, is dBASE.
- Whether or not you use a substitution variable, be sure that no regular SQL operators in SELECT, FROM, or WHERE clauses start with the ampersand (&), because that is the Essbase identifier for substitution variables.

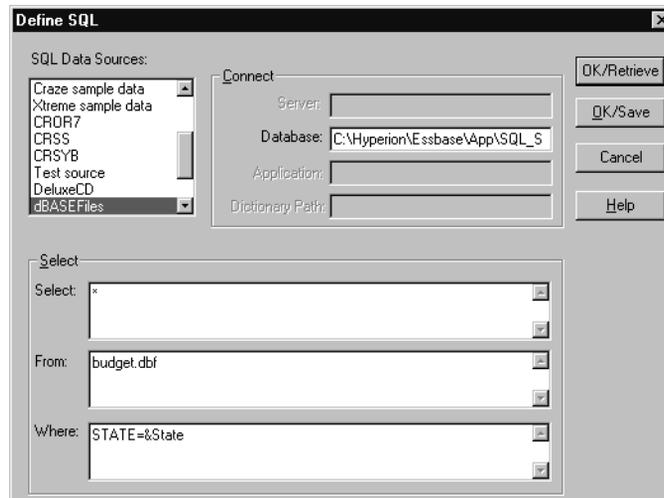
Creating Substitution Variables

- To create and use a substitution variable, use this procedure:
 1. Create the substitution variable using the instructions in the *Essbase Database Administrator's Guide* section *Creating Applications and Databases*.
 2. Open the rules file you plan to use, then select File > Open SQL from the menu. The Define SQL dialog box appears.
 3. In the Define SQL dialog box, use the substitution variable instead of a “field=value” string in the Select, From or Where box. Remember to type the ampersand (&) in front of the substitution variable name.
 4. Click OK/Retrieve to retrieve the data using the rules file.

For example, if you want to create a substitution variable to change the value of MONTH or STATE without having to create a new data rules file for each change, you could create substitution variables `this_month` and `State`:



Then, in the Define SQL dialog box, you can specify the substitution variable:



When you click OK/Retrieve, the data load rules file is created, and you can save it. When you need to change the value for state or month, simply change the value of the substitution variables and open the same data load rules file.

Creating a Data Load Rules File

To load SQL data into the Essbase database, you must create and use a data load rules file.

Data load rules are a set of operations that Essbase performs on data when it loads it into an Essbase database. The operations are stored in a rules file that tells Essbase how to load the SQL data and map it to your database outline. The rules file can also reject invalid records in the data source.

You can reuse a rules file with any data source that needs the same set of data load rules.

Summary of Steps to Create a Data Load Rules File

To create a data load rules file, follow these steps:

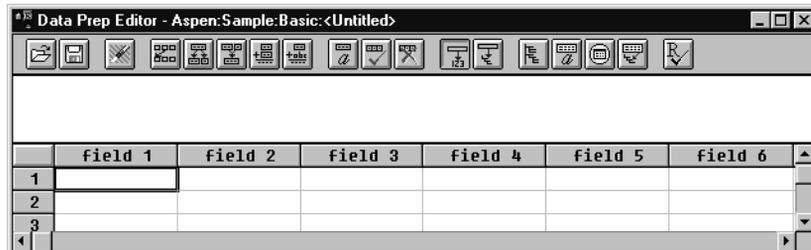
1. Open the Data Prep Editor. See “Step 1: Opening the Data Prep Editor” on page 37 for instructions.
2. Select the SQL data source, `budget.dbf` in the example. See “Step 2: Selecting the SQL Data Source” on page 38 for instructions. If you plan to create SQL queries in Essbase, see “Step 2a: Creating SQL Queries (optional)” on page 40.
3. Define the data load rules in a rules file. See “Step 3: Defining the Data Load Rules” on page 45 for instructions.
4. Associate the data load rules file with the database outline, `Sample Basic` in the example. See “Step 4: Associating the Rules File with an Outline” on page 50 for instructions.
5. Validate the data load rules file. See “Step 5: Validating the Rules File” on page 51 for instructions.
6. Save the data load rules file. See “Step 6: Saving the Rules File” on page 52 for instructions.

Step 1: Opening the Data Prep Editor

- To open the Data Prep Editor:
 1. Start Application Manager and connect to your server. The Application Manager Desktop window is displayed:



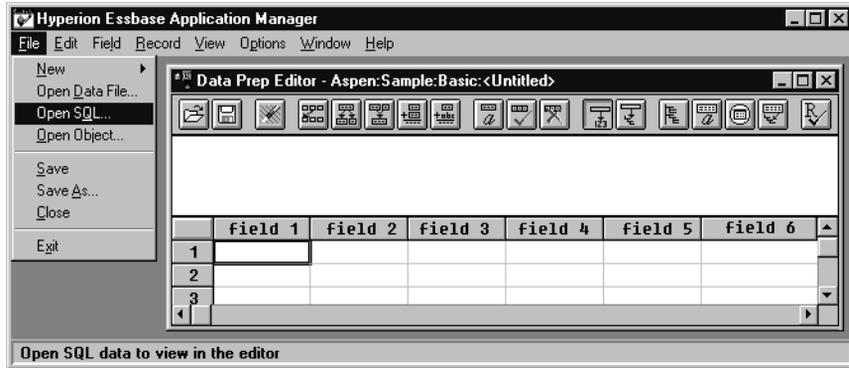
2. From the **Applications** list box, select **Sample**.
3. From the **Databases** list box, select **Basic**.
4. Click the **Data Load Rules** button, . The existing data load rules are listed in the Desktop Window, and the New button becomes active.
5. Click **New** to display the **Data Prep Editor**:



Step 2: Selecting the SQL Data Source

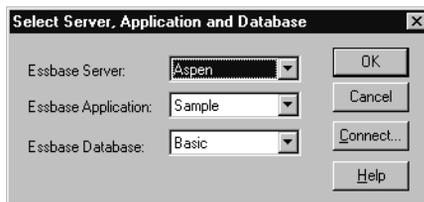
➤ To select the data source, `budget.dbf` in the sample screens, use this procedure:

1. Choose **File > Open SQL**:

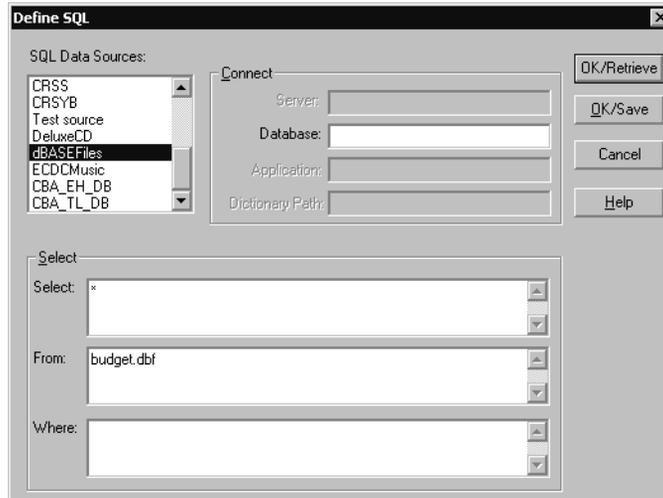


Note: This is the point where SQL Interface is first invoked.

