Hyperion® Integration Server

Release 2.0

Installation Guide

Hyperion®

Hyperion Solutions Corporation

P/N: H70D07-2000000

Notice: The information contained in this document is subject to change without notice. Hyperion Solutions Corporation shall not be liable for errors contained herein or consequential damages in connection with the furnishing, performance, or use of this material.

Hyperion Solutions Corporation
1344 Crossman Avenue
Sunnyvale, CA 94089

Printed in the U.S.A.

Contents

Preface

Chapter 1: Preparing to Install Hyperion Integration Server

Chapter 2: Installing Server Software

Chapter 3: Installing Client Software

Chapter 4: Creating OLAP Metadata Catalogs

Chapter 5: Configuring Relational Data Sources

Chapter 6: Setting up the Sample Application

Chapter 7: Performing Advanced Installation Tasks

Index

Preface

Purpose

This guide provides you with the information that you need to install and configure OLAP Integration Server, Hyperion Integration Server Desktop, OLAP Metadata Catalog, OLAP Command Interface, and the sample application on all supported platforms.

Audience

This guide is for database administrators who are responsible for installing, implementing, and deploying the Hyperion Integration Server software. This guide is also for client users who need to install Hyperion Integration Server Desktop and OLAP Command Interface client software.

To use the information in this book, you need the following skills:

- Knowledge of where the data for your business resides (for example, in a relational database)
- A fundamental understanding of Microsoft Windows and basic Microsoft Windows terminology, such as dialog box, list box, and button.

  See Microsoft Windows documentation for more information on these terms.
- For UNIX system users, a fundamental understanding of UNIX system administration, such as directory navigation, environment variables, login scripts, and text file editing

  See the documentation for the UNIX system you are using for more information.

Document Structure

This document contains the following information:

Chapter 1, "Preparing to Install Hyperion Integration Server," describes the tasks that you must complete before you start the installation process.

Chapter 2, "Installing Server Software," describes how to install the server component of Hyperion Integration Server (OLAP Integration Server) and related components.

Chapter 3, "Installing Client Software," describes how to install the client component of Hyperion Integration Server (Hyperion Integration Server Desktop and related components).

Chapter 4, "Creating OLAP Metadata Catalogs," describes how to create the tables in an OLAP Metadata Catalog database.

Chapter 5, "Configuring Relational Data Sources," describes how to configure Open Database Connectivity (ODBC) data sources, including an OLAP Metadata Catalog and relational data sources.

Chapter 6, "Setting up the Sample Application," describes how to set up the sample application, from creating a sample OLAP Metadata Catalog (TBC_MD) to creating the sample relational data source (TBC).

Chapter 7, "Performing Advanced Installation Tasks," describes how the installation program modifies computer systems and how to perform manual configuration tasks on Windows systems and UNIX systems.

Index, provides a list of Hyperion Integration Server terms and their page numbers. In the online version, select an index entry to view that page.

Related Documentation

The following documents contain additional related information:

Hyperion Integration Server documents:

- *Hyperion Integration Server System Administrator's Guide*, for information about how to use OLAP Integration Server, OLAP Command Interface, and tables in the OLAP Metadata Catalog, and how to troubleshoot connections within Hyperion Integration Server.

- *Hyperion Integration Server Desktop OLAP Model User's Guide*, for information about basic multidimensional modeling concepts and about how to design, create, and maintain OLAP models based on a relational data source.
- *Hyperion Integration Server Desktop OLAP Metaoutline User's Guide*, for information about how to design, create, and maintain one or more metaoutlines based on an OLAP model.

Hyperion Essbase OLAP Server documents:

- *Hyperion Essbase Database OLAP Server Administrator's Guide*, for more information about basic multidimensional concepts and how to design, create, and maintain a Hyperion Essbase database.
- *Hyperion Essbase OLAP Server Installation Guide*, for information about how to install Hyperion Essbase software components on all supported platforms.
- *Hyperion Essbase SQL Interface Guide*, in online PDF format in the `\essbase\docs` directory, for information about how to use Hyperion Essbase SQL Interface to load data from SQL, relational, and flat-file data sources into a Hyperion Essbase OLAP Server database.
- *Hyperion Essbase Spreadsheet Add-in User's Guide for Excel*, in online PDF format in the `\essbase\docs` directory, for information about how to use Hyperion Essbase Spreadsheet Add-in with Microsoft Excel.
- *Hyperion Essbase Spreadsheet Add-in User's Guide for Lotus 1-2-3*, in online PDF format in the `\essbase\docs` directory, for information about how to use Hyperion Essbase Spreadsheet Add-in with Lotus 1-2-3 for Windows.
- The online *Quick Technical Reference* in the `\essbase\docs` directory, for a complete listing and description of functions, calculation commands, report commands, and file settings.
- The online API Reference in the `\essbase\docs` directory, for a complete listing and description of functions available through the Hyperion Essbase Application Programming Interface.

Online Help

➤ To access online help:

1. In any dialog box, click **Help**.
2. In the **Help** dialog box, click an item to display information about the item.

➤ To print an online help topic, display the topic and select File > Print.

Online Guides

The online guides are an electronic version of the printed documentation.

➤ To display an online guide, select Help > Help Topics.

Conventions

The following table shows the conventions used in this document:

Table viii: Conventions Used in this Document

Item	Meaning
➤	Arrows indicate the beginning of a procedure consisting of sequential steps.
Brackets []	In examples, brackets indicates 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 commands and various IDs. (Example: CLEARBLOCK command)
`Example text`	Courier font indicates that the material shown is a code or syntax example.

Table viii: Conventions Used in this Document (Continued)

Item	Meaning
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.
`Courier italics`	Courier italic text indicates a variable field in command syntax. Substitute a value in place of the variable shown in Courier italics.
Ellipses (…)	Ellipsis points indicate that text has been omitted from an example.
Mouse orientation	This document provides examples and procedures using a right-handed mouse. If you are using a left-handed mouse, adjust the procedures accordingly.
Menu options	Options in menus are shown in the following format: *Menu name* > *Menu command* > *Extended menu command* For example: View > Properties > Table
n, x	The variable *n* indicates that you must supply a generic number; the variable *x* indicates that you must supply a generic letter.

Note: The term right-click, used throughout this guide, means to click the secondary mouse button to open a pop-up menu.

Additional Support

In addition to the documentation and online help, Hyperion offers the following additional support for product information:

Ordering Documentation

Additional copies of printed documentation may be ordered through your local support office.

Training Services

Hyperion offers a variety of training options, including instructor-led training, custom training, and multimedia training. This training covers all Hyperion applications and technologies and is geared to end users, administrators, and information systems (IS) professionals.

Instructor-led training is delivered in formats and in locations suited to Hyperion's diverse, global customers. Hyperion Authorized Training Centers offer courses that they develop, as well as those developed by Hyperion. Custom training—training on the configured and tailored applications that employees use on the job—is another option to enhance user productivity and to ensure smooth day-to-day operations. Multimedia training—including computer-based training, Web-based training, and interactive distance learning—provides a cost-effective means of giving users a hands-on introduction to product features and functions. Computer-based training (CBT) provides high-quality interactive training at the user's convenience, regardless of location.

For more information about training, contact your local training services representative.

Consulting Services

Hyperion Consulting Services assists customers in maximizing the use of, and the return on investment in, Hyperion products. Experienced Hyperion consultants and Hyperion Alliance Partners assist organizations in tailoring solutions to their particular requirements, such as reporting, analysis, modeling, and planning. Specific services include implementation consulting, custom business solutions, data integration, and technical consulting. Additionally, Hyperion offers a variety of Services Packages and Reviews.

For more information about Consulting Services, Services Packages, and Reviews, as well as the services offered by Alliance Partners, contact your local consulting services representative.

Technical Support

Hyperion provides telephone and Web-based support to ensure that clients resolve product issues quickly and accurately. This support is available for all Hyperion products at no additional cost to clients with a current maintenance agreement.

When standard support does not meet specific requirements, a Hyperion support package that meets your needs can usually be designed. For more information, contact your local support office.

For information about technical support, visit the Hyperion Technical Support Web site located at `http://support.hyperion.com`.

Chapter

1

Preparing to Install Hyperion Integration Server

This chapter describes what you need to know and do before you install the Hyperion Integration Server product family. You install server and client software from one CD.

This chapter contains the following topics:

Hyperion Installation Workflow

Use the following workflow for installing Hyperion Integration Server, creating an OLAP Metadata Catalog, configuring data sources, and setting up the sample application:

1. **Get ready to install:** Verify that you have the necessary software product components and that the server and client computer systems on which you will install Hyperion Integration Server meet system requirements. Check for any migration issues. Install Hyperion Essbase, if necessary, and register the Hyperion Integration Server software.
2. **Install the server:** Install OLAP Integration Server on Windows NT, Windows 2000, or UNIX. For UNIX, you must perform additional manual configuration.
3. **Install the client:** Install Hyperion Integration Server Desktop on Windows 95, Windows 98, Windows NT, or Windows 2000.
4. **Create an OLAP Metadata Catalog:** Use the SQL scripts provided with Hyperion Integration Server to create an OLAP Metadata Catalog in a relational database.

5. **Configure data sources:** On the computer running OLAP Integration Server, create Open Database Connectivity (ODBC) connections to an OLAP Metadata Catalog and to the relational data source that you use for creating OLAP models and metaoutlines with Hyperion Integration Server.

6. **Set up the sample application:** Use the scripts provided with Hyperion Integration Server to create a sample database, sample OLAP Metadata Catalog, sample OLAP model, and sample metaoutline.

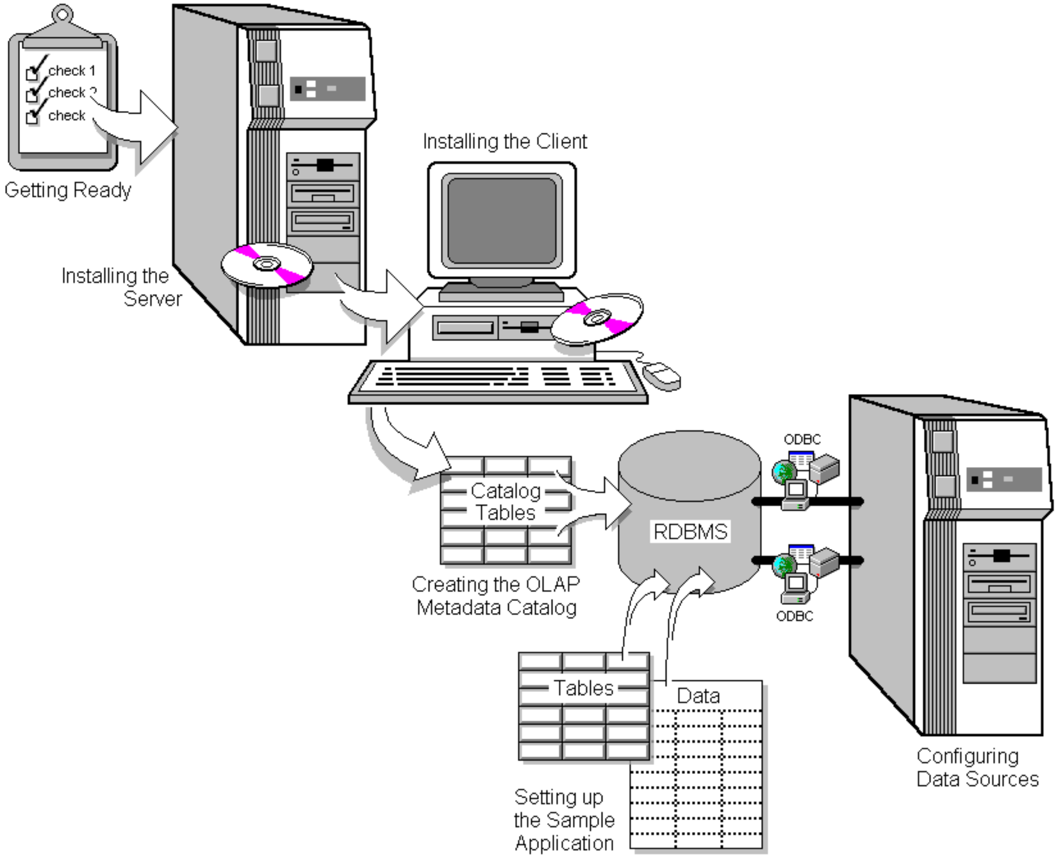


Figure 1: Workflow for Installing Hyperion Integration Server

Getting Ready to Install

Before installing Hyperion Integration Server, verify that you have all of the product components necessary for the software and make sure that your computer systems meet the system requirements for the software. In addition, if you are upgrading a previous release of Hyperion Integration Server, assess the migration issues before beginning the upgrade.

If Hyperion Essbase Release 6.0 or later is not already installed on a computer in your network, you must install it.

Verifying Product Contents

Before installing Hyperion Integration Server, make sure that you have all of the following product components:

- The CD that contains the Hyperion Integration Server product family software: Hyperion Integration Server Desktop, OLAP Integration Server, PDF documentation, and the `readme` file
- A software registration card, containing a new license number for each Hyperion Essbase OLAP Server that you can enable to work with Hyperion Integration Server
- The *Hyperion Integration Server Start Here* guide, describing system and platform requirements, migration requirements, and ODBC driver support
- The *Hyperion Integration Server New Features* guide, describing new features and documentation
- This book, the *Hyperion Integration Server Installation Guide*
- The *Hyperion Integration Server System Administrator's Guide*, describing how to use OLAP Integration Server, OLAP Metadata Catalog, and OLAP Command Interface and how to troubleshoot ODBC and server connections

- The *Hyperion Integration Server Desktop OLAP Model User's Guide*, describing how to create centralized and reusable OLAP models
- The *Hyperion Integration Server Desktop OLAP Metaoutline User's Guide*, describing how to create metaoutlines that are based on OLAP models and how to load members and data into a Hyperion Essbase database

Note: If the *Hyperion Integration Server Desktop OLAP Model User's Guide* and *Hyperion Integration Server Desktop OLAP Metaoutline User's Guide* are not included with your product components, refer to the Notice document in your product box.

Determining Product Component Configuration

Hyperion Integration Server includes software components that can run on a single computer or multiple computers. Before installing the product, determine where you will install each of the components. The following diagram illustrates the components of Hyperion Integration Server and a configuration in which all server components are installed on a single server computer.

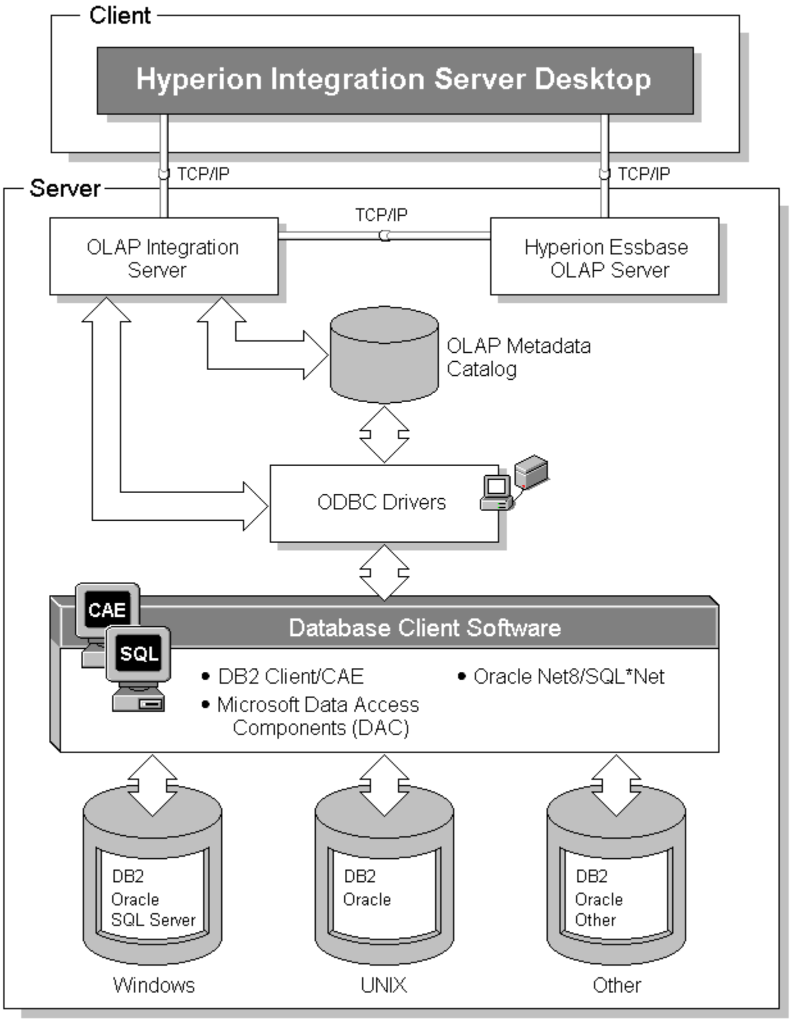


Figure 2: Configuring the Hyperion Integration Server Product Family

The server components illustrated in Figure 2 can be further distributed across additional computers. The Hyperion Essbase OLAP Server component and the relational database each can be installed on a separate computer. The OLAP Integration Server component, ODBC drivers, and the database client software for the relational database must be installed on a single computer.

