

IBM[®] DB2[®] OLAP Server[™]



Installation Guide

Version 8.1

IBM[®] DB2[®] OLAP Server[™]



Installation Guide

Version 8.1

Note

Before using this information and the product it supports, be sure to read the information in “Notices” on page 147.

First Draft (April 2002)

This edition applies to version 8.1 of DB2 OLAP Server (program number 5724-OLP) and to all subsequent releases and modifications until otherwise indicated in new editions.

Portions of this manual © Copyright Hyperion Solutions Corporation 1991, 2002

© Copyright International Business Machines Corporation 1998, 2002. All rights reserved.

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

Contents

About this book	vii	Installing DB2 OLAP Server Miner on Windows	49
Who should read this book	vii	Setting up client installation from a network file server	50
Publications	vii	Loading the client image onto the network	51
Additional support	ix	Installing the clients from the network	53
Chapter 1. What's new in Version 8.1	1	Maintaining the common clients	54
What's new in DB2 OLAP Server 8.1	1	Configuring Client and Server for Non-English DB2 OLAP Server	55
What's new in DB2 OLAP Integration Server 8.1	11	Starting DB2 OLAP Server on Windows.	56
Chapter 2. Before you install.	25	Automating server startup	57
Read the installation checklist	25	Automating application startup	57
Prepare to upgrade DB2 OLAP Integration Server	26	Stopping DB2 OLAP Server	57
Preserving an existing ODBC Driver	27	Optimizing network throughput on Windows NT	58
Migrating existing OLAP Metadata Catalogs	27	Adding additional software or users later	58
Check compatibility with earlier versions	28	Using agent commands	59
Chapter 3. Hardware and software Requirements	31	Chapter 5. Installing on AIX, Solaris Operating Environment, and HP-UX	61
Server hardware and software requirements	31	Installing DB2 OLAP Server on UNIX	62
Windows server requirements	32	Installation steps for DB2 OLAP Server	62
Solaris Operating System server requirements	33	Installing DB2 OLAP Integration Server on UNIX	64
AIX server requirements	34	Updating the Environment for the Database Client	64
HP-UX server requirements	36	Installing DB2 OLAP Server Miner on AIX.	65
Client hardware and software requirements	37	Updating the environment settings	66
API client hardware and software requirements	39	For all UNIX operating systems	66
Supported ODBC drivers and relational data sources	39	For Solaris Operating Environment only	67
For DB2 OLAP Server.	39	For HP-UX only.	67
For DB2 OLAP Integration Server.	41	Starting and shutting down the server	68
Supported JDBC Drivers for XML Import/Export in OIS.	42	Starting the server in the foreground.	68
Chapter 4. Installing on Windows	45	Starting the server in the background	69
Installing DB2 OLAP Server.	45	Securing a password script	69
Installation steps	46	Shutting down the server	70
Security	47	Adding additional software or users later	70
Network Protocol Files	47	Using agent commands	70
Installing DB2 OLAP Integration Server on Windows	48	Chapter 6. Installing and configuring Java 73	
		Setting up Java on UNIX.	73
		Installing Java on UNIX	74
		Updating Java variables for DB2 OLAP Server on UNIX.	75