The **Select Server, Application and Database** dialog box is displayed. The **Sample Basic** database is selected:



- Click **OK** to display the **Define SQL** dialog box:



In the **SQL Data Sources** list box, **dBASE Files** is selected. This is the SQL source that you configured in Chapter 2, “Configuring a Data Source.”

The wildcard character (*) appears in the **Select** text box. This loads all the fields from the dBASE file.

- Type the name of the data source file in the **From** text box (`budget.dbf` in this example). You defined the path for this file in Chapter 2, “Configuring a Data Source,” so you do not need to define it here.

Leave the **Where** text box empty to load all the rows from the table.

- Click **OK/Retrieve**.

Essbase opens the data source file, `budget.dbf`, in the **Data Prep Editor**. See “Step 3: Defining the Data Load Rules” on page 45 for more information.

Note these facts about data source files:

- The data source file must be stored on OLAP Server, not on the client machine.
- On UNIX platforms, if you have defined the path for the SQL data source file in the `.odbc.ini` file, you can either enter that path or a different path in the Database text box of the **Define SQL** dialog box.

- On Windows, if you have not defined the path for the SQL source file in the ODBC Administrator, you can type the path in the **Database** text box of the **Define SQL** dialog box.
- If you do not define a path, Essbase looks for the data source file in the directory from which you are running OLAP Server.

Step 2a: Creating SQL Queries (optional)

If you have not created a table or view to retrieve only the information you need from your data source, you can write a SELECT statement to achieve this goal.

Note: Creating a SELECT statement in Application Manager is usually slower than creating a table or view in the source database.

The Define SQL dialog box has Select, From, and Where text boxes to help you write SQL queries. You can specify multiple data sources, filter the display of records, and specify the desired order and grouping of records that will appear in the Data Prep Editor.

The rest of this section contains examples of queries used for data preparation. You can review these examples for help with your site or skip to the next step:

- “Example: Selecting Fields and Records from a Table” on page 40
- “Example: Querying Data from Two Tables” on page 41
- “Example: Order By with No Where Clause” on page 42
- “Example: Order By with a Where Clause” on page 42
- “Example: Group By with No Where Clause (Incorrect)” on page 43
- “Example: Group By with a Where Clause (Correct)” on page 44

Example: Selecting Fields and Records from a Table

The example in “Step 2: Selecting the SQL Data Source” on page 38 uses the wildcard character (*) in the Select text box to load data from all the fields (columns) in the dBASE file. Alternatively, you can type the names of all the fields:

```
VERSION, PRODNO, STATE, ACCOUNT, MONTH, MEASURE
```

Or, you can type only the names of the fields that you want to load, for example:

```
PRODNO , STATE , MONTH
```

The example in “Step 2: Selecting the SQL Data Source” on page 38 also uses the name of a single dBASE file (BUDGET.DBF) in the From text box, and leaves the Where text box empty.

You can also use the Where text box to choose a subset of records (rows) to load from the SQL source files.

- To load only records (rows) for January from `budget.dbf` into the Sample Basic database,
 1. Type the following text in the **From** text box:


```
budget.dbf
```
 2. Type the following text in the **Where** text box:


```
MONTH = "Jan"
```
 3. Click **OK/Retrieve**.

Example: Querying Data from Two Tables

If you are loading data from two relational tables, you can join the tables by specifying both of the table names, separated by a comma (,) in the From text box, and specifying the common (key) fields in the Where text box, separated by an equal sign (=).

- To load data from two tables, `table1.dbf` and `table2.dbf`, where the `Key_No` field of `table1.dbf` contains the same data as the `Key_Code` field of `table2.dbf`:
 1. Type the following text in the **From** text box:


```
TABLE1.DBF , TABLE2.DBF
```
 2. Type the following text in the **Where** text box:


```
TABLE1.Key_No = TABLE2.Key_Code
```
 3. Click **OK/Retrieve**.

Example: Order By with No Where Clause

This SQL statement will query all columns from the relational source, and will sort the results in month order:

The screenshot shows a SQL query builder window with three sections: 'Select', 'From', and 'Where'. The 'Select' section contains an asterisk (*). The 'From' section contains the text 'budget.dbf order by month'. The 'Where' section is empty. Each section has a small dropdown arrow on the right side.

Example: Order By with a Where Clause

This SQL statement will return rows from the relational source where the State column is equal to New York, California, or Nevada. It will sort the results in month order:

The screenshot shows a SQL query builder window with three sections: 'Select', 'From', and 'Where'. The 'Select' section contains an asterisk (*). The 'From' section contains the text 'budget'. The 'Where' section contains the text 'State in ("New York", "California", "Nevada") order by Month'. Each section has a small dropdown arrow on the right side.

Example: Group By with No Where Clause (Incorrect)

This SQL statement will return one row per Prodno, State, Account, and Month. It will summarize the metric contained in the Measure column, producing incorrect results:

The screenshot shows a SQL query editor with the following text:

```
Select: Prodno, State, Account, Month, sum(Measure)
From: budget.dbf group by Prodno, State, Account, Month
Where:
```

Now assume that `budget.dbf` looks like this table:

Table 1: Abbreviated Data Source Example

Version	Prodno	State	Account	Month	Measure
Budget	100-10	New York	Sales	Jan	500
Budget	100-10	New York	Sales	Jan	200
Actual	100-10	New York	Sales	Jan	100

The SQL statement returns these results:

```
100-10, New York, Sales, Jan, 800
```

Budget and Actual data have been added together, which is not the intended result. See the next example for an approach that produces correct results.

Example: Group By with a Where Clause (Correct)

This SQL statement will select only records that have a Version of Budget and a State equal to New York, California, or Nevada:

The screenshot shows a SQL query editor window with the following content:

```

Select:
  Prodno, State, Account, Month, sum(Measure)
From:
  budget.dbf
Where:
  Version = "Budget" and State in ("New York","California","Nevada")
  group by Prodno, State, Account, Month
  
```

This SQL Statement will return one row per Prodno, State, Account, and Month. It will summarize the metric contained in the measure column.

Assume that `budget.dbf` looks like Table 1. The SQL statement returns these results:

```
100-10, New York, Sales, Jan, 700
```

See the *MERANT DataDirect Connect ODBC Reference* for more information about SQL statements for your specific driver.