Verifying System Requirements

Before installing Hyperion Integration Server, confirm that the computers on which you intend to install the software meet the basic system requirements. The installation requirements for the server software are different if you are installing on computers running Windows NT and Windows 2000, rather than on computers running UNIX. The client software also has different requirements from the server installation requirements.

For server installations, confirm that you have ODBC drivers that are compatible with both the relational database and the operating system of the server on which Hyperion Integration Server is installed. Hyperion Integration Server includes ODBC drivers from MERANT (formerly INTERSOLV). However, we recommend that you use the ODBC drivers provided by your relational database vendor. To verify the compatibility of the drivers with Hyperion Integration Server, the relational database, and the operating system of the server, see "Supported ODBC Drivers for Servers" on page 24.

Server System Requirements on Windows Systems

Hyperion Integration Server requires Hyperion Essbase Release 6.0 or later. The system requirements of Hyperion Integration Server server software on Windows NT and Windows 2000 are described in Table 1.

:

Table 1: System Requirements for Server Software on Windows NT or Windows 2000

Component	Requirement
Microprocessor	Pentium or later computers
RAM	• 64 MB or greater • Member Load Memory Requirements Formula: 10 MB + (700 Bytes x number of members)
OS version	• Windows NT 4.0, Service Pack 5 • Windows 2000

Table 1: System Requirements for Server Software on Windows NT or Windows 2000 (Continued)

Component	Requirement
Disk space	28 MB for OLAP Integration Server and OLAP Command Interface
	• 1 MB for OLAP Metadata Catalog (as shipped) • 30 MB for OLAP Metadata Catalog (on the server computer containing the catalog database) See Chapter 4, “Creating OLAP Metadata Catalogs.”
	• 3 MB for the sample application (as shipped) • 20 MB for the sample application (on the server computer containing the sample application database) See Chapter 6, “Setting up the Sample Application.”
	Optional items: • 10 MB for Hyperion Integration Server PDF documentation • 3 MB for MERANT driver PDF documentation
	• 45 MB TOTAL (as shipped) • 95 MB TOTAL (when set up)
Network protocol	TCP/IP

Note: `setup.exe` also uses some disk space on drive C during installation.

Server System Requirements on UNIX

Hyperion Integration Server requires Hyperion Essbase Release 6.0 or later.The system requirements of Hyperion Integration Server server software on AIX, HP-UX, and Solaris are described in Table 2.

Table 2: System Requirements for Server Software on AIX, HP-UX, and Solaris

Component	Requirement
Microprocessor	• AIX—PowerPC computers, including RS 6000 • HP-UX— PA-RISC 32-bit computers • Solaris—Sun SPARC or ULTRASPARC computers
RAM	• 64 MB or greater • Member Load Memory Requirements Formula: 50 MB + (700 Bytes x number of members)
OS version	• AIX—Version 4.3.3 • HP-UX—Version 11.0 (32-bit) • Solaris—Version 2.6, 7, or 8 (Sun OS—Version 5.6, 5.7, or 5.8)

Table 2: System Requirements for Server Software on AIX, HP-UX, and Solaris (Continued)

Component	**Requirement**
Disk space	28 MB for OLAP Integration Server and OLAP Command Interface
	• 1 MB for OLAP Metadata Catalog (as shipped) • 30 MB for OLAP Metadata Catalog (on the server computer containing the catalog database) See Chapter 4, “Creating OLAP Metadata Catalogs.”
	• 3 MB for the sample application (as shipped) • 20 MB for the sample application (on the server computer containing the sample application database) See Chapter 6, “Setting up the Sample Application.”
	Optional items: • 10 MB for Hyperion Integration Server PDF documentation • 3 MB for MERANT driver PDF documentation
	• 45 MB TOTAL (as shipped) • 95 MB TOTAL (when set up)
Network protocol	TCP/IP
Library	• AIX—kernel-threads with pthread API for AIX • HP-UX—kernel-threads • Solaris (Sun OS)—Solaris threads

Supported ODBC Drivers for Servers

Hyperion Integration Server supports the ODBC drivers for the relational database management systems (RDBMSs) and operating systems shown in Table 3:

Table 3: Supported ODBC Drivers per Operating System and RDBMS

	ODBC Driver per Supported Operating System			
Tested Relational Database Server (Database Client)	**Windows NT 4.0 Windows 2000**	**Solaris 2.6, 7, or 8 (Sun OS 5.6/5.7/5.8)**	**AIX 4.3.3**	**HP-UX 11.0[1]**
DB2 UDB 6.1 (DB2 UDB 6.1)	DB2 6.1 ODBC	DB2 6.1 ODBC	DB2 6.1 ODBC	DB2 6.1 ODBC
DB2 UDB 7.1 (DB2 UDB 7.1)	DB2 7.1 ODBC	DB2 7.1 ODBC	DB2 7.1 ODBC	DB2 7.1 ODBC
Oracle 8.04 (Oracle SQL *Net 8.0)	MERANT 3.6	MERANT 3.6	MERANT 3.6	MERANT 3.6
Oracle 8i (Oracle SQL *Net 8.0)	MERANT 3.6	MERANT 3.6	MERANT 3.6	MERANT 3.6
MS SQL Server 6.5 (No client required)	MS SQL Server 6.5 ODBC	Not currently supported	Not currently supported	Not currently supported
MS SQL Server 7.0 (no client required)	MS SQL Server 7.0 ODBC	Not currently supported	Not currently supported	Not currently supported
Sybase 11.*x*	Not currently supported	Not currently supported	Not currently supported	Not currently supported
Informix 9.*x*	Not currently supported	Not currently supported	Nor currently supported	Not currently supported

[1] MERANT 3.6 ODBC drivers are supported only on HP-UX 11.00 computers using 32-bit relational database client software.

Hyperion Integration Server provides MERANT (formerly INTERSOLV) 3.6 ODBC drivers for AIX, HP-UX, Solaris, Windows NT, and Windows 2000.

Note: ODBC drivers used with Hyperion Integration Server must be thread-safe. All drivers in the matrix are thread-safe. HP-UX requires kernel-threads. Solaris requires Solaris threads, and AIX requires kernel-threads with pthread API.

Client System Requirements on Windows Systems

The system requirements of Hyperion Integration Server client software (Hyperion Integration Server Desktop and OLAP Command Interface) for Windows 95, Windows 98, Windows NT, and Windows 2000 are described in Table 4.

Table 4: System Requirements for Client Software on Windows Systems

Component	Requirement
Microprocessor	Pentium or later computers
RAM	• 32 MB or greater for Hyperion Integration Server Desktop • 16 MB for Hyperion Essbase Spreadsheet Add-in on Windows 95, Windows 98, Windows NT 4.0 SP5, or Windows 2000
OS version	Windows 95, Windows 98, Windows NT 4.0 SP5, or Windows 2000
Disk space	• 45 MB for Hyperion Integration Server Desktop • 18 MB for common files
	Optional items: 10 MB for Hyperion Integration Server PDF documentation
	• 63 MB TOTAL • 73 MB TOTAL (to include Hyperion PDF documentation)
Network protocol	TCP/IP

Note: Hyperion Integration Server client software on Windows 95 requires Windows 95 patch 400.950B. Without the patch, icons and member names are misaligned, and icon fields are in reverse video. The patch is available only with new computers.

Migrating to Hyperion Integration Server Release 2.0

If you are migrating to Hyperion Integration Server Release 2.0 from a previous release, you should assess several system environment and data migration issues before installing the new software. This topic explains and recommends solutions for the following migration issues:

- "Preserving Existing ODBC Driver" on page 26
- "Choosing an Installation Directory" on page 27
- "Verifying Path Variable Settings" on page 27
- "Migrating Existing OLAP Metadata Catalogs" on page 29

Read the following topics carefully to assess migration issues that may affect your Hyperion Integration Server installation. Note that the following tasks cannot be completed until *after* you have installed Hyperion Integration Server:

- On UNIX systems, you cannot update environment variables until after the installation process has completed and the `is.csh` (or `is.sh`) files you need have been created (see "After You Install on UNIX Systems" on page 42).
- On all systems, you cannot migrate existing OLAP Metadata Catalogs until after the installation process has completed and the SQL scripts you need have been created (see "Upgrading the OLAP Metadata Catalog" on page 59).

Preserving Existing ODBC Driver

Starting with Release 2.0, Hyperion Integration Server has a new architecture that concentrates ODBC driver configuration solely on the computer that runs OLAP Integration Server. With this new architecture, you do not have to configure ODBC drivers for client computers that run Hyperion Integration Server Desktop or worry about upgrading ODBC drivers on clients.

If you use the INTERSOLV (now MERANT) Version 3.10 ODBC drivers supplied with Hyperion Integration Server Release 1.0 or Version 3.11 supplied with Release 1.1, you may want to preserve the older drivers for compatibility with the other applications. In general, however, you should upgrade the ODBC drivers to the versions provided with the latest release of Hyperion Integration Server. Hyperion Integration Server Release 2.0 supports MERANT Version 3.6 ODBC drivers.

Note: Hyperion Integration Server does not operate with MERANT Version 3.10 or Version 3.11 ODBC drivers.

Install the new MERANT Version 3.6 ODBC drivers and preserve the older drivers by installing Hyperion Integration Server to a new location. Be sure that you do not update the ODBC register settings when prompted to do so by the installation program. You can continue to use the previous ODBC definitions.

Choosing an Installation Directory

If you are upgrading from an earlier release of Hyperion Integration Server, you should install the software to the same directory as the earlier release to take advantage of the previous environment settings. However, if you are upgrading from Release 1.0 or Release 1.1 and want to preserve older ODBC driver settings for other applications, you should install Hyperion Integration Server to a different directory. For more information, see “Preserving Existing ODBC Driver” on page 26.

Verifying Path Variable Settings

If you are upgrading to the current release of Hyperion Integration Server from an earlier release, allow the `setup` program to update the environment variables for you. After installation is complete, however, you should check that the path variables are set properly for the system configuration:

- If you install Hyperion Integration Server on a computer that runs Hyperion Essbase, verify that the ARBORPATH environment variable is set to the main directory for Hyperion Essbase (default installation location on Windows systems is `C:\hyperion\essbase`).
- If you install Hyperion Integration Server on a computer that is not running Hyperion Essbase, verify that the ARBORPATH environment variable is set as follows:
  - For Windows systems: `ARBORPATH=`*ISHOME*`\esslib`
  - For UNIX systems: `ARBORPATH=$ISHOME` (see “After You Install on UNIX Systems” on page 42).

- For all upgrade scenarios, verify that the path environment variables are set as shown in Table 5 after installation:

Table 5: Path Variable Settings per Supported Operating System

Operating System	Path Variable Settings[1]

Windows NT or Windows 2000[2]	`PATH=`*`CURRENT_PATH`*`;%ISHOME%\Bin;`
	`%ISHOME%\odbclib;%ISHOME%\hyperlib;`
	`%ISHOME%\esslib`

AIX[3]	`LIBPATH=$LIBPATH:$ISHOME/odbclib/$ISHOME/esslib`
	`PATH=$PATH:$ISHOME/bin`

HP-UX[3]	`SHLIB_PATH $SHLIB_PATH:$ISHOME/ odbclib/$ISHOME/esslib`
	`PATH=$PATH:$ISHOME/bin`

Solaris[3]	`LD_LIBRARY_PATH=`
	`$LD_LIBRARY_PATH:$ISHOME/odbclib/$ISHOME/esslib`
	`PATH=$PATH:$ISHOME/bin`

[1] %ISHOME% represents the main installation directory of Hyperion Integration Server on Windows systems. $ISHOME represents the main installation directory of Hyperion Integration Server on UNIX systems.

[2] For Windows systems, check to make sure that the setup program did not add duplicate path entries for the Hyperion Integration Server directories.

[3] If you are using MERANT (formerly INTERSOLV) ODBC drivers, place the MERANT library at the beginning of the library path; for example, `LIBPATH=$ISHOME/odbclib:$LIBPATH:$ISHOME/esslib`).

For Windows systems, check that the `setup` program has not added duplicate path entries for the Hyperion Integration Server directories.

For more information about viewing and manually editing the paths and other environment variables on Windows systems, see "Manually Updating the Environment on Windows Systems" on page 97. For more information about viewing and manually editing the paths and other environment variables on UNIX systems, see "Manually Updating the Environment on UNIX Systems" on page 104.

Migrating Existing OLAP Metadata Catalogs

If you have an existing OLAP Metadata Catalog from an earlier Hyperion Integration Server release, you must upgrade it before using the OLAP Metadata Catalog with the current release of the software.

You cannot upgrade an existing OLAP Metadata Catalog until after you have completed the installation process.The Hyperion Integration Server setup program installs the scripts that you need for the upgrade (see "Upgrading the OLAP Metadata Catalog" on page 59).

CAUTION: If you have installed the sample application from a previous release of Hyperion Integration Server, you should back up your existing sample database, OLAP Metadata Catalog, and the OLAP models and metaoutlines stored in the catalog. You can then upgrade your existing catalog to be compatible with the current release of the software. You cannot, however, store new sample OLAP models and metaoutlines in your previous catalog.

To upgrade an existing OLAP Metadata Catalog, perform the following tasks:

- Complete the upgrade steps for the existing OLAP Metadata Catalog described in "Upgrading the OLAP Metadata Catalog" on page 59.
- When connecting to the relational database, make sure to use the same user name and password that you used when you created the original OLAP Metadata Catalog. This procedure adds new tables to the OLAP Metadata Catalog without disturbing the existing information.

CAUTION: After you upgrade an OLAP Metadata Catalog, you cannot roll back to the previous version. The new version of the OLAP Metadata Catalog is not compatible with previous releases of Hyperion Integration Server. In addition, do not attempt to use the Release 2.0 OLAP Metadata Catalog with previous Release 1.1 software; doing so could result in corrupted catalog data.

Registering the Hyperion Integration Server Software

Hyperion Integration Server requires Hyperion Essbase OLAP Server Release 6.0 or later to be installed on a computer on the network. If Hyperion Essbase is not already installed on the network, install it before installing Hyperion Integration Server.

After Hyperion Essbase is installed, you must enable at least one Hyperion Essbase OLAP Server on the network to work with the Hyperion Integration Server software. The Hyperion Integration Server product includes a registration card with a new license number for each Hyperion Essbase server that you are licensed to enable.

Note: If you are upgrading to a new release of Hyperion Integration Server and have already registered using the Hyperion Essbase `register` program, you do not need to register again.

➤ To enable a Hyperion Essbase server to work with Hyperion Integration Server, complete the following steps:

1. Stop the Hyperion Essbase server.
2. Find `register.exe` in *`ARBORPATH`*`\bin` (`register` in *`ARBORPATH`*`/bin` on UNIX) on the Hyperion Essbase server computer.
3. Run `register.exe` (`register` on UNIX), and enter your new license number exactly as it is written on your Hyperion Integration Server registration card.
4. Start the Hyperion Essbase server.
    - On Windows NT, *`ARBORPATH`*`\bin` is `C:\hyperion\essbase\bin` if you installed Hyperion Essbase in the default directory. For more information, see the *Hyperion Essbase Installation Guide*.
    - On UNIX, *`ARBORPATH`* is set to `/home/hyperion/essbase` if you installed Hyperion Essbase in the default directory. For more information, see "Manually Updating the Environment on UNIX Systems" on page 104. On UNIX, `register` is in the `bin` directory where you installed Hyperion Essbase. For more information, see the *Hyperion Essbase Installation Guide*.

    At runtime, Hyperion Integration Server Desktop and OLAP Integration Server perform a check to see that the target Hyperion Essbase server is enabled to work with Hyperion Integration Server software.
5. Keep the registration card in a secure place for future reference.