Giving DB2 OLAP Server access to JAXP on UNIX	78	Creating an OLAP Metadata catalog manually	108
Updating Java variables for DB2 OLAP Integration Server on UNIX	79	Upgrading the OLAP Metadata Catalog	110
Finding which shell you are using	81		
Setting up Java on Windows	82	Chapter 10. Setting up the sample applications	111
Setting Java environment variables on Windows	82	Samples for DB2 OLAP Server	111
Setting JVMMODULELOCATION on Windows	83	Loading the sample databases for DB2 OLAP Server	111
Chapter 7. Configuring the SQL Interface	85	Providing user access to the sample applications	112
Configuring SQL Interface on Windows	85	Setting the environment for the partitioning sample applications	113
Configuring SQL Interface on UNIX	86	Loading Data into Samppart and Sampeast.	116
Workflow for configuring SQL Interface on UNIX	86	Samples for DB2 OLAP Integration Server	117
Setting up RDBMS for the SQL Interface on UNIX	87	About the DB2 OLAP Integration Server samples	118
Directories and files created	88	Before you set up the sample application	118
Setting up ODBC drivers before linking them to SQL Interface	88	Setting up the sample application automatically	119
Linking SQL Interface to the ODBC driver on UNIX	90	Setting up the sample application manually	120
After you link SQL Interface to your ODBC Driver on UNIX	91	After you set up the sample application	129
Configuring the data source and ODBC driver	94		
Chapter 8. Configuring data sources for DB2 OLAP Integration Server	97	Chapter 11. Upgrading and migrating from earlier versions	131
About configuring relational data sources	97	Migrating DB2 OLAP Server	131
Configuring the database client software Supported ODBC drivers	98	Migrate from RSM to MSM before upgrading to Version 8.1	131
Configuring data sources on Windows systems	98	Understanding Input/Output defaults and upgrading.	132
Configuring the OLAP metadata catalog on Windows systems.	100	Considerations when upgrading from previous releases	135
Configuring data sources on UNIX systems	101	Upgrading databases to Version 8.1.	136
Editing the odbc.ini file	101	Migrating applications and databases across servers	138
Configuring the OLAP metadata catalog on UNIX systems	104	Migrating DB2 OLAP Integration Server	141
		ODBC drivers on Windows	141
		Java Runtime Environment	141
		Migrating existing OLAP metadata catalogs	142
		Security Migration Tool	143
		Main features of the Security Migration tool	143
		Supported platforms.	143
		Running the Security Migration tool	143
		Other notes on migration	145
Chapter 9. Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server	105		
About OLAP metadata catalogs	105		
Creating an OLAP Metadata Catalog	106		
Creating an OLAP Metadata catalog automatically	106		
Deleting an OLAP Metadata Catalog	107		
		Notices	147

Trademarks 148

Glossary 151

Index 155

Contacting IBM 157

Product Information 157

About this book

DB2 OLAP Server is based on products from Hyperion Solutions Corporation. This book was written by IBM and is intended to be used with books and online help written by Hyperion. You will see references to Hyperion and Hyperion's products in the interface and throughout the Hyperion books. Only the relevant Hyperion books are included with the DB2 OLAP Server book; this book and the exceptions are listed in "Publications".

This book is for DB2 OLAP Server system administrators who need to install and configure OLAP Server software on Windows or UNIX platforms. It is also intended to assist OLAP system administrators in setting up sample applications, working with network protocols, and helping client users install client software.

Who should read this book

Read this book if you are a DB2 OLAP Server administrator who is responsible for:

- Installing and configuring DB2 OLAP Server.
- Designing and creating OLAP applications and multidimensional databases using DB2 OLAP Server.
- Setting up security for DB2 OLAP Server.
- Maintaining OLAP applications and multidimensional databases that are migrated to or created using DB2 OLAP Server.

As the DB2 OLAP Server administrator, you should have experience in networking and system administration, and should understand the analytical requirements of the users who will use the applications created with DB2 OLAP Server.

Publications

This section lists books that are included with DB2 OLAP Server and its add-on products. DB2 OLAP Server users can find these books from the **Start > Programs > IBM DB2 OLAP Server 8.1 > Documentation** information map. The books are stored in the `x:\ibm\db2olap\docs` directory, where `x:\ibm\db2olap` is the directory in which you installed DB2 OLAP Server workstation applications.

If you choose the **Typical** option during installation, DB2 OLAP Server installs only a subset of the documentation. To install all the documentation, choose **Custom** during installation and select the books you want.

Table 1 lists the publications that are included with the DB2 OLAP Server base product.