Step 3: Defining the Data Load Rules

When you click OK/Retrieve in the Define SQL dialog box, Essbase displays the dBASE fields in the Data Prep Editor. Notice that the field names from the data source differ from the Sample Basic dimension names:

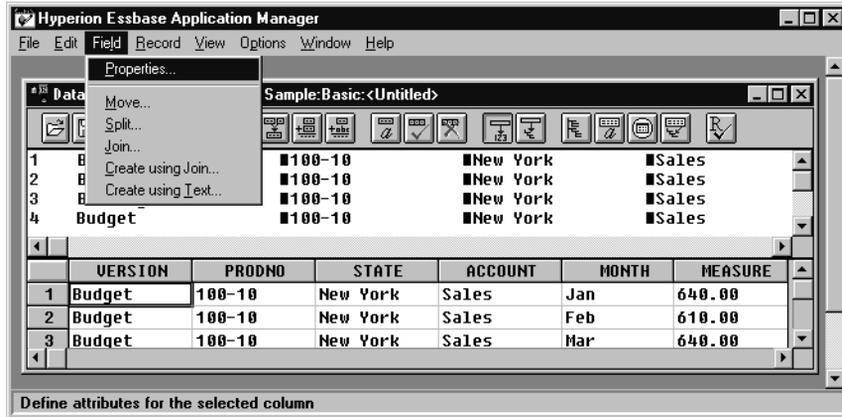
	VERSION	PRODNO	STATE	ACCOUNT	MONTH	MEASURE
1	Budget	100-10	New York	Sales	Jan	640.00
2	Budget	100-10	New York	Sales	Feb	610.00
3	Budget	100-10	New York	Sales	Mar	640.00

Edit the field names and properties to match:

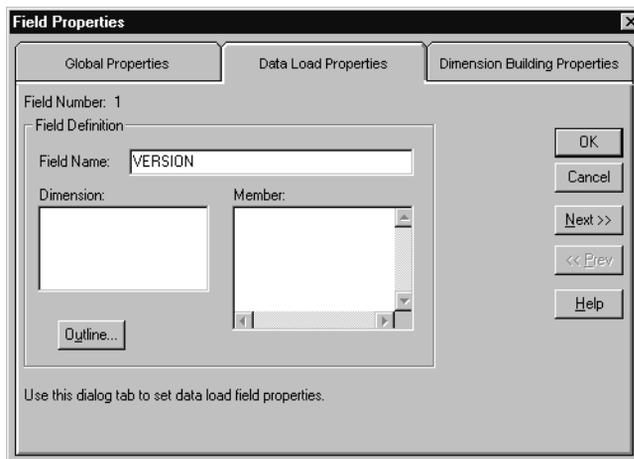
- “Changing Field Names” on page 46
- “Changing Field Properties” on page 48

Changing Field Names

- To change the field names to match the dimension names in the Essbase outline (Sample Basic in the sample screens), use this procedure:
 1. After completing “Step 3: Defining the Data Load Rules” on page 45, select the first cell in the first field (VERSION) and choose **Field > Properties** from the Application Manager menu:



The **Field Properties** dialog box is displayed:



2. Select the **Data Load Properties** tab.

The data load properties for **Field Number 1** are displayed.

3. In the **Field Name** text box, highlight `VERSION` and type the dimension name `Scenario` over it, and click **Next**.

The data load properties for **Field Number 2** are displayed.

4. Type `Product` over the existing name (`PRODNO`) and click **Next**.

The data load properties for **Field Number 3** are displayed.

5. Type `Market` over the existing name (`STATE`) and click **Next**.

The data load properties for **Field Number 4** are displayed.

6. Type `Measures` over the existing name (`ACCOUNT`) and click **Next**.

The data load properties for **Field Number 5** are displayed.

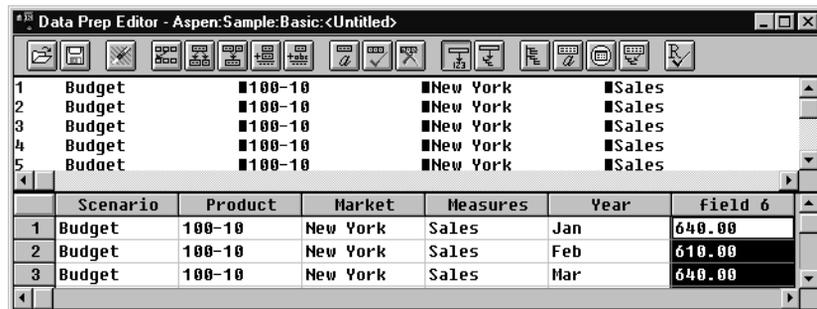
7. Type `Year` over the existing name (`MONTH`) and click **Next**.

The data load properties for **Field Number 6** are displayed. This is the last field in `budget.dbf`. It contains data values. Because you have fully specified each dimension, you do not match this field to a dimension name.

8. Delete the existing name (`MEASURE`) in the **Field Name** text box and leave the box blank.

9. Click **OK** to save all your changes.

The **Data Prep Editor** is displayed again with the fields renamed. Notice that the last column has a temporary name of field 6:

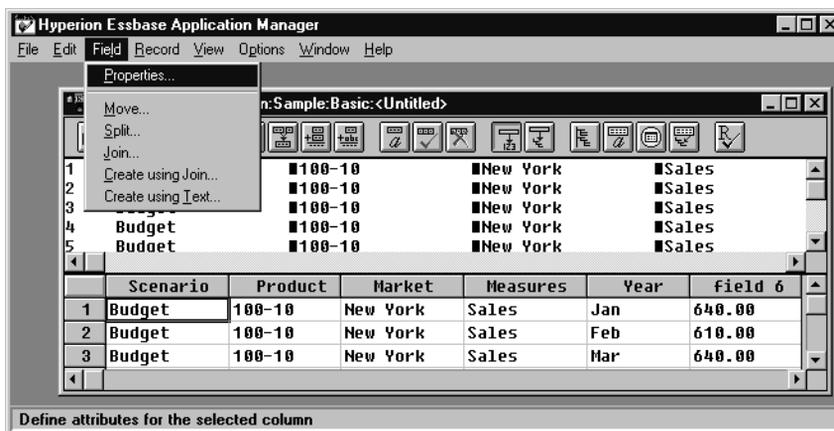


	Scenario	Product	Market	Measures	Year	Field 6
1	Budget	100-10	New York	Sales	Jan	640.00
2	Budget	100-10	New York	Sales	Feb	610.00
3	Budget	100-10	New York	Sales	Mar	640.00

Changing Field Properties

The fields in the data source file may contain unnecessary leading and trailing white spaces.

- To ensure that Essbase deletes these white spaces in each of the fields when it loads the data, use this procedure:
 1. After completing “Step 3: Defining the Data Load Rules” on page 4-45, select the first cell in the first field (**Scenario**) and choose **Field > Properties** from the Application Manager menu:



- Click the **Global Properties** tab:

The screenshot shows the 'Field Properties' dialog box with the 'Global Properties' tab selected. The 'Field Number' is 1. The 'Replace' section has empty text boxes for 'Replace:' and 'With:', and checkboxes for 'Case Sensitive (C)', 'Match Whole Word (W)', and 'Replace All Occurrences (A)'. The 'Case' section has radio buttons for 'Original', 'Upper Case', and 'Lower Case'. The 'Data Field' section has checkboxes for 'Data Field', 'Ignore field during dataload', 'Ignore field during dimension build', 'Drop leading/trailing whitespace' (checked), 'Convert spaces to underscores', and 'Scale'. Navigation buttons 'Next >>' and '<< Prev' are visible.