Getting Ready to Install Checklist

Use the following checklist to make sure that you are ready to install Hyperion Integration Server:

Getting Ready to Install

Verify Product Components

- □ Hyperion Integration Server installation CD
- □ Hyperion Integration Server software registration card
- □ Hyperion Integration Server documentation

Check System Requirements

- □ Server system requirements
- □ Server ODBC driver compatibility
- □ Client system requirements

Assess Migration Issues

- □ Preserve old ODBC drivers? Yes or No
- □ Choose installation directory location:

  ____

- □ Note current path variable settings:

  ____

- □ Decide whether to migrate existing OLAP Metadata Catalogs

Install Hyperion Essbase and Register Software

- □ Install Hyperion Essbase on network (if not installed)
- □ Register Hyperion Integration Server

Install Hyperion Integration Server

Check and Update After Installation

- □ Check updated path variable settings
- □ Migrate existing OLAP Metadata Catalogs

Chapter 2

Installing Server Software

This chapter describes the installation procedures for Hyperion Integration Server server software (OLAP Integration Server, OLAP Command Interface, OLAP Metadata Catalog scripts, and the sample application scripts, batch files, and data) and ODBC drivers on Windows NT, Windows 2000, and UNIX platforms. Read this chapter for information on server installation procedures and for information on starting OLAP Integration Server.

This chapter contains the following topics:

- "Installing Server Software on Windows Systems" on page 34
- "Starting the Server on Windows Systems" on page 38
- "Installing Server Software on UNIX Systems" on page 38
- "Starting the Server on UNIX Systems" on page 43

To complete the installation, see Chapter 3, "Installing Client Software," Chapter 4, "Creating OLAP Metadata Catalogs," and Chapter 5, "Configuring Relational Data Sources." To work with the sample application, see Chapter 6, "Setting up the Sample Application."

For detailed information about installed files, directories, environment changes made by the installation program, and uninstalling the software, see Chapter 7, "Performing Advanced Installation Tasks."

Tip: During installation, configuration, and testing, keep a list of all the user names and passwords you use to create or modify the system, including the applications with which they are associated and their purpose.

Installing Server Software on Windows Systems

The following procedure describes how to install the server software from the Hyperion Integration Server CD to a PC that is running Windows NT 4.0 S-5 or Windows 2000.

CAUTION: You must have system administrator privileges to install on Windows NT and Windows 2000. If you do not have appropriate privileges, parts of the installation will fail.

➤ To install the server software:

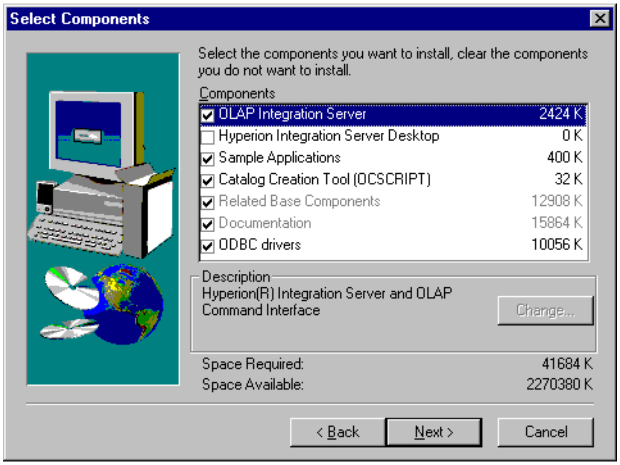
1. Perform the steps in "Getting Ready to Install" on page 17.
2. Close all applications.
3. Insert the Hyperion Integration Server CD into the CD-ROM drive and run `setup` from the root directory of the CD.
4. Follow the prompts, and provide the information requested.

   Note: To cancel installation at any time, click **Cancel**, and then click **Exit `setup`**.

5. In the **Choose Destination Location** dialog box, click **Next** to accept the default directory, or click **Browse** to enter a directory path or to select a directory where you want to install the server software.

   The default directory is `C:\hyperion\is`

6. In the **Select Components** dialog box shown in Figure 3, select the software components that you want to install.



Select the components you want to install, clear the components you do not want to install.

Components

☑	OLAP Integration Server	2424 K
□	Hyperion Integration Server Desktop	0 K
☑	Sample Applications	400 K
☑	Catalog Creation Tool (OCSCRIPT)	32 K
☑	Related Base Components	12908 K
☑	Documentation	15864 K
☑	ODBC drivers	10056 K

Description
Hyperion(R) Integration Server and OLAP Command Interface

Space Required: 41684 K
Space Available: 2270380 K

Figure 3: Clearing Client Products to Install Only Server Products

Custom Options

- If you want to install all server software components, the sample application, and the documentation, clear **Hyperion Integration Server Desktop** and leave all other components selected.
- If you want to install the minimum number of server software components, select **OLAP Integration Server, Catalog Creation Tool (OCSCRIPT), Related Base Components,** and **ODBC Drivers**, and clear all other components.

Note: If you want to install the client software, leave **Hyperion Integration Server Desktop** selected. If you install client software and server software on the same computer, you must install both in the same directory.

7. In the **Select Components** dialog box, click **Next** to install the software components that you have selected.

8. If you are installing Hyperion Integration Server for the first time, or if you are installing Hyperion Integration Server to a new location, in the **Question** dialog box, click **Yes** to update the environment variables.

   Note: On Windows NT and Windows 2000, if the PATH is too long, `setup` warns you that you must update the PATH manually. For information on how to update the PATH, see "Manually Updating the Environment on Windows Systems" on page 97.

9. If the **Overwrite** dialog box opens, take one the of following actions:

   - In the **Overwrite** dialog box, click **Yes** to install and overwrite the existing MERANT ODBC drivers.
   - In the **Overwrite** dialog box, click **No** to retain the existing MERANT ODBC drivers.

CAUTION: The current release of Hyperion Integration Server does not operate with MERANT Version 3.10 or 3.11 ODBC drivers. If you were using the ODBC drivers provided with an earlier release of Hyperion Integration Server, see "Migrating to Hyperion Integration Server Release 2.0" on page 26.

When you install MERANT (previously INTERSOLV) ODBC drivers, you overwrite MERANT ODBC drivers provided with earlier releases of Hyperion Essbase and Hyperion Integration Server. The `setup` program does not affect ODBC drivers provided by other vendors.

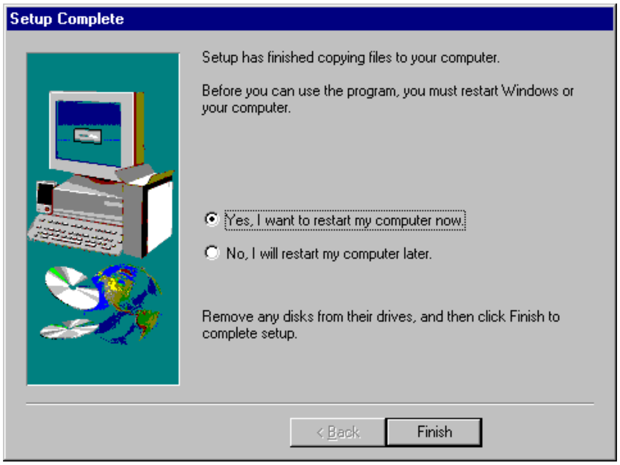
For information on MERANT ODBC drivers provided with Hyperion Integration Server, see "Supported ODBC Drivers for Servers" on page 24.

CAUTION: If you do not have the correct ODBC drivers installed, you will not be able to run Hyperion Integration Server.

10. If you do not want `setup` to launch the `readme` file, in the **Display the README file** dialog box, clear the check box and click **Finish**.

    If you chose to update the environment variables, the `setup` program updates the environment and displays the **Setup Complete** dialog box.

11. Select whether you want to restart the computer now or later, and click **Finish**.



Setup Complete

Setup has finished copying files to your computer.

Before you can use the program, you must restart Windows or your computer.

● Yes, I want to restart my computer now.

○ No, I will restart my computer later.

Remove any disks from their drives, and then click Finish to complete setup.

Figure 4: Choosing to Restart the Computer

Before you use Hyperion Integration Server, restart the computer to apply the changes `setup` made to the environment.

After You Install on Windows Systems

After installing the server software on Windows systems, you must perform additional steps before you can use Hyperion Integration Server:

- Install the Hyperion Integration Server Desktop client software

  See Chapter 3, "Installing Client Software."

- Create an OLAP Metadata Catalog

  See Chapter 4, "Creating OLAP Metadata Catalogs."

- Configure a data source and an OLAP Metadata Catalog using ODBC

  See Chapter 5, "Configuring Relational Data Sources."

- Verify that the Windows NT or Windows 2000 Schedule service is running

  See "Starting the Schedule Service on Windows NT and Windows 2000" on page 103.

Starting the Server on Windows Systems

➤ After you install the server software, complete the following steps to start OLAP Integration Server on Windows NT or Windows 2000:

1. On the Windows desktop, click **Start**.
2. Select Programs > Hyperion Integration Server > OLAP Integration Server to display the OLAP Integration Server window.

You can also start OLAP Integration Server by typing using OLAP Command Interface and typing **olapisvr** at the command prompt.

When you first start OLAP Integration Server, it creates the `olapisvr.log` log file in the `bin` directory. To write to a log file in another directory, type the following command at the command prompt:

```
olapisvr -Emydir/mylog
```

To stop OLAP Integration Server, close the OLAP Integration Server window, or use OLAP Command Interface. For more information, see the *Hyperion Integration Server System Administrator's Guide*.

Installing Server Software on UNIX Systems

OLAP Integration Server is available for the following UNIX platforms: AIX from IBM, HP-UX from Hewlett-Packard, and Solaris from Sun Microsystems.

This topic contains the following subtopics that describe server software installation on UNIX platforms:

- "Before You Install on UNIX" on page 39
- "Installing the Server Software" on page 39
- "After You Install on UNIX Systems" on page 42

Before You Install on UNIX

➤ Before you install server products on any UNIX platform, perform the following tasks:

1. Create a user named hyperion with read, write, and execute access to the directory where you will install OLAP Integration Server.

   Ideally, the hyperion user owns this directory. The user name you use to install and run the product does not have to be "hyperion," but must have the privileges defined in this step.

2. Create the directory where you will install OLAP Integration Server; for example, `$HOME/hyperion/is`.

   The installation program prompts you for the target directory and creates it if it does not exist.

Installing the Server Software

➤ To install server software from the Hyperion Integration Server CD on AIX, HP-UX, or Solaris, perform the following tasks:

1. Perform the steps in "Getting Ready to Install" on page 17.
2. Close all applications.
3. Insert the Hyperion Integration Server CD into the CD-ROM drive.
4. Change to the root directory of the CD.
5. From the root directory of the CD, run `setup.sh`.

   `setup.sh` detects the UNIX platform, then calls `essinst.sh`. The `essinst.sh` program decompresses and installs files from the `is.tar` file in the `AIX`, `HP-UX`, or `SOLARIS` directory of the CD.

   `setup.sh` prompts you to specify where you want to install OLAP Integration Server:

Enter the full path of the directory where you wish to install Hyperion Integration Server: [home/hyperion/is]?

6. To install Hyperion Integration Server in the default location (`/home/hyperion/is`) press **Return**, or enter the full path of the directory where you want to install the server software, and press **Return**.

   Note: The path for the directory cannot include spaces and cannot exceed 255 characters.

   `setup` prompts you to install the server:

Install OLAP Integration Server [yes] {yes no}?

7. Type **yes** and press **Return** to install OLAP Integration Server.

   `setup` prompts you to install the scripts that you will use to create OLAP Metadata Catalogs:

Install OLAP Metadata Catalogs scripts [yes] {yes no}?

8. Type **yes** and press **Return** to install the OLAP Metadata Catalog scripts.

   `setup` prompts you to install the MERANT (formerly INTERSOLV) ODBC drivers for OLAP Integration Server:

   (For a list of supported ODBC drivers, see "Supported ODBC Drivers for Servers" on page 24.)

Install MERANT ODBC drivers [no] {yes no}?

9. Type **yes** and press **Return** to install the MERANT ODBC drivers.

   CAUTION: If you do not have the correct ODBC drivers installed, you will not be able to run Hyperion Integration Server.

   `setup` prompts you to install the sample application script files:

Install Sample Application scripts [yes] {yes no}?

10. Type **yes** and press **Return** to install the sample application script files.

    `setup` prompts you to install the Hyperion Integration Server documentation:

Install Hyperion Integration Server documentation [yes] {yes no}?

11. Type **yes** and press **Return** to install the Hyperion Integration Server documentation.

    `setup` prompts you to update the environment variables file:

Update the file used for setting your environment variables (is.sh and is.csh) [yes] {yes no}?

12. Type **yes** and press **Return** to update the environment variables file.

Tip: Use the `is.sh` file to set environment variables for Korm shell or the `is.csh` file to set environment variables for C shell (see "After You Install on UNIX Systems" on page 42).

Note: OLAP Integration Server on UNIX requires a multithreaded environment. On AIX, HP-UX, and Solaris, the threads software is incorporated in the operating system. Make sure that the correct threads software for the operating system is properly installed before you use OLAP Integration Server. For more information, see "Server System Requirements on UNIX" on page 22.

If you install OLAP Integration Server and the Hyperion Essbase server on the same UNIX computer, make the following ARBORPATH variable settings:

- Before you start OLAP Integration Server, set the ARBORPATH variable to the directory where OLAP Integration Server is installed.
- Before you start the Hyperion Essbase server, set the ARBORPATH variable to the directory where the Hyperion Essbase server is installed.

Tip: To maintain two separate values for the ARBORPATH variable for OLAP Integration Server and Hyperion Essbase server on the same computer, consider running each program in a separate session or run each with a different user account.

After You Install on UNIX Systems

After you install server products on any UNIX platform, you must configure the environment for the hyperion user to allow OLAP Integration Server to run and access the relational data source. These manual steps include configuring the hyperion user login scripts to perform all of the following actions:

- Automatically run the `is.csh` (or `is.sh`) file and update the environment variables for OLAP Integration Server.

  Note: The `is.csh` and `is.sh` files are created by the installation process and are not available until after you have completed installation of Hyperion Integration Server (see "Manually Updating the Environment on UNIX Systems" on page 104).

- Set PATH and other environment variables for using ODBC drivers and making ODBC connections.
- Set PATH and other environment variables for accessing the relational data source, including settings for client access software.
- Edit the `odbc.ini` file to configure relational databases for use with ODBC.
- Give authorization to the hyperion user to schedule tasks using the cron scheduling daemon.

For more information about configuring the environment for the hyperion user, see "Manually Updating the Environment on UNIX Systems" on page 104.

After the configurations are complete, you can access the relational data source as the user hyperion to create an OLAP Metadata Catalog and—if you choose—to set up the sample application. For more information about creating the OLAP Metadata Catalog, see Chapter 4, "Creating OLAP Metadata Catalogs." For more information about setting up the sample application, see Chapter 6, "Setting up the Sample Application."

Starting the Server on UNIX Systems

After you install the server software and update the UNIX environment, you can run OLAP Integration Server or OLAP Command Interface from any directory.

➤ To run OLAP Integration Server in the foreground, type the following command:

```
olapisvr
```

➤ To run OLAP Integration Server in the background, type the following command:

```
nohup olapisvr &
```

The nohup command keeps the OLAP Integration Server program running even if you log off. The ampersand (&) suffix makes the program a background process.

CAUTION: To allow the Hyperion Integration Server Desktop client to create OLAP models and metaoutlines, you must run OLAP Integration Server and keep it running while any clients are connected to it.

When you start OLAP Integration Server, it begins writing to the `olapisvr.log` log file in the `bin` directory. To write to a log file in another directory, type the following command:

```
olapisvr -Emydir/mylog
```

Use OLAP Command Interface to stop OLAP Integration Server. For more information, see the *Hyperion Integration Server System Administrator’s Guide*.

Chapter

3

Installing Client Software

This chapter describes the installation procedures for Hyperion Integration Server client software (Hyperion Integration Server Desktop, OLAP Command Interface, OLAP Metadata Catalog scripts, and the sample application scripts, batch files, and data) on Windows 95, Windows 98, Windows NT, and Windows 2000. Read this chapter for information on client installation procedures, starting Hyperion Integration Server Desktop and OLAP Command Interface, and connecting to OLAP Integration Server, Hyperion Essbase OLAP Server, and relational data sources.

This chapter contains the following topics:

- "Installing the Client Software" on page 46
- "Starting Hyperion Integration Server Desktop" on page 48
- "Connecting to Servers and Relational Data Sources" on page 49

To complete the installation, see Chapter 4, "Creating OLAP Metadata Catalogs" and Chapter 5, "Configuring Relational Data Sources." To work with the sample application, see Chapter 6, "Setting up the Sample Application."

Tip: During installation, configuration, and testing, keep a list of all the user names and passwords you use to create or modify the system, including the applications with which they are associated and their purpose.

Installing the Client Software

The following procedure describes how to install the client software from the Hyperion Integration Server CD to a PC running Windows 95, Windows 98, Windows NT, or Windows 2000.