Table 1. DB2 OLAP Server Publications

Publication Name	Description
<i>Installation Guide</i> (SC27-1228)	Explains how to install and configure DB2 OLAP Server and its features.
<i>Quick Path</i> (SC18-7000)	Summarizes the major tasks of creating OLAP applications using DB2 OLAP Server.
<i>OLAP Miner User's Guide</i> (SC27-1611)	Explains how to use DB2 OLAP Server Miner, a no-charge feature that combines the powerful IBM data mining technologies with OLAP technologies.
<i>Spreadsheet Add-in User's Guide for Excel</i> (SC27-1232)	Explains how to use the Spreadsheet Add-in with Microsoft Excel for Windows to analyze data.
<i>Spreadsheet Add-in User's Guide for 1-2-3</i> (SC27-1231)	Explains how to use the Spreadsheet Add-in with Lotus 1-2-3 for Windows to analyze data.
<i>Database Administrator's Guide, Volumes 1, 2, and 3</i> (SC18-7001, SC18-7002, and SC18-7003, respectively)	Provides administrators with strategies and techniques to implement, design, build, and maintain an optimized multidimensional database. Provides instructions for how to design and build a security system. Provides instructions for how to load, calculate, and report your data. Also provides instructions for performing tasks using the Application Manager.
<i>MaxL User's Guide</i> (SC18-7005)	Describes how to use MaxL, the multi-dimensional database access language for DB2 OLAP Server. MaxL is a flexible way to automate Hyperion Essbase administration and maintenance tasks.
<i>SQL Interface Guide</i> (SC18-7004)	Explains how to load data from SQL, relational, and flat-file data sources. The SQL Interface tool is part of the Tools Bundle, which is an add-on product for DB2 OLAP Server.
<i>Technical Reference</i>	Provides reference about writing formulas, calculation scripts, report scripts, MaxL statements, and custom-defined macros and function. It also includes ESSBASE.CFG settings and ESSCMD commands.

Table 1. DB2 OLAP Server Publications (continued)

Publication Name	Description
<i>API Reference</i>	Provides reference information for functions that are available through the Application Programming Interface, which is part of the DB2 OLAP Server Tools Bundle.
<i>Message Reference</i>	Lists the most commonly encounter messages in DB2 OLAP Server and their most likely causes and solutions.
<i>Integration Server Administration Guide (SC27-1227)</i>	Explains how to use the Administration Manager component of the DB2 OLAP Integration Server.
<i>Integration Server Data Preparation Guide (SC18-7006)</i>	Explains how to prepare relational data for use with the DB2 OLAP Integration Server.
<i>MERANT DataDirect Connect ODBC Reference</i>	Provides reference information for the MERANT (INTERSOLV) ODBC drivers.

The DB2 OLAP Server publications are also available online at:

www.ibm.com/software/data/db2/db2olap/library.html

Several IBM Redbooks about DB2 OLAP Server and related topics are available. To view OLAP Redbooks online, visit the IBM Redbook web site at:

www.redbooks.ibm.com

The following Hyperion Essbase and Essbase Integration Services books are not included with this product. The information in this book supersedes the information in following books:

- *Hyperion Essbase New Features*
- *Hyperion Essbase Installation Guide*
- *Hyperion Essbase Installation Checklist*
- *Essbase Integration Services New Features*
- *Essbase Integration Services Installation Guide*

Additional support

The Web site for DB2 OLAP Server and its features now offers an extensive selection of hints, tips, frequently asked questions, and more information, as well as FixPaks::

www.ibm.com/software/data/db2/db2olap/support.html

Chapter 1. What's new in Version 8.1

This chapter describes the enhancements made to DB2 OLAP Server and DB2 OLAP Integration Server since the previous version.

What's new in DB2 OLAP Server 8.1

This section provides an overview of significant changes in DB2 OLAP Server Version 8.1.

OLAP Miner for Deviation Discovery

DB2 OLAP Server Miner is a no charge feature of DB2 OLAP Server. OLAP Miner combines the powerful IBM data mining technologies with OLAP technologies.

OLAP Miner enables you to examine your OLAP cubes (multidimensional databases) more thoroughly and easily by finding atypical or deviant values automatically. Instead of searching through your data manually to find data values that might be higher or lower than you expected, OLAP Miner can do this work for you. The OLAP Miner algorithm is optimized so that it can be deployed on large OLAP cubes.