- Select the **Drop leading/trailing whitespace** check box if it is not already checked.
- Click **Next**, and select **Drop leading/trailing whitespace** for all fields.
- When you reach **Field 6**, select the **Data Field** check box as well as **Drop leading/trailing whitespace**, to tell Essbase that this field contains data values:

This close-up shows the 'Data Field' section of the dialog box. The 'Data Field' checkbox is checked. Other options include 'Ignore field during dataload', 'Ignore field during dimension build', 'Drop leading/trailing whitespace' (checked), 'Convert spaces to underscores', and 'Scale' with an empty text box.

- Click **OK** to close the **Field Properties** dialog box.

You have defined the data load rules for the dBASE file. For more information on creating data load rules files, see the *Essbase Database Administrator's Guide*.

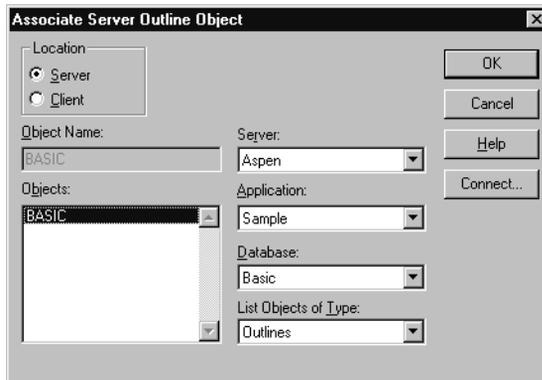
Step 4: Associating the Rules File with an Outline

► To associate the rules file with the Sample Basic outline, use this procedure:

1. Choose **Options > Associate Outline**:



The **Associate Server Outline Object** dialog box is displayed:



2. From the **Objects** list box, select **BASIC** and click **OK**.

Step 5: Validating the Rules File

- To validate the rules file against the Sample Basic outline:
 1. Choose **Options > Validate**:



If Essbase can validate the rules file, it displays the **Verify** message box:



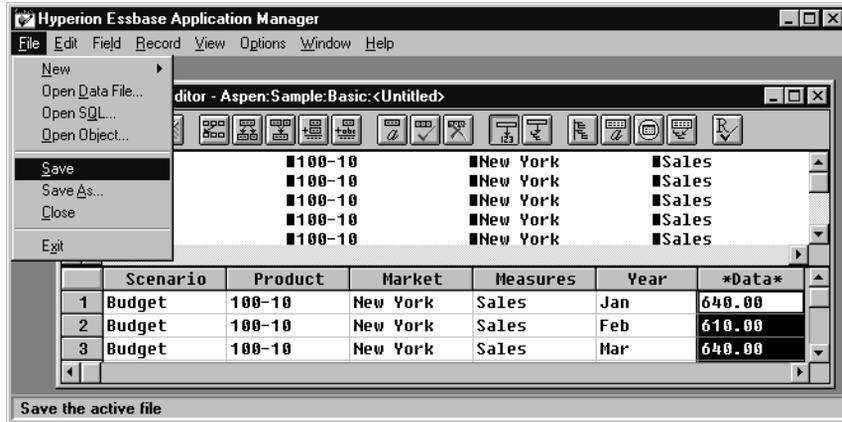
If there are validation problems, Essbase displays the problems in a **Validate Rules** dialog box. If a field has validation problems, ensure that the field name is correct. For more information on validating rules files, see the *Essbase Database Administrator's Guide*.

2. Click **OK** to close this message box.

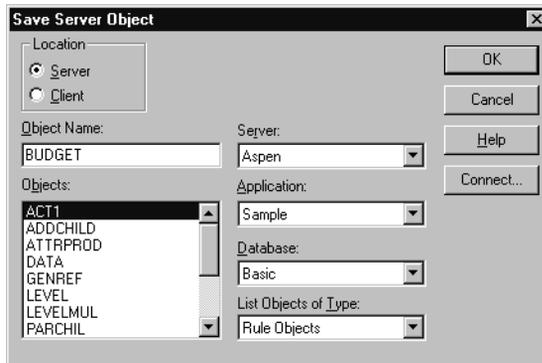
Step 6: Saving the Rules File

► Use this procedure to save the rules file:

1. Choose **File > Save**:



The **Save Server Object** dialog box is displayed:



2. In the **Object Name** text box, type BUDGET.

This gives the rules file the name BUDGET.RUL.

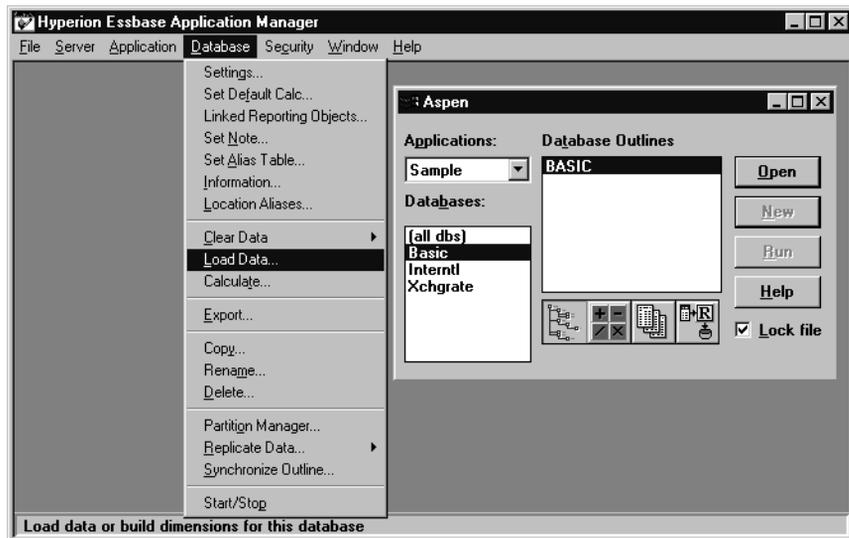
3. Click **OK**.

4. Choose **File > Close** to close the **Data Prep Editor**.

You can now use the BUDGET rules file to load the data from the dBASE file into the Sample Basic database.