CAUTION: You must have system administrator privileges to install on Windows NT or Windows 2000. If you do not have system administration privileges, parts of the installation will fail.

➤ To install client software:

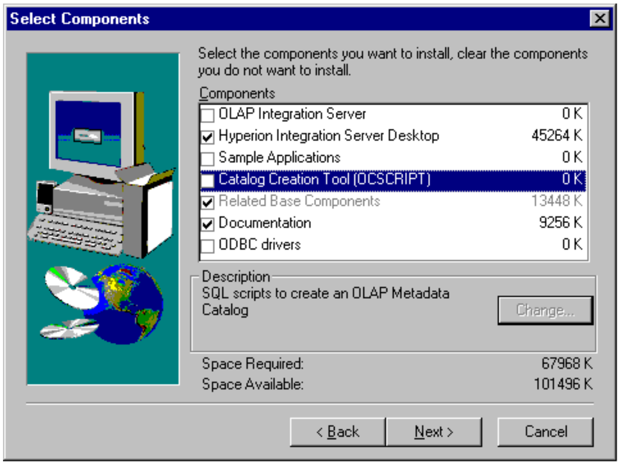
1. Perform the steps in "Getting Ready to Install" on page 17.
2. Close all applications.
3. Insert the Hyperion Integration Server CD into the CD-ROM drive and run `setup` from the root directory of the CD.
4. Follow the prompts and provide the information requested.

   Note: To cancel installation at any time, click **Cancel**, and then click **Exit `setup`**.

5. In the **Choose Destination Location** dialog box, click **Next** to accept the default directory, or click **Browse** to enter a directory path or to select the directory where you want to install the client software.

   The default directory is `C:\hyperion\is`

6. In the **Select Components** dialog box shown in Figure 5, select the software components that you want to install.



Select Components

Select the components you want to install, clear the components you do not want to install.

Components

□ OLAP Integration Server	0 K
☑ Hyperion Integration Server Desktop	45264 K
□ Sample Applications	0 K
□ Catalog Creation Tool (OCSCRIPT)	0 K
☑ Related Base Components	13448 K
☑ Documentation	9256 K
□ ODBC drivers	0 K

Description
SQL scripts to create an OLAP Metadata Catalog

Space Required: 67968 K
Space Available: 101496 K

Figure 5: Clearing Server Products to Install Only Client Products

Custom Options

- If you want to install all client software components, the sample application, and the documentation, clear **OLAP Integration Server** and leave all other components selected.
- If you want to install the minimum number of client software components, select **Hyperion Integration Server Desktop**, and **Related Base Components**, and clear all other components.

7. In the **Select Components** dialog box, click **Next** to install the software components that you selected.

8. In the **Question** dialog box, click **Yes** to update the environment variables in the following cases:

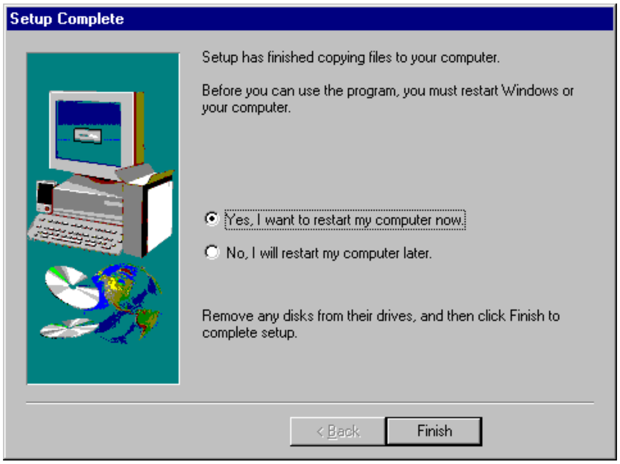
   - If you are installing Hyperion Integration Server for the first time
   - If you are installing Hyperion Integration Server to a new location

Note: On Windows NT or Windows 2000, if the PATH is too long, `setup` warns you that you must update the PATH manually. For information on how to update the PATH, see "Manually Updating the Environment on Windows Systems" on page 97.

9. In the **Display the README file** dialog box, clear the check box if you do not want `setup` to launch the `readme` file, and click **Finish**.

   If you chose to update the environment variables, the **Setup Complete** dialog box opens.

10. In the **Setup Complete** dialog box shown in Figure 6, select whether you want to restart the computer to implement the changes `setup` made to the environment, and click **Finish**.



Setup Complete

Setup has finished copying files to your computer.

Before you can use the program, you must restart Windows or your computer.

● Yes, I want to restart my computer now.

○ No, I will restart my computer later.

Remove any disks from their drives, and then click Finish to complete setup.

Figure 6: Choosing to Restart the Computer

Before you use Hyperion Integration Server, you must restart the computer to apply the changes `setup` made to the environment.

Starting Hyperion Integration Server Desktop

➤ After you install the client software, complete the following steps to start Hyperion Integration Server Desktop:

1. On the Windows desktop, click **Start**.
2. Select Programs > Hyperion Integration Server > Hyperion Integration Server Desktop.

Connecting to Servers and Relational Data Sources

To use Hyperion Integration Server Desktop to create OLAP models and metaoutlines, you must connect the client software to two servers: OLAP Integration Server and Hyperion Essbase OLAP Server. You must also connect to a relational data source and to an OLAP Metadata Catalog where you want to store the OLAP models and metaoutlines you create. Servers and relational data sources are typically located on computers on your network.

After you start Hyperion Integration Server Desktop, you are prompted to connect to an OLAP Integration Server. You can also connect to a Hyperion Essbase server and create default connection settings for both types of servers.

Note: An OLAP Metadata Catalog must be created and configured before you can connect to it. For information on creating an OLAP Metadata Catalog, see Chapter 4, "Creating OLAP Metadata Catalogs." For information on configuring an OLAP Metadata Catalog as a relational data source, see Chapter 5, "Configuring Relational Data Sources."

For more information on configuring relational data sources, see Chapter 5, "Configuring Relational Data Sources." For information on troubleshooting connections to servers, see the *Hyperion Integration Server System Administrator's Guide*.

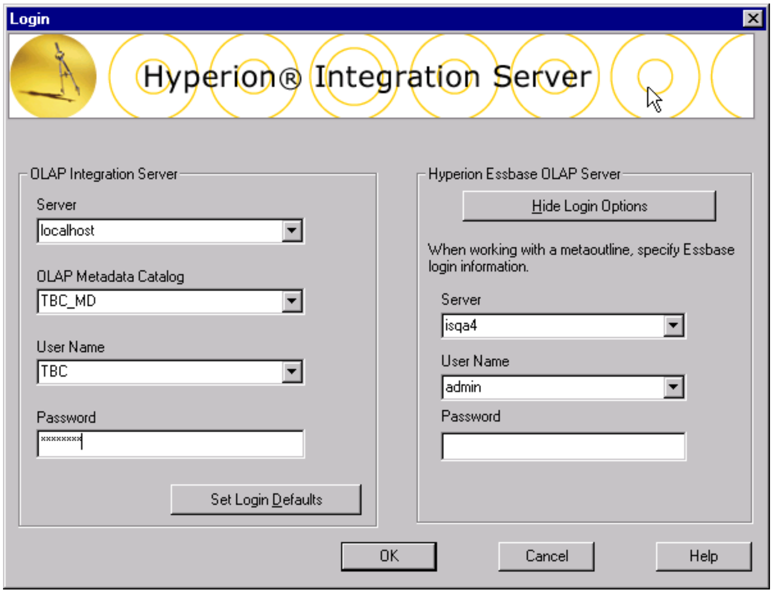
Connecting to OLAP Integration Servers

OLAP Integration Server manages access to OLAP Metadata Catalogs and to relational data sources. An OLAP Metadata Catalog stores information that defines the relational data source and the structure of OLAP model and metaoutline elements. To create an OLAP model or a metaoutline, you must first connect to an OLAP Integration Server.

Tip: The system administrator for OLAP Integration Server must provide you with a user name and password to an OLAP Metadata Catalog before you can connect to the server.

➤ To connect to an OLAP Integration Server:

1. To display the **Login** dialog box shown in Figure 7, from the OLAP Model main window or the OLAP Metaoutline main window, select Connections > OLAP Metadata Catalog > Connect.



Server: localhost

OLAP Metadata Catalog: TBC_MD

User Name: TBC

Password: ********

When working with a metaoutline, specify Essbase login information.

Server: isqa4

User Name: admin

Password:

Figure 7: Connecting to an OLAP Integration Server

2. In the **Server** text box, under **OLAP Integration Server**, enter the name of an OLAP Integration Server computer; for example, cypress.
3. In the **OLAP Metadata Catalog** text box, enter the name of an OLAP Metadata Catalog; for example, TBC_MD for the sample TBC (The Beverage Company) database.
4. In the **User Name** and **Password** text boxes, type your user name and password.

   Use the same user name and password that you use to connect directly to the relational database server and to access the database that contains the OLAP Metadata Catalog.

5. Click **OK**, or enter information for connecting to a Hyperion Essbase server.

   For more information, see "Connecting to Hyperion Essbase Servers" on page 51.

Connecting to Hyperion Essbase Servers

Hyperion Essbase OLAP Server creates and manages Hyperion Essbase databases. You do not have to connect Hyperion Integration Server Desktop to a Hyperion Essbase server to create an OLAP model or metaoutline, but you must connect to a Hyperion Essbase server to view a sample Hyperion Essbase outline or load members or data into a Hyperion Essbase database.

In the Login dialog box, you can specify the information for connecting to a Hyperion Essbase server, but Hyperion Integration Server Desktop does not connect to the server until you attempt to view a sample Hyperion Essbase outline or perform a member or data load.

Tip: The system administrator for the Hyperion Essbase OLAP Server must provide you with a user name and password before you can connect.

➤ To set connection information for a Hyperion Essbase server:

1. To display the **Login** dialog box, in the OLAP Model main window or OLAP Metaoutline main window, select Connections > OLAP Metadata Catalog > Connect.
2. In the **Server** text box, under **Hyperion Essbase OLAP Server**, enter the name of a Hyperion Essbase server computer; for example, sequoia.
3. Enter your Hyperion Essbase user name and password.

   The user name and password for the Hyperion Essbase server can be different from the OLAP Integration Server connection information.
4. Click **OK**.

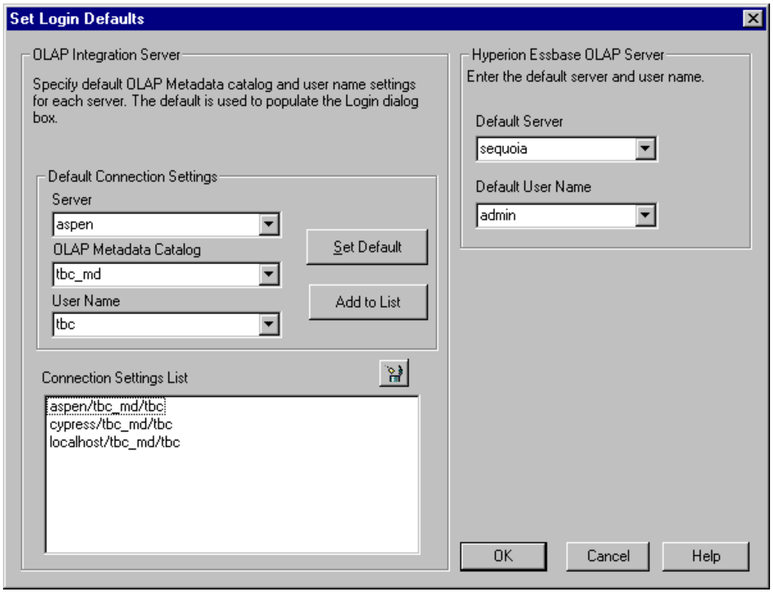
Setting Connection Defaults

If you typically use a specific OLAP Integration Server, OLAP Metadata Catalog, and Hyperion Essbase server, you can save the settings as the default. After you set default connection information, you can reuse these settings without needing to reenter the information.

If you use multiple OLAP Integration Servers and OLAP Metadata Catalogs, you can also set multiple relationships and select the appropriate one to use so that you do not need to reenter the information.

➤ To set connection defaults:

1. To display the **Login** dialog box shown in Figure 7, in the OLAP Model main window or OLAP Metaoutline main window, select Connections > OLAP Metadata Catalog > Connect.
2. To display the **Set Login Defaults** dialog box shown in Figure 8, click **Set Login Defaults** in the **Login** dialog box.



Set Login Defaults

OLAP Integration Server

Specify default OLAP Metadata catalog and user name settings for each server. The default is used to populate the Login dialog box.

Default Connection Settings

Server: aspen

OLAP Metadata Catalog: tbc_md

User Name: tbc

Connection Settings List

aspen/tbc_md/tbc
cypress/tbc_md/tbc
localhost/tbc_md/tbc

Hyperion Essbase OLAP Server

Enter the default server and user name.

Default Server: sequoia

Default User Name: admin

Figure 8: Set Login Defaults Dialog Box

3. In the **Server** text box, enter the name of an OLAP Integration Server computer; for example, aspen.
4. In the **OLAP Metadata Catalog** text box, enter the name of an OLAP Metadata Catalog; for example, TBC_MD for the sample TBC database.
5. In the **User Name** text box, enter a user name for the OLAP Metadata Catalog.

6. Take one of the following actions:
    - To set the connection defaults, click **Set Default**.
    - To save the connection information without setting the information as the default, click **Add to List**.
7. In the **Default Server** text box, under **Hyperion Essbase OLAP Server,** enter the name of a Hyperion Essbase server computer; for example, sequoia.
8. In the **Default User Name** text box, enter a user name for the server; for example, **admin**.
9. Click **OK** to return to the **Login** dialog box.

Connecting to Relational Data Sources

A relational data source is an external data repository—typically a large database—whose data you want to analyze by using Hyperion Essbase. You must connect Hyperion Integration Server to a relational data source to create OLAP models and metaoutlines and to load data into a Hyperion Essbase database.

Note: You can connect to only one relational data source at a time with Hyperion Integration Server Desktop.

➤ To connect to a relational data source:

1. In the Hyperion Integration Server Desktop **Welcome** dialog box, click the appropriate icon to create a new OLAP model or metaoutline, or select the **Existing** or **Recent** tab and double-click an OLAP model or metaoutline to open it for editing.
2. When prompted, select the data source to use; for example, TBC in the sample application.

    The external relational data source that you want to use must be configured on the computer that is running OLAP Integration Server. If the relational data source that you need is not visible in the scroll list, contact your database administrator or system administrator. For more information about troubleshooting server and relational data source connections, see the *Hyperion Integration Server System Administrator's Guide*.
3. In the **Data Source** dialog box, enter your user name and password, and click **OK**.

Chapter 4

Creating OLAP Metadata Catalogs

Before you run Hyperion Integration Server, you must create an OLAP Metadata Catalog to store the OLAP models that you create and the metaoutlines that you build from the OLAP models.

Note: If you intend to use the sample application to learn Hyperion Integration Server, skip to Chapter 6, “Setting up the Sample Application.” The sample application has its own relational data source, OLAP Metadata Catalog, OLAP model, and metaoutline. Return to this chapter when you are ready to use a relational data source to build OLAP models and metaoutlines.

This chapter tells you what to consider before you create an OLAP Metadata Catalog in a relational database and which SQL scripts you must run to create tables for the catalog. It assumes you know how to create a database using a relational database management system (RDBMS) and how to create tables by running SQL scripts.

This chapter contains the following topics:

About OLAP Metadata Catalogs

An OLAP Metadata Catalog is a relational database containing 32 tables in which Hyperion Integration Server stores OLAP models and metaoutlines.

Hyperion Integration Server supports the following RDBMSs:

- DB2 UDB
- Oracle
- MS SQL Server

The RDBMS for an OLAP Metadata Catalog can run on any platform as long as you have the ODBC driver and database client software to connect to it from the computer that runs OLAP Integration Server. The RDBMS for the relational data source that you use to create OLAP models and build metaoutlines also can run on any platform, if you have the ODBC driver and database client software to access it. For a detailed matrix of specifically supported databases and ODBC drivers, see "Supported ODBC Drivers for Servers" on page 24.

The RDBMS for the OLAP Metadata Catalog can be different from the RDBMS for the relational data source, and the platforms for the two RDBMSs do not need to be the same.

You can have more than one OLAP Metadata Catalog, but you cannot move OLAP models and metaoutlines from one OLAP Metadata Catalog to another.

Creating an OLAP Metadata Catalog

Note: To create a database for an OLAP Metadata Catalog, you must have database administrator or similar access privileges required by the RDBMS you are using.

➤ To create an OLAP Metadata Catalog, you must complete the following steps:

1. Perform the steps in "Getting Ready to Install" on page 17.
2. Create a database for OLAP Metadata Catalog tables using an RDBMS.

   See "Creating a Database for the OLAP Metadata Catalog Tables" on page 57.

3. Create tables for the OLAP Metadata Catalog database by running SQL scripts.

   See "Creating Tables for the OLAP Metadata Catalog" on page 57.

Creating a Database for the OLAP Metadata Catalog Tables

Create a database for the OLAP Metadata Catalog tables in the same way you create any database using the RDBMS.

- Create a database device or tablespace
- Allot 30 MB for storage
- Create user names and passwords, if needed
- Grant user privileges or permissions

Although an OLAP model must be created from a single relational data source, one OLAP Metadata Catalog can store all OLAP models and metaoutlines from numerous relational data sources. You can also create separate OLAP Metadata Catalogs to store additional OLAP models and metaoutlines for different projects.

Creating Tables for the OLAP Metadata Catalog

Create the tables for the OLAP Metadata Catalog with the same utility program you normally use to create tables by running SQL scripts.

The SQL scripts to create tables for the OLAP Metadata Catalog are in the `ocscript` directory where you installed Hyperion Integration Server. The utilities listed in Table 6 have been tested to work with the SQL scripts:

Table 6: SQL Scripts for Creating an OLAP Metadata Catalog

Database	SQL Script	Utility Program

DB2	`ocdb2.sql`
	`ocdrop_db2.sql`
	`ocdb2_upgrd20.sql`
		DB2 Command Center, or
		`>DB2 -tvf`

Oracle	`ocoracle.sql`
	`ocdrop_oracle.sql`
	`ocoracle_upgrd20.sql`
		SQL*Plus

Table 6: SQL Scripts for Creating an OLAP Metadata Catalog (Continued)

Database	SQL Script	Utility Program

MS SQL Server	`ocsqlsrv.sql`
	`ocdrop_sqlsrv.sql`
	`ocsqlsrv_upgrd20.sql`
		ISQL (MS SQL Server 6.5)
		Query Analyzer (MS SQL Server 7.0)

Hyperion Integration Server provides three SQL scripts for each RDBMS:

- `oc`*`database_name`*`.sql` to build tables
- `ocdrop_`*`database_name`*`.sql` to drop tables
- `oc`*`database_name`*`_upgrd20.sql` to upgrade tables.

If you need to rebuild tables, you must first drop the tables before you build them again.

➤ To create tables for the OLAP Metadata Catalog database, follow these steps:

1. Start the utility program.
2. Connect to the database you created for the OLAP Metadata Catalog as the user hyperion.
3. Open the appropriate SQL script file in the `ocscript` directory.
4. Run the SQL script to build tables.

   On SQL Server, you receive a message that you did not create data or rows. This message is normal because you created only tables and columns.
5. Verify that you have created the OLAP Metadata Catalog tables. For example, type the following command:

   ```
   SELECT * FROM JOIN_HINTS
   ```

   or start the RDBMS and verify that the OLAP Metadata Catalog has the new tables.
6. Close the utility program.

After You Create an OLAP Metadata Catalog

After you create an OLAP Metadata Catalog, you must map the catalog to a supported ODBC driver.

For more information about supported ODBC drivers, see "Supported ODBC Drivers for Servers" on page 24. For more information about configuring the OLAP Metadata Catalog as an ODBC data source, see Chapter 5, "Configuring Relational Data Sources."

Note: If you try to access an SQL Server database with the Microsoft native ODBC driver without access permission, SQL Server connects you to the default database without notifying you.

Upgrading the OLAP Metadata Catalog

If you have OLAP Metadata Catalogs from a release earlier than Hyperion Integration Server Release 2.0, you need to upgrade the OLAP Metadata Catalogs before you can use them with this release.

Upgrade the tables of an OLAP Metadata Catalog with the same utility program you normally use to create tables running SQL scripts. The SQL scripts to upgrade tables for the OLAP Metadata Catalog are in the `ocscript` directory where you installed Hyperion Integration Server. The upgrade scripts are named `oc`*`database_name`*`_upgrd20.sql` and are listed in Table 6, along with the utility programs with which they have been tested.

➤ To upgrade tables for the OLAP Metadata Catalog database, perform the following tasks:

1. Start the utility program.
2. Connect to the database you created for the OLAP Metadata Catalog as the user who created the original OLAP Metadata Catalog tables.
3. Open the appropriate SQL script file in the `ocscript` directory.

4. Run the SQL script to upgrade the tables.

   On SQL Server, you receive a message that you did not create data or rows. This message is normal because you created only tables and columns.

5. Verify that you have created the additional OLAP Metadata Catalog tables. For example, type the following command:

   ```
   SELECT * FROM OM_DESCRIPTION
   ```

   or start the RDBMS and verify that the OLAP Metadata Catalog has the new tables.

6. Close the utility program.

Chapter

5

Configuring Relational Data Sources

To create OLAP models or build metaoutlines, you must connect Hyperion Integration Server to both an OLAP Metadata Catalog and a relational data source. To make these connections, you must first configure each data source—that is, you map a supported Open Database Connectivity (ODBC) driver to each database. On Windows NT and Windows 2000, configure ODBC drivers using the ODBC Administrator. On AIX, HP-UX, and Solaris, edit the `odbc.ini` file. All configuration of ODBC data sources is done only on the computer that runs OLAP Integration Server.

Note: If a supported ODBC driver is already mapped to the relational data source, do not map it again. Simply map a supported ODBC driver to the OLAP Metadata Catalog.

This chapter contains the following topics that describe how to configure relational data sources:

About Configuring Relational Data Sources

A relational data source, including an OLAP Metadata Catalog, can run on any platform if you have the database client software and ODBC driver needed to access it. You must obtain and configure both the database client software and the ODBC driver.

The relational database management system (RDBMS) should include one of the following database clients:

- DB2 Client or Client Application Enabler (CAE)
- Oracle Net8 or SQL*Net
- SQL Server Client or Microsoft Data Access Components (DAC)

OLAP Integration Server runs on Windows NT, Windows 2000, AIX, HP-UX, or Solaris. You must install the following software on the OLAP Integration Server computer:

- OLAP Integration Server
- ODBC drivers
- Database client software

Hyperion Integration Server client software runs on Windows 95, Windows 98, Windows NT, or Windows 2000. You must install the Hyperion Integration Server Desktop software on Hyperion Integration Server client computers.

Note: Configure the ODBC connections to the relational data source and the OLAP Metadata Catalog only on the computer running OLAP Integration Server. You do not need to configure ODBC connections on Hyperion Integration Server client computers.

Figure 9 illustrates how to configure Hyperion Integration Server products.

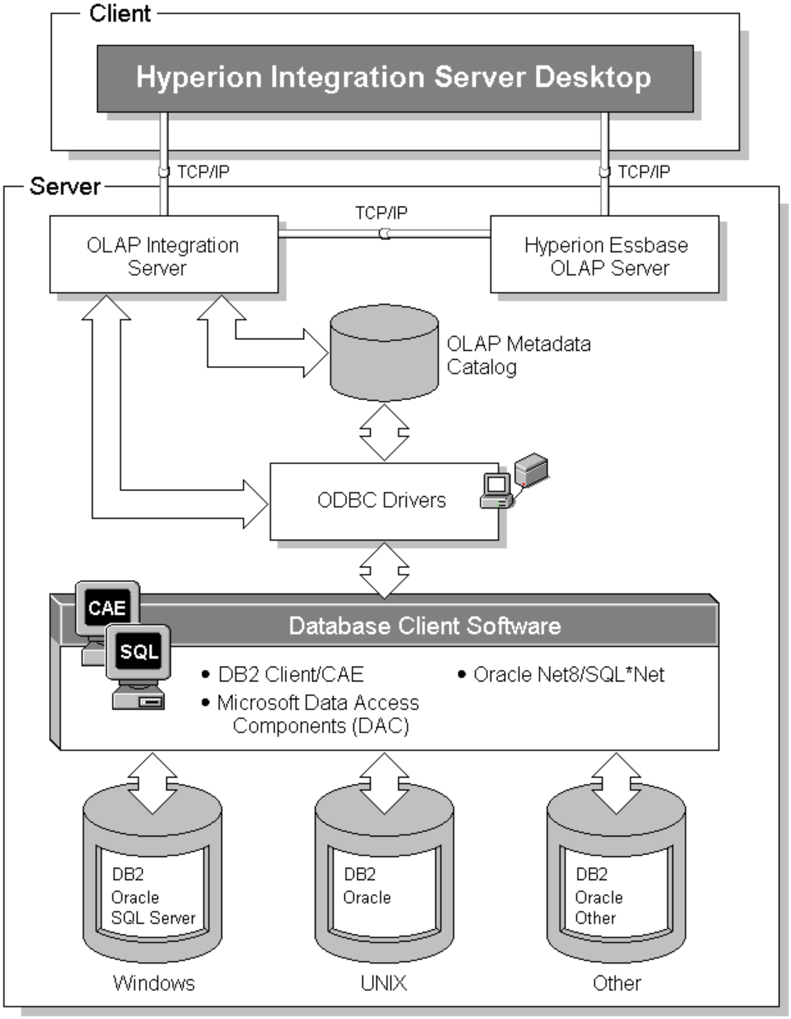


Figure 9: Configuring the Hyperion Integration Server Product Family

For more information on configuring Hyperion Integration Server product family components, see "Determining Product Component Configuration" on page 18.

Configuring the Database Client Software

The client software for the RDBMS must be installed on *the same computer as OLAP Integration Server*. The database client software is required—in addition to an ODBC driver—to make ODBC connections to RDBMS databases.

On UNIX systems, you must manually configure environment variables to access database client software and libraries. These variable settings must be defined for the environment of the user account (for example, hyperion) that you use to run OLAP Integration Server. For information on setting the UNIX environment to access the database client, see "Updating the Environment for the Database Client" on page 107.

On Windows systems, environment changes needed for running the database client software should be completed automatically by the RDBMS client installation program. For information about additional configuration of the database client software, see the installation documentation for the RDBMS that you are using.

Note: Before proceeding with the configuration of data sources using ODBC, make sure you can connect to RDBMS databases and access tables while logged on with the user account you use to run OLAP Integration Server. Use the SQL command line utility provided with the database client software that you are using.

For information about installing and configuring database client software, see the installation documentation for RDBMS that you are using.

Supported ODBC Drivers

Hyperion Integration Server supports specific ODBC drivers for connecting to RDBMSs. For a list of supported ODBC drivers by platform, see "Supported ODBC Drivers for Servers" on page 24.

For detailed information on the MERANT (formerly INTERSOLV) drivers provided with Hyperion Integration Server, see the *MERANT DataDirect Connect ODBC Reference* in PDF format in the `ODBCDocs` directory. Use Adobe Acrobat Reader in the `adobe` directory on the CD to read the PDF files.

For information on setting the UNIX environment to access ODBC libraries, see "Updating the Environment for ODBC" on page 106.

Configuring Data Sources on Windows Systems

To configure a relational data source on Windows NT or Windows 2000 systems, you must start ODBC Administrator and then create a connection to the data source that you will use for creating OLAP models and metaoutlines. Run the ODBC Administrator utility from the Windows Control Panel.

Note: This example creates a DB2 data source. The dialog boxes for other RDBMSs will differ.

➤ To configure a relational data source with ODBC Administrator, complete the following steps:

1. On the Windows desktop, select Start > Settings > Control Panel to open the **Control Panel** window.
2. In the **Control Panel** window, perform one of the following steps:
   - On Windows NT, double-click the **ODBC** icon to open the **ODBC Data Source Administrator** dialog box.
   - On Windows 2000, double-click the **Administrative Tools** icon, and then double-click the **Data Sources (ODBC)** icon to open the **ODBC Data Source Administrator** dialog box.
3. In the **ODBC Data Source Administrator** dialog box, click the **System DSN** tab.
4. Click **Add** to open the **Create New Data Source** dialog box.
5. In the driver list box of the **Create New Data Source** dialog box of ODBC Administrator, select an appropriate driver—for example, IBM DB2 ODBC Driver—and click **Finish** to open the **ODBC IBM DB2 Driver - Add** dialog box.
6. In the **ODBC IBM DB2 Driver - Add** dialog box **Database alias** drop-down list, select the name of the database for your relational source data (for example, TBC in the sample application).

   The name of the selected database is automatically displayed in the **Data Source Name** text box.
7. If you want to change the name of the data source, select the name displayed in the **Data Source Name** text box, press **Enter**, and type the name you want to use for the data source.

8. In the **Description** text box, type an optional description that indicates how you use this driver and click **Add**.

   For example, type the following words to describe the My Business database:

   Customers, products, markets

   You might type the following words to describe the sample application database:

   Sample relational data source

   The descriptions help to identify the available data sources for your selection when you connect from Hyperion Integration Server Desktop.

9. Click **OK** to return to the **ODBC Data Source Administrator** dialog box.

   The data source name you entered and the driver you mapped to it are displayed in the **System Data Sources** list box on the **System DSN** tab.

➤ To edit configuration information for a data source:

1. Select the data source name and click **Configure** to open the **ODBC IBM DB2 - Add** dialog box.
2. Correct any information you want to change.
3. Click **OK** twice to exit.

Configuring the OLAP Metadata Catalog on Windows Systems

To configure an OLAP Metadata Catalog on Windows NT or Windows 2000, start ODBC Administrator and then create a connection to the data source that contains the OLAP Metadata Catalog database.

Note: This example creates a DB2 data source. Dialog boxes for other RDBMSs will differ.

➤ To create a data source for the OLAP Metadata Catalog, complete the following steps:

1. On the desktop, select Start > Settings > Control Panel to open the **Control Panel** window.
2. In the **Control Panel** window, perform one of the following steps:
   - On Windows NT, double-click the **ODBC** icon to open the **ODBC Data Source Administrator** dialog box.
   - On Windows 2000, double-click the **Administrative Tools** icon, and then double-click the **Data Sources (ODBC)** icon to open the **ODBC Data Source Administrator** dialog box.
3. In the **ODBC Data Source Administrator** dialog box, click the **System DSN** tab.
4. Click **Add** to open the **Create New Data Source** dialog box.
5. In the **Create New Data Source** dialog box of ODBC Administrator, select an appropriate driver—for example, IBM DB2 Driver—in the list box, and click **Finish** to open the **ODBC IBM DB2 Driver - Add** dialog box.
6. In the **Database alias** drop-down list, select the name of the database for your OLAP Metadata Catalog (for example, TBC_MD in the sample application).

   The name of the selected database is automatically displayed in the **Data Source Name** text box.
7. If you want to change the name of the data source, select the name displayed in the **Data Source Name** text box, type a new name to indicate how you use this driver, and click **Add**.

   For example, you might type the following name to indicate that you are using the driver to connect to the first OLAP Metadata Catalog:

   OLAP Catalog #1

   You would type the following name to indicate that you are connecting to the sample application OLAP Metadata Catalog database:

   TBC_MD

8. In the **Description** text box, enter a description that indicates how you use this driver. For example, you might type the following words to describe the OLAP Metadata Catalog:

   My first models and metaoutlines

   You might type the following words to describe the sample application OLAP Metadata Catalog database:

   Sample models and metaoutlines

   The descriptions help you to identify the catalog that you want to select when you connect to the OLAP Metadata Catalog from Hyperion Integration Server Desktop.

9. Click **OK** to return to the **ODBC Data Source Administrator** dialog box.

   The data source name you entered and the driver you mapped to it are displayed in the **System Data Sources** list box on the **System DSN** tab.

To edit configuration information for a data source:

1. Click **Configure** to open the **ODBC IBM DB2 - Add** dialog box.
2. Correct any information you want to change.
3. Click **OK** twice to exit.

Configuring Data Sources on UNIX Systems

ODBC Administrator is not available on UNIX systems. On AIX, HP-UX and Solaris, you must manually set environment variables for ODBC and edit the `odbc.ini` file to configure the relational data source and OLAP Metadata Catalog.

Note: Edit the `odbc.ini` file if you add a new driver or data source or if you change the driver or data source.

Configuring ODBC Environment Variables

On UNIX systems, you must set environment variables to enable access to ODBC core components. The `is.sh` and `is.csh` shell scripts that set the required variables are provided in the Hyperion Integration Server home directory.

You must run one of these scripts before using ODBC to connect to data sources. You should include these scripts in the login script for the user name you use to run OLAP Integration Server. For information on setting the UNIX environment to access ODBC libraries, see "Updating the Environment for ODBC" on page 106.