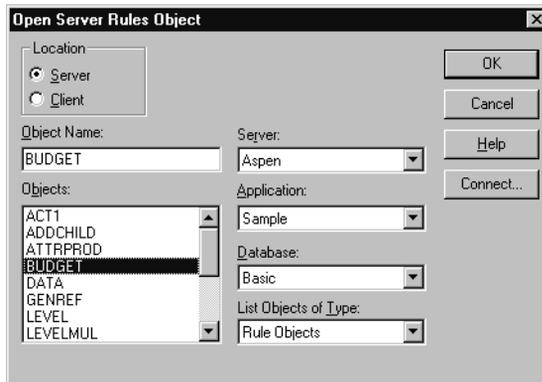
Loading the Data

- To load the data into Sample Basic:
 1. In the Application Desktop Window, select the Sample application and the Basic database.
 2. Choose **Database > Load Data** to display the **Data Load** dialog box:

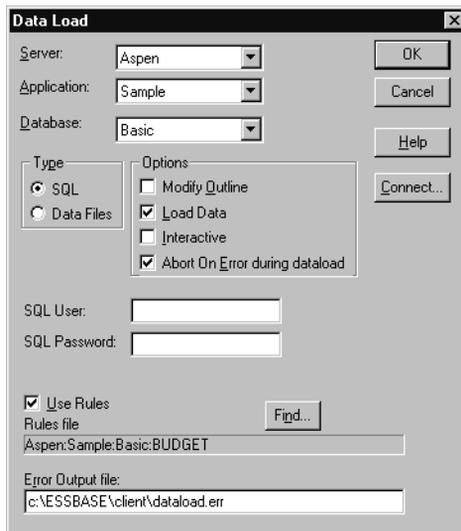


3. In the **Type** group of the **Data Load** dialog box, select the **SQL** option.
4. Select the **Use Rules** check box.

- Click **Find** to display the **Open Server Rules Object** dialog box:



- Ensure that the Sample application and Basic database are selected.
- Choose the BUDGET rules file from the **Objects** list box.
- Click **OK** to display the **Data Load** dialog box again.
- In the **Options** group, select the **Load Data** and **Abort On Error during dataload** check boxes. Ensure that all other options in the group are not selected:



10. Click **OK** to load the data.

You may need to type an SQL user name and SQL password when you connect to some SQL databases (for example, when you connect to an Oracle database). You do not need to enter a user name and password when you connect to a dBASE database.

When Essbase has loaded the data, a message box tells you that the load was successful:

**11.** Click **OK** to close the message box.

Essbase loads the SQL data into the Sample Basic database. The dBASE file contains budget data for the West and the East regions. Essbase loads the data into member combinations for these regions. If you have problems loading the data, see the *Essbase Database Administrator's Guide* for information on data loading or troubleshooting.

Using Non-MERANT ODBC Drivers

You can use ODBC database drivers with Essbase other than the MERANT drivers provided with SQL Interface. For example, you can use a Microsoft ODBC driver to connect to Microsoft Access, or use an IBM ODBC driver to connect to IBM DB2/6000. Some non-MERANT ODBC drivers are tested with Essbase. For more information on tested and supported drivers and data sources, see the *Essbase Installation Guide* or the `readme.txt` file.

To use a non-MERANT driver, you need to:

1. Configure the driver.
2. Configure the data source.

This chapter contains the following topics that describe how to use non-MERANT ODBC drivers:

- “Configuring a Non-MERANT Driver” on page 58
- “Creating a Configuration File for Non-MERANT Drivers” on page 58
- “Configuring Non-MERANT Data Sources” on page 65
- “Support for Non-MERANT Drivers” on page 70

Configuring a Non-MERANT Driver

When you configure a MERANT driver, Essbase recognizes the basic configuration information for the driver, especially the name of the driver and whether the name and password are case-sensitive. However, when you use a non-MERANT driver, you must give Essbase the configuration information for the driver that you use.

You give Essbase the configuration information for your non-MERANT driver by creating a driver configuration file called `esssql.cfg`, which is an ASCII text file. Place this file in your `ARBORPATH\BIN` directory on OLAP Server. This is `c:\hyperion\essbase\` if you installed Essbase in the default directory (`/home/hyperion/essbase/` on UNIX platforms).

Note: If you do not create a configuration file, Essbase uses its own default values. You may not be able to connect to the SQL database using the default values.

Creating a Configuration File for Non-MERANT Drivers

The configuration file must contain at least some of the following configuration values for each non-MERANT ODBC driver that you use. Required values may be different for different drivers. See the online help for each driver or see the documentation for specific information.

This section contains information about what you need to include in the configuration file, and in one case, how to find it:

- “Including the Minimum Required Contents” on page 59
- “Finding a Driver Name on Windows” on page 62
- “Finding a Driver Name on UNIX” on page 63
- “Example: Creating a Configuration File for the MS Access Driver on Windows” on page 63
- “Example: Creating a Configuration File for the IBM DB2 Driver on AIX” on page 64

Including the Minimum Required Contents

At a minimum, your configuration file must contain at least the *DriverName* for every non-MERANT ODBC driver that you use. Surround all values for each driver with brackets ([]). Leave one or more spaces between the configuration keyword and its corresponding value.

Use this template to help you create a configuration file:

```
[
Description           "Description of Driver"
DriverName            DriverName
UserId                Value
Password              Value
Database              Value
Server                Value
Application           Value
Dictionary            Value
Files                 Value
SingleConnection     Value
UpperCaseConnection  Value
IsQEDriver            Value
]
```

Description of Driver is your description of the driver. Enclose the description in quotation marks (" "). The default value is "".

DriverName is the file name of the non-MERANT ODBC driver and is required.

For all other configuration keywords, set *Value* to be either 1 or 0, according to is table:

Configuration Information	Value	Description
UserId	1	User ID required.
	0	User ID not required. Default: 0
Password	1	Password required.
	0	Password not required. Default: 0
Database	1	Database name required.
	0	Database name not required. Default: 0
Server	1	Server name required.
	0	Server name not required. Default: 0
Application	1	Application name required.
	0	Application name not required. Default: 0
Dictionary	1	Dictionary name required.
	0	Dictionary name not required. Default: 0
Files	1	Files name required.
	0	Files name not required. Default: 0
SingleConnection	1	Driver not thread-safe. One active connection allowed. Default: 1 Recommendation for MERANT drivers: 1
	0	Driver thread-safe. Multiple active connections allowed. CAUTION: Specifying 0 (multiple active connections allowed) for MERANT or non-MERANT drivers may lead to instability.
UpperCaseConnection	1	Driver not case-sensitive. Connection information is converted to uppercase.
	0	Driver case-sensitive. Connection information is unchanged. Default: 0

Configuration Information	Value	Description
IsQEDriver	1	Driver is a MERANT driver. You can specify configuration information for MERANT drivers (for example, if you have a later version of an MERANT driver which Essbase does not yet support).
	0	Driver is a non-MERANT driver. Default: 0

The defaults apply if you do not specify the value for the driver. These defaults are different from the Essbase default values that apply if you do not create an `esssql.cfg` file.

Include configuration information for all the non-MERANT drivers in one configuration file. Enclose the values for each driver in brackets ([]).

Use this template to help you create a configuration file:

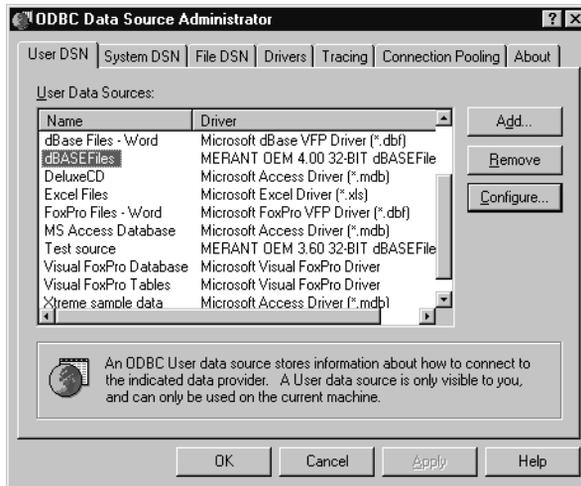
```
[
Description      "Description of Driver"
DriverName      DriverName
...
]

[
Description      "Description of Driver"
DriverName      DriverName
...
]
```

Finding a Driver Name on Windows

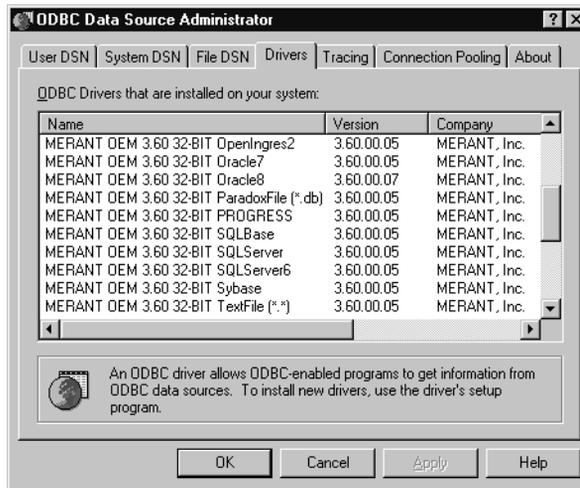
- To display the name of the drive, use this procedure:
 1. Start the ODBC Administrator using any method from Step 1 in “Configuring a Data Source on Windows” on page 19. If you do not have the ODBC components, see “New for Release 6.2” on page 16 for information about obtaining them.

The **ODBC Data Source Administrator** dialog box, **User DSN tab** appears:



Data sources that you have configured are in the **User Data Sources** list box. Drivers that have not been properly configured may also be in the User Data Sources list box. Ignore them.

2. Select the **Drivers** tab to display the **Drivers** screen:



3. Obtain the file name of the driver by scrolling to the right. For example, the file name for the Microsoft Access Driver is ODBCJT32.DLL.
4. Click **OK** when you have selected the driver.

For more information on the ODBC Administrator, see “Configuring an MS Access Data Source on Windows” on page 65.

Finding a Driver Name on UNIX

On UNIX, the driver’s name is in the `.odbc.ini` file. For more information on the `.odbc.ini` file, see “Configuring a Data Source on UNIX” on page 27.

Example: Creating a Configuration File for the MS Access Driver on Windows

The driver name for the Microsoft Access driver on Windows is ODBCJT32.DLL. This driver requires a database name, server name, application name, and dictionary (directory path). This driver is not a MERANT driver, so you must create a type its configuration file, `esssql.cfg` in ASCII-text format.

Use this example as a template:

```
[
Description           "Microsoft Access Driver (*.MDB)"
DriverName            ODBCJT32.DLL
Database              1
Server                1
Database              1
Application           1
Dictionary            1
IsQEDriver            0
]
```

Note: See “Configuring an MS Access Data Source on Windows” on page 65.

Example: Creating a Configuration File for the IBM DB2 Driver on AIX

The driver name for the IBM DB2/6000 driver on AIX is `DB2.0`. This driver requires a user ID, password, and database name, but not a server name. This driver is thread-safe and case-sensitive. It is not a MERANT driver, so you must create a configuration file, `esssql.cfg`, in ASCII-text format.

Use this example as a template:

```
[
Description           "IBM DB2/6000 ODBC Driver"
DriverName            DB2.0
UserId                1
Password              1
Database              1
SingleConnection      0
UpperCaseConnection  0
IsQEDriver            0
]
```

Note: See “Configuring an IBM DB2 Data Source on AIX” on page 69.

Configuring Non-MERANT Data Sources

Configure a non-MERANT data source in the same way that you configure a MERANT data source. See Chapter 2, “Configuring a Data Source” for more information. The following sections tell you how to configure a Microsoft Access data source on Windows NT and an IBM DB2/6000 data source on AIX:

- “Configuring an MS Access Data Source on Windows” on page 65
- “Configuring an IBM DB2 Data Source on AIX” on page 69

Note: Before you configure a non-MERANT data source, you must first configure the driver by creating an `esssql.cfg` configuration file. See “Creating a Configuration File for Non-MERANT Drivers” on page 58.

Configuring an MS Access Data Source on Windows

The following example describes how to configure a Microsoft Access data source on OLAP Server on Windows. Your data source and driver may differ, but the steps that you take to configure the data source are essentially the same. The example assumes:

- You have installed the Microsoft Access ODBC 32-bit database driver on OLAP Server.
- OLAP Server has access to a sample Microsoft Access (*.MDB) file.

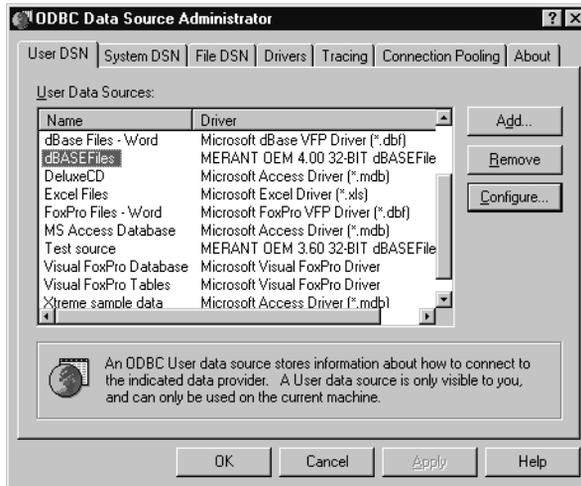
On Windows, use the ODBC Administrator, `ODBCAD32.EXE`, to configure a non-MERANT data source, just as you use it to configure a MERANT data source.

This example uses the `ORDERS.MDB` file, which is installed with Microsoft Access, but you can use any `.MDB` file to test the data source configuration.

The Microsoft Access ODBC 32-bit database driver is also installed with Microsoft Access, but it can be installed in other ways. For Microsoft Access, you need to specify the data source name and database. You can also provide a description. Other ODBC drivers may require additional information. See the documentation for each driver, and online help, for what is required.

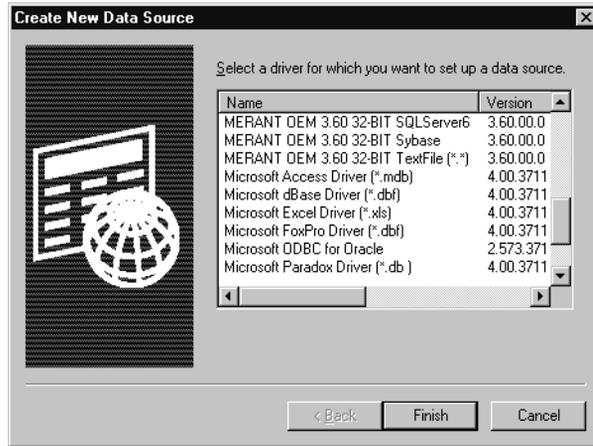
- To configure the Microsoft Access data source:
1. Start the ODBC Administrator in either of the ways described in Step 1 of “Configuring a Data Source on Windows” on page 19.