Editing the odbc.ini File

To configure a data source in an `odbc.ini` file, you must add a name and description for the ODBC data source and provide the ODBC driver path, file name, and other driver settings in a separate section that you create for the data source name.

The Hyperion Integration Server installation program installs a sample `odbc.ini` file in the *ISHOME* directory. The file contains generic ODBC connection and configuration information for supported ODBC drivers. Use the file as a starting point to map the ODBC drivers that you use to the relational data source and OLAP Metadata Catalog.

Note: If you use a different file than the `odbc.ini` file, be sure to set the ODBCINI environment variable to the name of the file you use.

➤ To add a data source to an `odbc.ini` file:

1. On the computer running OLAP Integration Server, open the `odbc.ini` file by using a text editor such as vi.
2. Find the section starting with `[ODBC Data Sources]` and add a new line with the data source name and description; for example:

   ```
   mydata=data source for analysis
   ```

   To minimize confusion, the name of the data source should match the name of the database in the RDBMS.
3. Add a new section to the file by creating a new line with the name of the new data source enclosed in brackets; for example:

   ```
   [mydata]
   ```

4. On the lines following the data source name, add the full path and file name for the ODBC driver required for this data source and any other required ODBC driver information.

   Use the examples shown in the following sections as a guideline to map to the data source on your RDBMS.

   Note: Make sure that the ODBC driver file actually exists in the location you specify for the `Driver=` setting.

5. When you have finished editing `odbc.ini`, save the file and exit the text editor.

For more information about the `odbc.ini` file and the ODBC driver settings for each RDBMS, see the *MERANT DataDirect Connect ODBC Reference* in PDF format and the related `Readme` files in the `ODBCDocs` directory. For more information about native ODBC driver settings, refer to the installation documentation for your RDBMS native ODBC drivers.

Example of ODBC Settings for DB2

Following is an example of how you might edit `odbc.ini` to connect to a relational data source, db2data, on DB2 Version 6.1 (on AIX), using an IBM DB2 native ODBC driver.

Use the `vi $ODBCINI` command to edit the `odbc.ini` file and insert the following statements:

```
[ODBC Data Sources]
db2data=DB2 Source Data on AIX

...

[db2data]
Driver=/home/db2inst1/sqllib/lib/db2.o
Description=DB2 Data Source - AIX, native
```

Example of ODBC Settings for Oracle

Here is an example of how you might edit `odbc.ini` to connect to a relational data source, oradata, on Oracle Version 8 (on Solaris), using a MERANT Version 3.6 ODBC driver:

Note: `LogonID` and `Password` in this example are overridden with the actual values used in the Hyperion Integration Server user name and password.

```
[ODBC Data Sources]
oradata=Oracle8 Source Data on Solaris

...

[myoracle] Driver=
/export/home/users/dkendric/is200/odbclib/ARor815.so
Description=my oracle source
```

Note: ServerName=mytns`ServerName` must refer to the name of an Oracle database defined in the `tnsnames.ora` file. For more information, see the Oracle installation documentation.

Tip: Run `ivtestlib` to verify that the environment is set to run the correct odbc driver file. For example, run `ivtestlib` and paste the path and file name that follow `Driver=` in the `odbc.ini` file that you have edited.

Configuring the OLAP Metadata Catalog on UNIX Systems

Configuring an OLAP Metadata Catalog on AIX, HP-UX, and Solaris is similar to configuring a data source. For the OLAP Metadata Catalog database, add a data source name and section to the `odbc.ini` file, as described in "Editing the odbc.ini File" on page 69. No other changes are required.

You must create an OLAP Metadata Catalog database in a supported RDBMS before configuring it as an ODBC data source.

Here is an example how you might edit odbc.ini to connect to the OLAP Metadata Catalog, TBC_MD, on DB2 Version 6.1 (on Solaris), using a native ODBC driver:

```
[ODBC Data Sources]
ocd6a5a=db2 v6
...

[ocd6a5a]
Driver=/home/db2instl/sqllib/lib/db2.0
Description=db2
```

After You Configure a Data Source

After you configure the relational data source and OLAP Metadata Catalog, you can connect to them from Hyperion Integration Server. You can then create, modify, and save OLAP models and metaoutlines. For information on viewing OLAP models and metaoutlines for the sample application, see "Viewing TBC Tables, Columns, OLAP Models, and Metaoutlines" on page 85.

Note: The SQL Server ODBC driver may time out during a call to an SQL Server database. Try again when the database is not busy. Increasing the driver time-out period may avoid this problem. For more information, see the ODBC documentation for the driver you are using.

For more information on ODBC connection problems and solutions, see the *Hyperion Integration Server System Administrator's Guide*.

Chapter 6

Setting up the Sample Application

The Hyperion Integration Server installation program enables you to install a sample application with the server or client software. The sample application consists of a database from a fictitious company called The Beverage Company (TBC) and a sample OLAP model (TBC Model) and metaoutline (TBC Metaoutline) created from that database. You create a special OLAP Metadata Catalog to store the sample OLAP models and metaoutlines.

To set up the sample application, you must:

- Create the relational data source database for the sample application, TBC.
- Create a special OLAP Metadata Catalog database (TBC_MD) that contains an OLAP model (TBC Model) and a metaoutline (TBC Metaoutline) based on data in the sample TBC relational data source.

CAUTION: If you have installed the sample application from a previous release of Hyperion Integration Server, you should back up and preserve your existing sample database, OLAP Metadata Catalog, and the OLAP models and metaoutlines it contains. You can then upgrade your existing catalog to be compatible with the current release of the software (see "Upgrading the OLAP Metadata Catalog" on page 59). However, you cannot store newly-created OLAP models and metaoutlines in your previous catalog.

This chapter tells you what to consider before and after you set up the sample application and which scripts and batch files you must run to create and load the TBC database and the TBC_MD OLAP Metadata Catalog. This chapter assumes you know how to create a database using a relational database management system (RDBMS) and how to create tables and load them with data running SQL scripts, batch files, and shell scripts. For information on these topics, see the documentation for the RDBMS you are using.

This chapter contains the following topics:

About the Sample Application

You set up the sample application by creating two databases in an RDBMS: TBC (the sample database) and TBC_MD (an OLAP Metadata Catalog). You then create tables in each of these databases and load data into them, using scripts provided with Hyperion Integration Server.

After you create the TBC and TBC_MD databases in a supported RDBMS, you run four scripts to create the sample application:

- A table creation script for the TBC sample database, located in the `samples\tbcdbase` directory
- A data load script for the TBC sample database, located in the `samples\tbcdbase` directory
- A table creation script for the TBC_MD OLAP Metadata Catalog database, located in the `ocscript` directory
- A data load script for the TBC_MD OLAP Metadata Catalog database, located in the `samples\tbcmodel` directory

Different versions of the scripts are provided for each of the supported RDBMSs.

Note: Some of the sample application scripts require slightly different procedures, depending on the RDBMS you are using. Be sure to follow the procedure for your specific RDBMS.

Before You Set up the Sample Application

Before you set up the sample application, you must install the sample application scripts, catalog creation scripts, and the database client software for a supported RDBMS. When setting up the sample application, you must use a computer that has the necessary database client software installed. For more information, see "About Configuring Relational Data Sources" on page 62.

The sample application scripts do not have to be installed on the same computer, but you must be able to access the script files from the computer you are using.

Install the sample application scripts and catalog creation scripts using the Hyperion Integration Server installation program. For information about installing these scripts, see Chapter 2, "Installing Server Software."

In addition to the sample application scripts, the sample application requires the database client software for the RDBMS you are using. The database client software is required to run the sample application scripts. Verify that the database client software for the RDBMS is installed, including the client utility programs listed in the following table:

Table 7: Required Utilities for Setting Up the Sample Application

Database	**Utility Programs**
DB2 UDB	DB2 Command Window, or > DB2 -tvf
Oracle	• SQL*Plus • SQL*Loader command line utility
SQL Server	• ISQL (MS SQL Server 6.5 • Query Analyzer (MS SQL Server 7.0)

For more information about installing and configuring database client software, see "Configuring the Database Client Software" on page 64.

The remainder of this chapter assumes that you have installed the sample application scripts, the catalog creation scripts, and the required database client software. If you have not, see Chapter 2, “Installing Server Software.”

Setting up TBC

The relational data source for the sample application is TBC.

Note: To create a database, you must have database administrator or similar access privileges required by the RDBMS that you are using.

➤ To set up TBC, follow these steps:

1. Perform the steps in “Getting Ready to Install” on page 17.
2. Create the TBC database using an RDBMS.

   For more information, see “Creating the TBC Database” on page 76.
3. Create tables for the TBC database by running SQL scripts.

   For instructions, see “Creating Tables for the TBC Database” on page 77.
4. Load data into the TBC database by running SQL scripts.

   For instructions, see “Loading Data into the TBC Tables” on page 78.

Creating the TBC Database

Create the TBC database in the same way you create any database using an RDBMS:

- Create a database device or tablespace named TBC.
- Allot 20 MB for storage.
- Create a user TBC who can drop and create tables.
- Grant user privileges or permissions.

CAUTION: The user TBC must create the tables for the TBC database or portions of the sample application will not work.

Creating Tables for the TBC Database

Create tables for the TBC database with the same utility program you normally use to create tables by running SQL scripts.

The sample application SQL scripts needed to create tables for the TBC database are in the `samples\tbcdbase` directory where you installed Hyperion Integration Server.

The utilities listed in Table 8 have been tested to work with the SQL scripts:

Table 8: Tested Utilities for Creating TBC Tables

Database	SQL Script	Utility Program
DB2	`tbcdb2.sql`	DB2 Command Window, or `>DB2 -tvf`
Oracle	`tbcoracl.sql`	SQL*Plus SQL*Loader command line utility
SQL Server	`tbcsqlsv.sql`	ISQL (MS SQL Server 6.5) Query Analyzer (MS SQL Server 7.0)

CAUTION: You must create the tables for the TBC database as user TBC or portions of the sample application will not work.

➤ To create tables for the TBC database, follow these steps:

1. Start the utility program.
2. Connect to the TBC database as user TBC.
3. In the `samples\tbcdbase` directory, open the appropriate SQL script file.
4. Run the SQL script.

   On SQL Server, you receive a message that you did not create data or rows. This message is normal because you created only tables and columns.

5. Verify that you have created the TBC tables by typing the following command:

   ```
   SELECT * FROM TBC_MARKET
   ```

   or start the RDBMS and verify that the TBC database has the new tables.

6. Close the utility program.

Loading Data into the TBC Tables

To load data into the TBC tables, perform one of the following steps:

- Run an SQL script using the same utility program you normally use to load tables by running SQL scripts.
- Run a batch file from the MS-DOS command prompt.
- Run a shell script from the UNIX command prompt.

The utilities listed in Table 9 have been tested to work with SQL scripts, batch files, and shell scripts:

Table 9: Tested Utilities for Loading Data into TBC Tables

Database	SQL Script, Batch File, or Shell Script	Utility Program
DB2	`lddb2.sql`	DB2 Command Window, or `>DB2 -vf`
Oracle	`lcoracle.bat` (`lcoracle.ksh` on UNIX)	command line
SQL Server	`ldsqlsrv.bat`	command line

Loading Data into TBC on DB2

➤ To load data into the TBC tables on DB2, complete the following steps:

1. From the command line, move to the `samples\tbcdbase` directory where you installed OLAP Integration Server.
2. Start the utility program.
3. Connect to the TBC database as user TBC.
4. In the `sample\tbcdbase` directory, open the appropriate SQL script file.

5. Run the SQL script.
6. Verify that you have loaded data into the TBC tables; for example, type:

   ```
   SELECT * FROM TBC.LOOKUP_PRODUCT
   ```

   or start the RDBMS and execute a query.
7. Close the utility program.

Loading Data into TBC on Oracle and SQL Server

➤ To load data into the TBC tables on Oracle and SQL Server, complete the following steps:

1. From the command line, move to the `samples\tbcdbase` directory where you installed OLAP Integration Server.
2. Run the appropriate batch file or shell script; for example, on Windows 95, Windows 98, Windows NT, and Windows 2000, type:

   ```
   ldsqlsrv.bat servername password_for_sa
   ```

   Note: With SQL Server, *servername* is the database client name for the database.

   With Oracle, you must enter the user ID, password, and database name (*username/password@service_name*). For Oracle 8 on Windows NT and Windows 2000, type:

   ```
   ldoracle.bat TBC/password@service_name
   ```

   and for Oracle 7.3 or Oracle 8 on AIX, HP-UX, and Solaris, type:

   ```
   ldoracle.sh TBC/password@service_name
   ```

   Note: Service_name refers to the name of an Oracle database defined in the `tnsnames.ora` file. For more information, see the Oracle installation documentation.
3. Verify that you have loaded data into the TBC tables; for example, start the RDBMS, log in and connect to TBC as user TBC, and execute the following query:

   ```
   SELECT * FROM TBC_PRODUCT
   ```

Setting up TBC_MD

The OLAP Metadata Catalog for the sample application is TBC_MD. For more information on OLAP Metadata Catalogs, see Chapter 4, “Creating OLAP Metadata Catalogs.”

➤ To set up TBC_MD, complete the following steps:

1. Create a TBC_MD database using an RDBMS.

   For more information, see “Creating the TBC_MD Database” on page 80.

2. Create tables for the TBC_MD database by running SQL scripts.

   For instructions, see “Creating Tables for the TBC_MD Database” on page 81.

3. Load data into the TBC_MD database by running SQL scripts (or batch files or shell script files containing SQL scripts).

   For instructions, see “Loading Data into the TBC_MD Tables” on page 82.

Note: If you have a previous release of Hyperion Integration Server and are upgrading your OLAP Metadata Catalog, you cannot roll back to the previous version. The new version of the OLAP Metadata Catalog is not compatible with earlier releases of Hyperion Integration Server.

Creating the TBC_MD Database

Create the TBC_MD database in the same way that you create any database using an RDBMS:

- Create a database device or tablespace named TBC_MD.
- Allot 20 MB for storage.
- Create a user TBC who can drop and create tables.
- Grant user privileges or permissions.

CAUTION: You must create the tables for the TBC_MD database as user TBC or portions of the sample application will not work.

Creating Tables for the TBC_MD Database

Create tables for the TBC_MD database with the same utility program you normally use to create tables running SQL scripts.

The sample application SQL scripts used to create tables for the TBC_MD database are in the `ocscript` directory where you installed OLAP Integration Server.

The SQL scripts in the `ocscript` directory are the same scripts you use to create any OLAP Metadata Catalog. For information on OLAP Metadata Catalogs, see Chapter 4, "Creating OLAP Metadata Catalogs."

The utilities listed in Table 10 have been tested to work with the SQL scripts:

Table 10: Tested Utilities for Creating TBC_MD Tables

Database	SQL Script	Utility Program

DB2	`ocdb2.sql`
	`ocdrop_db2.sql`
	`ocdb2_upgrd20.sql`
		DB2 Command Center, or
		`>DB2 -tvf`

Oracle	`ocoracle.sql`
	`ocdrop_oracle.sql`
	`ocoracle_upgrd20.sql`
		SQL*Plus

SQL Server	`ocsqlsrv.sql`
	`ocdrop_sqlsrv.sql`
	`ocsqlsrv_upgrd20.sql`
		ISQL (MS SQL Server 6.5)
		Query Analyzer (MS SQL Server 7.0)

Hyperion Integration Server provides three SQL scripts for each RDBMS:

- `oc`*`database_name`*`.sql` to build tables
- `ocdrop_`*`database_name`*`.sql` to drop tables
- `oc`*`database_name`*`_upgrd20.sql`) to upgrade tables

If you need to rebuild tables, you must first drop the tables before you build them again.

CAUTION: You must create the tables for the TBC_MD database as user TBC or portions of the sample application will not work.

➤ To create tables for the TBC_MD database:

1. Start the utility program.
2. Connect to the TBC_MD database as user TBC.
3. In the `ocscript` directory, open the appropriate SQL script file.
4. Run the SQL script.

   On SQL Server, you receive a message that you did not create data or rows. This message is normal because you created only tables and columns.
5. Verify that you have created the TBC_MD tables; for example, type the following command:

   ```
   SELECT * FROM TBC.MS_INFO
   ```

   or start the RDBMS and verify that the TBC_MD database has the new tables.
6. Close the utility program.

Loading Data into the TBC_MD Tables

To load OLAP model and metaoutline data into the TBC_MD tables, perform one of the following steps:

- Run an SQL script using the same utility program that you normally use to load tables running SQL scripts.
- Run a batch file from the MS-DOS command prompt.
- Run a shell script from the UNIX command prompt.