The **User DSN** tab of the **ODBC Data Source Administrator** dialog box is displayed:



Data sources that you have previously configured are displayed in the **User Data Sources** list box. Drivers that have not been properly configured may also be in the User Data Sources list box. Ignore or remove them.

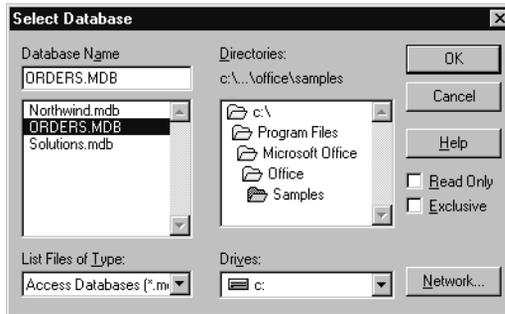
- Click **Add** to display the **Create New Data Source** dialog box:



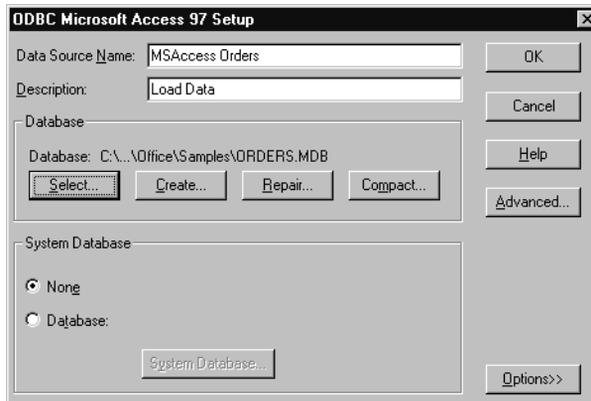
- Scroll down the list of available drivers and select **Microsoft Access Driver (*.mdb)**.
- Click **Finish** to display the **ODBC Microsoft Access 97 Setup** dialog box.



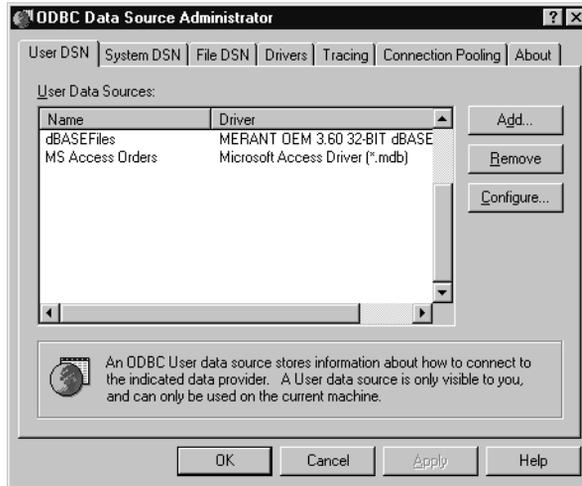
5. Type `MSAccess Orders`, or any name that will help you identify the source, in the **Data Source Name** text box.
6. Type `Load Data` in the **Description** text box.
7. Click **Select** in the **Database** group to display the **Select Database** dialog box.
8. Locate and select `ORDERS.MDB` (or the name of your `.MDB` file).



9. Click **OK**. The **ODBC Microsoft Access 97 Setup** dialog box is displayed with the path and file that you selected.



10. Click **OK** to display the **ODBC Data Source Administrator** dialog box.



The Microsoft Access data source is in the **User Data Sources** list box, with other data sources that may or may not have been configured.

11. Click **OK**.

You can now load data from the Microsoft Access data source file that you chose into a Essbase database. For information on loading SQL data, see Chapter 4, “Loading SQL Data.” For general information on loading data, see the *Essbase Database Administrator’s Guide*.

Note: Remember that source files must be accessible to OLAP Server. They can be on the OLAP Server machine or on the network.

Configuring an IBM DB2 Data Source on AIX

If you use AIX, you can connect to an IBM DB2/6000 data source using the IBM DB2/6000 ODBC driver. Consult the database administrator for information on configuration, authorization, binding, and other steps that may be required to connect to the data source.

Note: See “Example: Creating a Configuration File for the IBM DB2 Driver on AIX” on page 64.

Support for Non-MERANT Drivers

Some non-MERANT drivers are not tested or supported for use with Essbase. The driver that you choose might not be specifically tested to work with Essbase. For information on qualified drivers and data sources, see the *Essbase Installation Guide*.

Symbols

- * (wildcard characters), in SQL queries, 40
- , (commas), in SQL queries, 41
- .odbc.ini file (UNIX), 27
- = (equal signs), in SQL queries, 41

A

- Access, Microsoft
 - connecting with ODBC, 57
 - data source, 65
 - data source configuration, 66
 - ODBC driver, 67
- AIX environments, IBM DB2/6000
 - data source files, 69
 - driver, 64
- ARBORPATH, 58
- ASCII data, loading, 17

B

- Basic database, 37
- Basic outline, 50
- BIN directory, 58
- budget.dbf, 22, 25, 33
 - Sample application dBASE file, 8
 - selecting, 38
- building Hyperion Essbase dimensions, 17

C

- cfg (.CFG) file, creating, 58
- choosing, a driver, 20
- configuration file, for non-MERANT drivers, 58
- configuring

- data sources
 - dBASE, 20, 23
 - IBM DB2/6000, 69
 - Microsoft Access, 65
 - non-MERANT, 65
 - online help, 23, 26
 - overview, 19
 - IBM DB2/6000, ODBC driver, 64
 - Microsoft Access ODBC driver, 63
 - non-MERANT drivers, 58, 61
- connecting
 - to IBM DB2/6000 data source, 69
 - to SQL databases, 55
- converting SQL statements, 18
- CREATE privileges, 32
- creating
 - data load rules file, 36
 - driver configuration file, 58, 60
 - for IBM DB2/6000, 64
 - Microsoft Access driver configuration file, 63
 - single table or view, 30 to 31
 - required security privileges, 32

D

- data
 - load preparation, 29
 - load preparation methods, 31
 - SQL, mapping, 36
 - values, fields, 47
- Data Load Properties tab, 47
- Data Load Rules button, 37
- data load rules file
 - associating with outline, 50
 - creating, 36