The utilities listed in Table 11 have been tested to work with SQL scripts, batch files, shell scripts:

Table 11: Tested Utilities for Loading Data into TBC_MD Tables

Database	SQL Script, Batch File, or Shell Script	Utility Program
DB2	`lcdb2.sql`	DB2 Command Window, or `>DB2 -vf`
Oracle	`lcoracle.bat` (`lcoracle.ksh` on UNIX)	command line
SQL Server	`lcsqlsrv.bat`	command line

A batch file (`lcoracle.bat`) and a shell script file (`lcoracle.ksh`) are provided for Oracle 8 on UNIX systems.

Loading Data into TBC_MD on DB2

➤ To load data into the TBC_MD tables on DB2, complete the following steps:

1. From the command line, move to the `samples\tbcmodel` directory where you installed OLAP Integration Server.
2. Start the utility program.
3. Connect to the TBC_MD database as user TBC.
4. In the `samples\tbcmodel` directory, open the appropriate SQL script file.
5. Run the SQL script.
6. Verify that you have loaded data into TBC_MD; for example, type:

   ```
   SELECT * FROM TBC.MS_INFO
   ```

   or start the RDBMS and execute a query.
7. Close the utility program.

Loading Data into TBC_MD on Oracle and SQL Server

➤ To load data into the TBC_MD tables on Oracle and SQL Server, complete the following steps:

1. From the command line, move to the `samples\tbcmodel` directory where you installed OLAP Integration Server.
2. Run the appropriate batch file or shell script.

   With SQL Server, you must enter the server name and the password for user `sa`. For example, type:

   ```
   lcsqlsvr.bat servername password
   ```

   With Oracle, you must enter the user ID, password, and database name (*username*/*password*@*service_name*). For Oracle 8 on Windows NT and Windows 2000, type:

   ```
   lcoracle.bat TBC/password@service_name
   ```

   and for Oracle 8 on AIX, HP-UX, and Solaris, type:

   ```
   lcoracle.sh TBC/password@service_name
   ```

   Note: *Service_name* refers to the name of an Oracle database defined in the `tnsnames.ora` file. For more information, see the Oracle installation documentation.

3. Verify that you have loaded data into the TBC_MD tables; for example, start the RDBMS, log on, and connect to TBC_MD as user TBC, and execute the following query:

   ```
   SELECT * FROM TBC.MS_INFO
   ```

After You Set up the Sample Application

You must connect to both TBC and TBC_MD from Hyperion Integration Server Desktop to create, modify, and store TBC OLAP models and TBC metaoutlines. To make these connections, you must first map each database (TBC and TBC_MD) to a supported ODBC driver. For information on mapping the databases, see Chapter 5, "Configuring Relational Data Sources."

After you map TBC and TBC_MD to supported ODBC drivers, you can view TBC columns, tables, OLAP models, and metaoutlines in Hyperion Integration Server Desktop. For more information, see "Viewing TBC Tables, Columns, OLAP Models, and Metaoutlines" on page 85.

For a quick reference to where you are in the installation process, see "Hyperion Installation Workflow" on page 15.

Note: You must connect to TBC and TBC_MD as user TBC unless you create user name aliases or synonyms in the RDBMS. For more information, see "Creating Database User Aliases and Synonyms" on page 108.

Viewing TBC Tables, Columns, OLAP Models, and Metaoutlines

After you set up the sample application and configure TBC and TBC_MD by mapping them to supported ODBC drivers, you can view TBC tables, columns, OLAP models, and metaoutlines in Hyperion Integration Server Desktop.

➤ To see TBC tables and columns in the left frame of Hyperion Integration Server Desktop, follow these steps:

1. Start OLAP Integration Server.

   For instructions, see "Starting the Server on Windows Systems" on page 38 or "Starting the Server on UNIX Systems" on page 43.

2. Start Hyperion Integration Server Desktop.

   For instructions, see "Starting Hyperion Integration Server Desktop" on page 48.

3. In a blank Hyperion Integration Server Desktop window, select Connections > OLAP Metadata Catalog > Connect, and connect to the OLAP Metadata Catalog, TBC_MD.

4. In the **New** tab, double-click the **OLAP Model** icon.
5. In the **Data Source** dialog box, connect to the TBC sample database.

➤ To see the OLAP model (TBC Model) in the right frame of Hyperion Integration Server Desktop, follow these steps:

1. Start OLAP Integration Server.

   For instructions, see "Starting the Server on Windows Systems" on page 38 or "Starting the Server on UNIX Systems" on page 43.

2. Start Hyperion Integration Server Desktop.

   For instructions, see "Starting Hyperion Integration Server Desktop" on page 48.

3. If the **Login** dialog box is not already displayed, in a blank Hyperion Integration Server Desktop window, select Connections > OLAP Metadata Catalog > Connect.
4. Connect to the OLAP Metadata Catalog TBC_MD.
5. Click the **Existing** tab, select **TBC Model**, and click **Open**.
6. In the **Data Source** dialog box, connect to the TBC sample database.

➤ To see the metaoutline (TBC Metaoutline) in the right frame of Hyperion Integration Server Desktop, follow these steps:

1. Start OLAP Integration Server.

   For instructions, see "Starting the Server on Windows Systems" on page 38 or "Starting the Server on UNIX Systems" on page 43.

2. Start Hyperion Integration Server Desktop.

   For instructions, see "Starting Hyperion Integration Server Desktop" on page 48.

3. If the **Login** dialog box is not already displayed, in a blank Hyperion Integration Server Desktop window, select Connections > OLAP Metadata Catalog > Connect.

4. Connect to the OLAP Metadata Catalog TBC_MD.
5. Click the **Existing** tab, then click the plus symbol (+) to the left of TBC Model to expand the view, and select TBC Metaoutline.
6. Click **Open**.
7. In the **Data Source** dialog box, connect to the TBC sample database.

Chapter 7 Performing Advanced Installation Tasks

This chapter contains information about the changes made by the Hyperion Integration Server installation program, advanced installation procedures, and an installation troubleshooting checklist.

Note: The instructions for manually updating the UNIX environment in this chapter are mandatory for proper installation of Hyperion Integration Server on UNIX systems.

This chapter contains the following topics:

What Happens When You Install

When you install Hyperion Integration Server software, the installation program creates directories and copies program files, scripts, and documentation files to the hard drive. In addition, the installation program makes changes to the operating system environment, such as environment variable settings.

The specific changes made to the computer depend on the operating system (Windows or UNIX), whether you are installing server or client software, and which components of Hyperion Integration Server you choose to install.

Note: On UNIX systems, the installation program prepares shell scripts for setting environment variables and you must manually add these to the login scripts.

Server Installations on Windows Systems

When you install the Hyperion Integration Server server software on Windows NT and Windows 2000 computers, the installation program performs the following actions:

- Creates a new directory and several subdirectories on the hard drive

  By default, `setup` creates the following directory for the software:

  ```
  C:\hyperion\is\
  ```

  Note: During installation, you can create a different directory, but you cannot change the subdirectories.

  If you install client and server software on the same computer, you must install all of them in the same directory.

  For a listing of the subdirectories created and an explanation of their purpose, see Table 12.

- Copies program files, scripts, and documentation to the created directories on the hard drive
- Copies shared library files to the Windows system directory
- Updates the Windows operating environment:
  - Sets the ISHOME, ARBORPATH, and ISLOADINFO environment variables
  - Updates the PATH environment variable
  - Adds entries to the Windows Registry for Hyperion Integration Server

Table 12 lists the directory structure that the Hyperion Integration Server installation program creates for a server installation on Windows NT and Windows 2000:

Table 12: Directories Created for Server Installation on Windows NT and Windows 2000

Directory Structure	Description
\hyperion\is\	
batch	Empty. Later, the batch directory will store scheduler scripts for batch jobs you create.
bin	OLAP Integration Server and OLAP Command Interface software. If you also install client software, the Hyperion Integration Server Desktop software is in the Bin directory. A server log file, created when you first run OLAP Integration Server, is also in the bin directory.
books	PDF documentation—*Hyperion Integration Server Start Here*, *Hyperion Integration Server New Features, Hyperion Integration Server Desktop OLAP Model User's Guide*, *Hyperion Integration Server Desktop OLAP Metaoutline User's Guide*, and *Hyperion Integration Server System Administrator's Guide*.
esscript	Empty. Later, the esscript directory will contain calculation and ESSCMD scripts you create.
esslib	Three Hyperion Essbase subdirectories—esslib\bin, esslib\client, and esslib\locale.
esslib\bin	Hyperion Essbase DLL files.
esslib\client	Empty. Hyperion Essbase libraries use the client directory.
esslib\locale	Hyperion Essbase national language support files.
hyperlib	Hyperion Integration Server support DLL files.
loadinfo	Empty. Later, the loadinfo directory will contain folders representing session IDs, which will contain the reject files created during a member load or data load.
locale	Hyperion Integration Server national language support files.

Table 12: Directories Created for Server Installation on Windows NT and Windows 2000 (Continued)

Directory Structure	Description
ocscript	SQL script files to create, drop, and upgrade tables for an OLAP Metadata Catalog in each of the supported databases. For example, you can use ocdb2.sql to create OLAP Metadata Catalog tables in a DB2 database.
odbcdocs	MERANT (INTERSOLV) ODBC PDF documentation.
odbclib	MERANT (INTERSOLV) ODBC library files.
samples	Two subdirectories—samples\tbcdbase and samples\tbcmodel.
samples\tbcdbase	SQL script files, batch files, and text files to create tables and load data for the TBC sample application database.
samples\tbcmodel	SQL script files, batch files and text files to load data for the OLAP model (TBC Model) and metaoutline (TBC Metaoutline) in the TBC_MD sample application OLAP Metadata Catalog database. You create tables for TBC_MD with an SQL script file in the ocscript directory. For more information, see Chapter 6, “Setting up the Sample Application.”

Server Installations on UNIX Systems

When you install server software on AIX, HP-UX, and Solaris computers, the Hyperion Integration Server installation program performs the following actions:

- Creates a new directory and several subdirectories on the hard drive

  By default, `setup` creates the following directory for the software:

  ```
  home/hyperion/is
  ```

  Note: Before or during installation, you can create the main directory, but you cannot change the subdirectories.

  For a listing of the subdirectories that are created and an explanation of their purpose, see Table 13.

- Copies program files, scripts, and documentation to the created directories on the hard drive

- Copies shared library files to the `esslib` directory
- Creates `is.csh` and `is.sh` shell script files for updating the UNIX operating environment and installs them to the directory you created during installation
- Creates a sample `odbc.ini` in the `odbclib` directory for use in configuring relational data sources for ODBC

Table 13 lists the directory structure that the Hyperion Integration Server installation program creates for server software installations on AIX, HP-UX, and Solaris systems:

Table 13: Directories Created for Server Installation on AIX, HP-UX, and Solaris

Directory Structure	**Description**
`home/hyperion/is`	
`batch`	Empty. Later, the `batch` directory will store scheduler scripts for batch jobs you create.
`bin`	OLAP Integration Server and OLAP Command Interface software. A server log file, created when you first run OLAP Integration Server, is also in the `bin` directory.
`books`	`PDF` documentation—*Hyperion Integration Server Start Here*, *Hyperion Integration Server New Features*, *Hyperion Integration Server Desktop OLAP Model User's Guide*, *Hyperion Integration Server Desktop OLAP Metaoutline User's Guide*, and *Hyperion Integration Server System Administrator's Guide*.
`esscript`	Empty. Later, the `esscript` directory will contain calculation and ESSCMD scripts you create.
`esslib`	Three Hyperion Essbase subdirectories—`esslib/bin`, `esslib/client`, and `esslib/locale`.
`esslib/bin`	Hyperion Essbase library files.
`esslib/client`	Empty. Hyperion Essbase libraries use the `client` directory.
`esslib/locale`	Hyperion Essbase national language support files.
`loadinfo`	Empty. Later, the `loadinfo` directory will contain folders representing session IDs, which will contain the reject files created during member loads and data loads.

Table 13: Directories Created for Server Installation on AIX, HP-UX, and Solaris (Continued)

Directory Structure	Description
`locale`	Hyperion Integration Server national language support files.
`ocscript`	SQL script files to create, drop, and upgrade tables for an OLAP Metadata Catalog in each of the supported databases. For example, you can use `ocdb2.sql` to create OLAP Metadata Catalog tables in a DB2 database.
`odbcdocs`	MERANT (INTERSOLV) ODBC `PDF` documentation.
`odbclib`	MERANT (INTERSOLV) ODBC library files and sample `odbc.ini` file.
`samples`	Two subdirectories—`samples/tbcdbase` and `samples/tbcmodel`.
`samples/tbcdbase`	SQL script files, shell script files, and text files to create tables and load data for the TBC sample application database.
`samples/tbcmodel`	SQL script files, batch files and text files to load data for the OLAP model (TBC Model) and metaoutline (TBC Metaoutline) in the TBC_MD sample application OLAP Metadata Catalog database. You create tables for TBC_MD with an SQL script file in the `Ocscript` directory. For more information, see Chapter 6, “Setting up the Sample Application.”

Client Installations on Windows Systems

When you install the Hyperion Integration Server client software on Windows 95, Windows 98, Windows NT and Windows 2000 computers, the installation program performs the following actions:

- Creates a new directory and several subdirectories on the hard drive

  By default, `setup` creates the following directory for client software:

  ```
  C:\hyperion\is
  ```

  If you install client and server software on the same computer, you must install them all in the same directory.

  Note: During installation, you can choose to create a different directory, but you cannot change the subdirectories.

  For a listing of the subdirectories and their purpose, see Table 14.

- Copies program files, scripts, and documentation to the created directories on the hard drive
- Copies shared library files to the Windows system directory
- Updates the Windows operating environment:
  - Sets the ARBORPATH and ISHOME environment variables
  - Updates the PATH environment variable
  - Adds entries to the Windows Registry for Hyperion Integration Server

Table 14 lists the directory structure that the Hyperion Integration Server installation program creates for client installations on Windows systems:

Table 14: Directories Created for Client Installation on Windows 95, Windows 98, Windows NT, and Windows 2000

Directory Structure	Description
`\hyperion\is\`	
`bin`	Hyperion Integration Server Desktop and OLAP Command Interface software.
`books`	`PDF` documentation—*Hyperion Integration Server Start Here*, *Hyperion Integration Server New Features*, *Hyperion Integration Server Desktop OLAP Model User's Guide*, *Hyperion Integration Server Desktop OLAP Metaoutline User's Guide*, and *Hyperion Integration Server System Administrator's Guide*.
`esslib`	Three Hyperion Essbase subdirectories—`esslib\bin`, `esslib\Client`, and `esslib\locale`.
`esslib\bin`	Hyperion Essbase `DLL` files.
`esslib\Client`	Empty. Hyperion Essbase libraries use the `Client` directory.
`esslib\locale`	Hyperion Essbase national language support files.
`hyperlib`	Hyperion Integration Server support `DLL` files.
`locale`	Hyperion Integration Server national language support files.

If you install the sample application and the OLAP Metadata Catalog scripts, you will have more directories. For a complete list of directories, see Table 12.

Manually Updating the Environment on Windows Systems

During installation, you can decide to update operating system environment variables. If you decided not to update the environment variables, you must update them manually or the Hyperion Integration Server software will not work.

Table 15 lists the environment variable required to run the Hyperion Integration Server client or server software on Windows 95, Windows 98, Windows NT, or Windows 2000, or to run the server software on Windows NT, or Windows 2000:

Table 15: Environment Variables Required for Hyperion Integration Server on Windows 95, Windows 98, Windows NT, and Windows 2000

Name	Purpose and Default Setting
ISHOME	Specifies the directory where the Hyperion Integration Server software is installed. `C:\hyperion\is`
ARBORPATH	Specifies the directory where the Hyperion Essbase libraries are installed.[1] `C:\hyperion\is\esslib`
ISLOADINFO	Specifies a directory location for the reject files created during a member load or data load. `C:\hyperion\is\loadinfo`

PATH	Specifies the list of directories to search for software programs and libraries.
	The default setting is the current PATH list, with the following entries added to the end of the list:
	`C:\hyperion\is\bin;`
	`C:\hyperion\is\esslib\bin;`
	`C:\hyperion\is\hyperlib;`
	`C:\hyperion\is\odbclib;`[2]

Table 15: Environment Variables Required for Hyperion Integration Server on Windows 95, Windows 98, Windows NT, and Windows 2000 (Continued)

Name	Purpose and Default Setting
TEMP	Specifies a directory for temporary files. `C:\temp`

[1] If Hyperion Essbase is installed, set this variable to the directory where that product is installed. The default directory is `C:\hyperion\essbase`.