- definition, 36
 - saving, 52
 - SQL records, 18
 - validating, 51
 - Data Prep Editor
 - button for opening, 37
 - displaying fields, 45
 - opening, 37
 - data sources
 - configuring
 - for IBM DB2/6000, 69
 - for Microsoft Access, 66
 - connecting to, 69
 - dBASE, 20, 23
 - describing, 68
 - Microsoft Access, 69
 - naming, 22, 25, 68
 - non-MERANT, 65
 - paths, 40
 - paths for Microsoft Access data source, 68
 - selecting, 38
 - storing on network, 69
 - storing on server, 39
 - databases
 - Basic, 33, 37
 - IBM DB2/6000, 57
 - Microsoft Access, 57
 - multidimensional, 7
 - multidimensional data, summary level, 17
 - DB2.0, 64
 - DB2/6000
 - See* IBM DB2/6000, 57
 - dBASE files
 - configuring, 20, 23
 - data loading, 33
 - selecting, 38
 - deleting white spaces, 48
 - describing data sources, 22, 25
 - dialog box
 - Open Server Rules Object, 54
 - Select Database, 68
 - dialog boxes
 - Associate Server Outline Object, 50
 - Create New Data Source, 21, 24, 67
 - Data Load, 53
 - Define SQL, 32, 39
 - Field Attributes, 46
 - OCBC Microsoft Access 97 Setup, 68
 - ODBC Data Source Administrator, 20, 24, 62
 - ODBC dBASE Driver Setup, 21, 25
 - Save Server Object, 52
 - Select Server, Application and Database, 38
 - Validate Rules, 51
 - dimensions
 - building, 17
 - mapping to fields, 46
 - directories
 - BIN, 58
 - data source, 22, 25
 - data sources, default, 40
 - driver configuration file, 58
 - driver configuration file
 - creating, 58, 60
 - for IBM DB2/6000, 64
 - for Microsoft Access, 63
 - default values, 58, 60
 - in BIN directory, 58
 - keyword values, 60
 - required values, 58
 - syntax, non-MERANT drivers, 61
 - drivers
 - choosing in ODBC Administrator, 19
 - configuring
 - for IBM DB2/6000, 64
 - non-MERANT, 65
 - description of, 59
 - file names, 63
 - Microsoft Access, 65, 67
 - naming, 59
 - non-MERANT, 57
 - tested, 70
 - password, 58
 - UNIX, names, 63
- ## E
- Essbase
 - default values for drivers, 58, 60
 - loading data into, 7
 - outline, mapping data to, 36
 - server on Windows NT, 65
 - esssql.cfg, 58, 60 to 61, 64

Excel, loading data from, 17

F

fields

- Data Prep Editor, 45
- data values, 47
- mapping to dimensions, 46
- selecting, 39
- selecting or naming, 40
- validating names of, 51

files

- .CFG (config), 58
- .DBF (dBASE), 36
- .MDB, 65, 67
- data load rules, 36
- flat file data, loading, 7
- FROM statement, 39 to 40
- to join tables, 41

H

help, online, 23, 26

Hyperion Essbase Application Manager

- joining tables, 31
- loading spreadsheet data, 17
- loading SQL data, 33
- SELECT statement, 31 to 32

Hyperion Essbase, loading data into, 33

I

IBM DB2/6000

- data source, 69
- data source database, 57
- data source, ODBC driver for, 64
- driver and database, 57

installed ODBC drivers, listed, 63

interpreting SQL records, 18

J

joining tables, 31

- FROM statement, 41
- Hyperion Essbase Application Manager, 31
- SELECT statement, 32

K

key fields, 41

keywords, driver configuration file, 60

L

loading data, 7

- data load rules, 36
- from single table, 30
- from spreadsheets, 17
- into outline, 53
- joining tables, 31 to 32
- member combinations, 55
- overview, 18, 33
- preparing for, 29
- summary level data, 17
- using Hyperion Essbase Application Manager, 33

Lotus 1-2-3, loading data from, 17

M

mapping

- fields to dimensions, 46
- SQL data to Essbase outline, 36

MDB (.MDB) file, 65, 67

member combinations, 55

Microsoft Access

- data source, 65 to 66, 68 to 69
- driver, 67
- ODBC driver for, 63

Microsoft, driver and database, 57

multidimensional databases

- building Hyperion Essbase dimensions, 17
- loading data into, 7, 18

multiple tables

- joining, 31 to 32
- loading from, 29
- preparing, 29

N

naming data sources, 22, 25

network, data source files, 69

non-MERANT

- data source, configuring, 65
- drivers

- configuring, 57
 - tested, 57
- drivers tested, 70

O

ODBC Administrator, 19, 66

- online help, 23, 26
- to configure non-MERANT driver, 65

ODBC drivers

- configuring for IBM DB2/6000, 64
- file names, 63
- Microsoft Access, 65, 67
- non-MERANT, 57
- tested, 57
- UNIX, names, 63

odbc.ini file, 39

ODBCAD32.EXE, 19, 65

ODBCJT32.DLL, 63

online documentation, help in ODBC Administrator, 23, 26

Open Server Rules Object dialog box, 54

opening Data Prep Editor, 37

ORDERS.MDB, 68

ORDERS.MDB, testing data source configuration, 65

outline

- associating with rules, 50
- Basic, 50
- loading data, 53

P

passing

- SELECT statements, 32
- SQL statements, 18

passwords

- for driver, 58
- for IBM DB2/6000 driver, 64
- for SQL database, 55

paths

- esssql.cfg, 58
- for data sources, 40

processing time, 31

R

relational data, loading, 17

- See also* data

requesting non-SQL data, 18

re-using data load rules file, 36

rules file. *See* data load rules file

S

Sample application, 8, 33, 37

- Basic database, 33, 37
- directory, 22, 25

saving rules files, 52

security privileges, 32

Select Database dialog box, 68

SELECT statement, 31 to 32

- processing time, 31
- selecting all fields, 40
- selecting specific fields, 39

selecting

- fields, 39 to 40
- SQL data sources, 38

servers

- data sources, 39
- machine, data source files, 69

spaces, deleting, 48

spreadsheets, loading data, 17

SQL

- queries, 40
- statements, 18

SQL data

- joining tables, 31
- loading, 7, 17
- mapping to Essbase, 36
- selecting source, 38

SQL databases

- connecting to, 55
- single tables in, 30 to 31

SQL server, 31

summary data, loading, 17

T

tables

- creating, 30
- efficiency, 31

- joining, 31 to 32
- loading from, 29
- SQL databases, 30
- tested
 - drivers, where to find information about, 57
 - non-MERANT drivers, 70
- text file data, loading, 17

U

- UNIX environments
 - budget.dbf, 33
 - configuring data sources on, 27
 - IBM DB2/6000 driver, 64
 - IBM DB2/6000, data source files, 69
 - odbc.ini file, 39, 63
- user names, 55

V

- validating rules file, 51
- views, 31
 - creating, 30
 - SQL databases, 30

W

- WHERE statements, 39 to 40
- WHERE statements key fields, 41
- white spaces, deleting, 48
- Windows NT environments
 - configuring data sources on, 19
 - configuring dBASE data sources, 20, 23
 - displaying driver names, 66
 - Essbase server, 65
 - ODBC Administrator, configuring data sources, 19
 - ODBCAD32.EXE, 19