[2] The `odbclib` directory is required only if you installed the ODBC drivers provided with Hyperion Integration Server.

On Windows 95 and Windows 98, update the environment variables by editing the `autoexec.bat` file. On Windows NT and Windows 2000, update the environment variables using the System Properties dialog box from the Windows Control Panel.

If you choose to update the environment variables during installation, the Hyperion Integration Server installation program, `setup`, updates the environment variables. It detects whether Hyperion Essbase is installed on the same machine, and takes the appropriate action:

- The `setup` program always creates the ISHOME system variable and points it to the directory where Hyperion Integration Server is installed. By default, the value of ISHOME is:

  `C:\hyperion\is`

- If Hyperion Essbase is installed on the same machine, `setup` does nothing to the `ARBORPATH` variable, and Hyperion Integration Server shares the Hyperion Essbase `DLL` files in *ARBORPATH*`\bin`.

Updating Environment Variables on Windows 95 and Windows 98

On Windows 95 and Windows 98 systems, update environment variables by editing the `autoexec.bat` file.

➤ To manually update the environment on Windows 95 and Windows 98, complete the following steps:

1. Using a text editor, open the `autoexec.bat` file.
2. Add a new line to the file and use the SET command to create an environment variable. For example, to create the ISHOME variable and set it to the Hyperion Integration Server default installation directory, type:

   ```
   SET ISHOME=C:\hyperion\is
   ```

3. Repeat step 2 for each of the environment variables listed in Table 15.

   Tip: When updating the PATH variable, add directory entries to the end of the existing list by using the variable name itself; for example, add a new line and type:

   ```
   SET PATH=%PATH%;%ISHOME%\bin;%ISHOME%\esslib\bin;
   %ISHOME%\hyperlib;%ISHOME%\odbclib;
   ```

4. Save the `autoexec.bat` file and then close the text editor.

To implement the changes you made to the environment, restart the computer.

Updating Environment Variables on Windows NT

On Windows NT systems, update environment variables using the System Properties dialog box located in the Control Panel.

Note: You must have system administrator privileges to create a system variable. Otherwise, you can create only user variables.

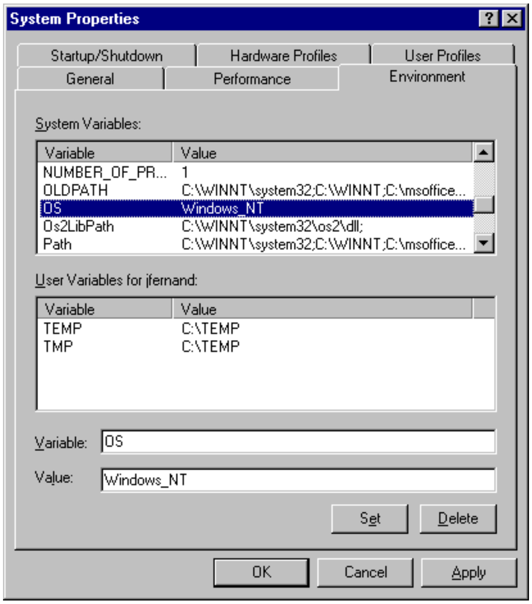
➤ To manually update environment variables on Windows NT, complete the following steps:

1. On the Windows desktop, click **Start**.
2. Select Settings > Control Panel.
3. In the **Control Panel** window, double-click **System**.

4. In the **System Properties** dialog box, select the **Environment** tab.

5. In the **System Variables** list box, select the name of any system variable. For example, select the OS system variable.

   The name of the system variable you selected is displayed in the **Variable** text box, and its value is displayed in the **Value** text box, as shown in Figure 10.



System Variables:

Variable	Value
NUMBER_OF_PR...	1
OLDPATH	C:\WINNT\system32;C:\WINNT;C:\msoffice...
OS	Windows_NT
Os2LibPath	C:\WINNT\system32\os2\dll;
Path	C:\WINNT\system32;C:\WINNT;C:\msoffice...

User Variables for jfernand:

Variable	Value
TEMP	C:\TEMP
TMP	C:\TEMP

Variable: OS

Value: Windows_NT

Figure 10: Selecting Any System Variable

6. In the **Variable** text box, in place of the name of the system variable you have selected, type the name of the variable you want to add; for example, **ISHOME**.

7. In the **Value** text box, in place of the value of the system variable you have selected, enter the value you want to add; for example, for the ISHOME variable, type **C:\hyperion\is** if you installed Hyperion Integration Server in the default destination directory.

8. Click **Set**.

   The variable and its value are displayed in the **System Variables** list box.

9. Repeat steps 5 through 8 for each of the environment variables listed in Table 15.

CAUTION: When updating the PATH variable, be sure to add the new directory entries to the end of the existing list. Do not delete existing directory entries from the list.

10. Click **Apply** and click **OK** to close the **System Properties** dialog box.

To implement the changes you made to the environment, log off Windows and log on again or restart the computer.

Updating Environment Variables on Windows 2000

On Windows 2000 systems, update environment variables using the System Properties dialog box, located in the Control Panel.

Note: You must have system administrator privileges to create a system variable. Otherwise, you can only create a user variable.

➤ To manually update environment variables on Windows 2000, complete the following steps:

1. On the Windows desktop, click **Start**.
2. Select Settings > Control Panel.
3. In the **Control Panel** window, double-click **System**.
4. In the **System Properties** dialog box, select the **Advanced** tab.
5. In the **Advanced** tab, click **Environment Variables**.

6. In the **Environment Variables** list box, perform one of the following steps:
   - To create a new variable, in the **Systems variables** group box, click **New**.
   - To edit a variable, in the **Systems variables** list box, select the variable you want to change, and click **Edit**.
7. In the **Variable Name** text box, type the name of the variable you want to add; for example, **ISHOME**.
8. In the **Variable Value** text box, enter the value you want to add; for example, for the ISHOME variable, type **C:\hyperion\is** if you installed Hyperion Integration Server in the default destination directory.
9. Click **OK**.

   The variable and its value are displayed in the **System Variables** list box.
10. Repeat steps 6 through 9 for each of the environment variables listed in Table 15.

CAUTION: When updating the PATH variable, be sure to add the new directory entries to the end of the existing list. Do not delete existing directory entries from the list.

11. Click **OK** to close the **System Properties** dialog box.

Before you use OLAP Integration Server, either log off Windows and log on again or restart the computer to apply the changes you made to the environment.

Starting the Schedule Service on Windows NT and Windows 2000

Hyperion Integration Server uses the Windows NT Schedule service and the Windows 2000 Task Scheduler service to perform scheduled member and data loads. To allow loads to be scheduled, you must start one of these services on the computer that runs OLAP Integration Server.

➤ To start the Windows NT Schedule service:

1. On the Windows desktop, click **Start**.
2. Select Settings > Control Panel.
3. In the **Control Panel** window, double-click the **Services** icon.
4. In the list of services, select **Schedule**.
5. In the **Services** dialog box, click the **Start** button, then click the **Startup** button.
6. In the **Services** dialog box, under **Startup Type**, select **Automatic**.

➤ To start the Windows 2000 Task Scheduler service:

1. On the Windows desktop, click **Start**.
2. Select Settings > Control Panel.
3. In the **Control Panel** window, double-click the **Administrative Tools** icon, and then double-click the **Services** icon.
4. In the list of services, double-click **Task Scheduler**.
5. In the **Task Scheduler Properties** dialog box, select the **General** tab.
6. Under **Startup type**, select **Automatic**.
7. To start the service, in the **Task Scheduler Properties** dialog box, click **Start**.

Manually Updating the Environment on UNIX Systems

After you install Hyperion Integration Server on UNIX systems, you must perform some manual configuration before you can run OLAP Integration Server or OLAP Command Interface. These manual configurations include the following steps:

- Set environment variables for Hyperion Integration Server.

  For more information, see "Updating the Environment for Hyperion Integration Server" on page 105.

- Set the environment variables for ODBC.

  For more information, see "Updating the Environment for ODBC" on page 106.

- Set the environment variables for the database client.

  For more information, see "Updating the Environment for the Database Client" on page 107.

- Authorize the user who starts OLAP Integration Server to run cron jobs (required only if you using the Hyperion Integration Server scheduler).

  For more information, see "Configuring the Cron Scheduling Daemon" on page 107.

Set the environment variables using shell scripts. Sample scripts for setting Hyperion Integration Server and ODBC environment variables are provided when you install Hyperion Integration Server. The database client should include a similar sample script for setting the environment variables it requires. For more information, see the installation documentation for the database client.

The shell scripts should be invoked in the login script for the user (typically, hyperion) who runs Hyperion Integration Server.

Tip: Set up the environment variables for Hyperion Integration Server and its supporting software under a specific user, rather than setting some environment variables at the system level. This approach allows you to set up and run different versions of the product independently of each other. In a high-availability environment, this allows you to set up a new version of Hyperion Integration Server while an older version is still running and then switch over to the new version when it is ready to be put into production.

Updating the Environment for Hyperion Integration Server

The Hyperion Integration Server installation program, `setup.sh`, creates and installs two shell scripts, `is.csh` and `is.sh`, for updating environment variables. Both shell scripts set the environment variables listed in Table 16:

Table 16: Environment Variables Required for Hyperion Integration Server on AIX, HP-UX, and Solaris

Name	Purpose and Default Setting
ISHOME	Specifies the directory where the Hyperion Integration Server software is installed. `/home/hyperion/is`
ARBORPATH	Specifies the directory where the Hyperion Essbase libraries are installed.[1] `$ISHOME`
ISLOADINFO	Specifies a directory location for the reject files created during a member load or data load. `$ISHOME/loadinfo`
PATH	Specifies the list of directories to search for software programs and libraries. The default setting is the current PATH list, with the following entry added to the end of the list: `$ISHOME/bin:`

Library path (Platform):
LIBPATH (AIX)
SHLIB_PATH (HP-UX)
LD_LIBRARY_PATH (Solaris)
	Specifies the list of directories to search for software libraries.
	The default setting is the current library path list, with the following entries added to the end of the list:
	`$ISHOME/bin;`
	`$ISHOME/esslib`
	`$ISHOME/odbclib;`

TEMP	Specifies a directory for temporary files. `/tmp`

[1] If Hyperion Essbase is installed, set this variable to the directory where that product is installed. The default directory is `/home/hyperion/essbase`.

Update the UNIX environment by adding `is.csh` or `is.sh` to the login script of the user who runs the Hyperion Integration Server software.

➤ To include the script in your login scripts, perform the following steps:

1. In a text editor, open the login script file for the user (typically, hyperion) who runs Hyperion Integration Server.
2. Add a new line to the file.
   - If you use Korn shell, type:

     `. /`*`full_path_to_directory`*`/is.sh`

   - If you use C shell, type:

     `source /`*`full_path_to_directory`*`/is.csh`

3. Save the login script file and close the text editor.

Updating the Environment for ODBC

The `is.sh` and `is.csh` scripts provided with Hyperion Integration Server set the library path to access the MERANT (formerly INTERSOLV) ODBC drivers provided with the product. If you use these drivers, include `is.sh` or `is.csh` in the login script for the user who runs Hyperion Integration Server to set the required variables.

Note: If you use ODBC drivers other than MERANT, the provider typically provides a shell script to set the library path to access the driver. Add this shell script to the login script for the user who runs the Hyperion Integration Server software.

An example of a shell script that sets the library path to access ODBC libraries on Solaris follows:

```
if env | grep LD_LIBRARY_PATH" > /dev/null
  then LD_LIBRARY_PATH=
  $ISHOME/esslib:$ISHOME/odbclib:$ISHOME/islib:$LD_LIBRARY PATH

else
  LD_LIBRARY_PATH=$ISHOME/esslib:$ISHOME/odbclib:$ISHOME/islib
```

See the ODBC driver documentation for details on how to set the library path to access the ODBC libraries.

Updating the environment for ODBC requires two additional steps:

- Configuring the `odbc.ini` file (for more information, see "Editing the odbc.ini File" on page 69)
- Configuring the environment to use the database client software (for more information, see "Updating the Environment for the Database Client" on page 107)

Updating the Environment for the Database Client

You must set the environment variables required for database client software in the login script of the user who runs Hyperion Integration Server. These environment settings are required for ODBC access to the databases in the RDBMS you are using.

The database vendor typically supplies a shell script to set any environment variables required by the database client. Add this shell script to the login script for the user who runs the Hyperion Integration Server software.

For more information on how to set environment variables for the database client, see the database client installation documentation.

Note: To verify that the database client software is set up correctly, you should log on as the user who runs OLAP Integration Server and use one of the database utilities to connect to the databases that you use with Hyperion Integration Server.

Configuring the Cron Scheduling Daemon

Hyperion Integration Server uses the cron scheduling daemon to perform scheduled member and data loads. This daemon must be started on the computer that runs OLAP Integration Server and the user (typically, hyperion) who starts the OLAP Integration Server program must be authorized to schedule cron jobs. For more information, see the documentation for the UNIX system you are using.

Creating Database User Aliases and Synonyms

When using the sample application, you must connect to TBC_MD as user TBC unless you set up your user name as an alias or synonym for TBC.

- On SQL Server, make your login user name an alias of TBC.
- On DB2 and Oracle, log on with your user name and password, and create synonyms that map to the tables in TBC_MD.

This user alias or table synonym enables you to access a table that is qualified by TBC because it was created by the TBC user; for example, TBC.MO_INFO in the TBC_MD OLAP Metadata Catalog.

For more information, see the documentation for the RDBMS you are using.

Removing Software from Windows Systems

Hyperion Integration Server software is registered with Windows systems so that it can be uninstalled.

➤ To remove the software, complete the following steps:

1. Click **Start** on the Windows desktop, and select Programs > Settings > Control Panel.
2. In the **Control Panel** window, double-click **Add/Remove Programs**.
3. In the **Add/Remove Programs Properties** dialog box, select Hyperion Integration Server and click **Add/Remove**.

4. Confirm that you want to remove the Hyperion Integration Server software.

Note: On Windows NT and Windows 2000, removing Hyperion Integration Server removes the ISHOME system variable, but does not remove the following items:

- The `ARBORPATH` system variable
- Hyperion Integration Server directory entries in the `Path` system variable
- Files created after the installation, such as `olapisvr.log`

On Windows 95 and Windows 98, removing Hyperion Integration Server does not change the `autoexec.bat` file.

Removing Software from UNIX Systems

To remove Hyperion Integration Server server software on AIX, HP-UX, or Solaris, type the following command:

```
essuninst.sh
```

and indicate which server components you want to remove. The `essuninst.sh` file is in the *ISHOME* directory.

Troubleshooting Installations

If you encounter problems after installing Hyperion Integration Server, check the troubleshooting list in Table 17 for solutions:

Table 17: Installation Troubleshooting List

Problem	Solution
Server will not start	Check the environment variable settings.
Client will not start	Check the environment variable settings.
Long wait or no response from server	Check to see if OLAP Integration Server or server computer is running.

Table 17: Installation Troubleshooting List (Continued)

Problem	Solution
Cannot log on to OLAP Metadata Catalog	• Passwords are often case-sensitive; make sure the Caps Lock key is disabled. • Make sure you are using the correct user name and password. Use the same user name and password that was used to create the catalog, or an alias of that user name. • Check the environment variable settings for the database client software on the server. • Check the environment variable and connection settings for ODBC on the server. • Check the ODBC data source configuration for the catalog on the server.
Cannot log on to data source	See solutions in this table for logging on to OLAP Metadata Catalog.
Cannot see a catalog in SQL Server	If you receive a message that OM_INFO is missing, be sure to configure the ODBC Data Source Name (DSN) to use the catalog database. Also, run `ivtestlib` to verify that the environment is set to run the correct odbc driver file. For example, run `ivtestlib` and paste the path and file name that you defined for `Driver=` in the `odbc.ini` file.

Most problems with Hyperion Integration Server are the result of connection difficulties and improperly set environment variables—especially on UNIX systems. For more information on diagnosing connection problems, see the *Hyperion Integration Server System Administrator's Guide*. For information on setting environment variables, see "Manually Updating the Environment on Windows Systems" on page 97 or "Manually Updating the Environment on UNIX Systems" on page 104.

The SQL Server ODBC driver may time out during a call to an SQL Server database. Try again when the database is not busy. Increasing the driver time-out period may avoid this problem. For more information, see the ODBC documentation.

CAUTION: If you install Hyperion Integration Server on the same machine where Hyperion Essbase is installed, and if you later remove Hyperion Essbase, you must change the environment to adjust for the removal of Hyperion Essbase. Otherwise, Hyperion Integration Server will not work. For information on the operating environment and how to change it, see "Manually Updating the Environment on Windows Systems" on page 97.

Index