You can search specific areas (subcubes) of your cubes by creating deviation detection definitions, which specify the way in which OLAP Miner examines the data that you selected for the subcube. OLAP Miner provides an easy-to-use wizard that helps you create deviation detection definitions. After you create and run a deviation detection definition, OLAP Miner applies an algorithm to the data to find deviations. Then you can view a simple list of deviations or bring up the Deviation Viewer, from which you can see the deviations in the context with other surrounding values. You can also view deviations in a standard spreadsheet application.

OLAP Miner was introduced in FixPak 7 of Version 7.1.

Application Manager replaces Administration Manager

The Administration Manager utility is not available in Version 8.1. All the function that was available previously in the Administration Manager is available in the Application Manager.

Relational Storage Manager is no longer available

Starting in Version 8.1, the Relational Storage Manager (RSM) is withdrawn from DB2 OLAP Server. Applications created using the RSM in previous versions must be migrated to the Multidimensional

Storage Manager (MSM) before installing Version 8.1. See “Migrate from RSM to MSM before upgrading to Version 8.1” on page 131 for more information.

DB2 OLAP Server is no longer enabled for Tivoli

Tivoli enablement is withdrawn in Version 8.1.

The following changes are specific to Release 6.5 of Hyperion Essbase OLAP Server.

Hybrid Analysis for relational data support

DB2 OLAP Server supports Hybrid Analysis, a method of accessing the lowest level members of an OLAP database from a relational database. This relationally stored data is not physically loaded into DB2 OLAP Server. The data is mapped to the appropriate OLAP hierarchies.

Use OIS and the Hybrid Analysis manager to create applications that access relational data using Hybrid Analysis. The manager generates queries dynamically against the relational source, and runs as part of the OLAP Server.

For more information about how to use Hybrid Analysis, see these documents:

- *Technical Reference* in the docs directory, in the Report Writer and Configuration Settings sections for information about Hybrid Analysis commands and settings
- OIS online help and Intelligent Help
- Application Manager help: Database Settings, Storage Tab, Access Mode

Parallel Calculation

In addition to the standard means of performing a calculation, where each task is performed in sequence or serially, DB2 OLAP Server now supports parallel calculation.

If you enable parallel calculation, DB2 OLAP Server analyzes all the tasks in a calculation pass and breaks them down into subtasks. The subtasks that can run independently of each other are scheduled to run simultaneously on up to four threads, each on a separate CPU.

See the Technical Reference, which entries in the configuration setting and calculation commands sections. You can enable and adjust parallel calculation by adding `CALCPARALLEL` (and optionally `CALCTASKDIMS`) to your `essbase.cfg` configuration file, or you can issue the calculation commands `SETCALCPARALLEL` (and optionally `SET CALCTASKDIMS`).

Simulated Calculations

Use the ESSCMD command SET MSG ONLY to perform simulated calculations. These simulated calculations can help you to select dense and sparse settings for the dimensions in your outline. See the *Technical Reference* in the docs directory for more information.

Empty Outline Support in Dimension Building

DB2 OLAP Server can now build dimensions in empty outline files. When a new database is created, DB2 OLAP Server creates an empty outline file. In previous releases, before building outlines from a data source, users had to add at least one dimension member to the empty outline manually. Now, DB2 OLAP Server no longer requires the manual step.

Substitution Variable Support in SQL Interface

You can use substitution variables in SQL strings, similar to substitution variable use in report scripts and calculation scripts. For details, see the *SQL Interface Guide*, available in PDF or HTML.

Connection Information Available

Client API programs can now access information about users currently connected to Essbase. The new function EssListLogins() provides a count of the users logged in to a specified connection and returns information on the connected client names and IP addresses.

Server Log Enhancements

Messages in the server log dealing with user logins and security changes now include additional details:

- The user name of anyone entering an incorrect password or changing a password
- The IP address of the computer on which a user logs in
- The user name and group name when the user or group is deleted

In addition, DB2 OLAP Server logs additional security activities in the server log, including the name of the user and group being changed and the user performing the change:

- Adding users to groups and removing users from groups
- Granting and revoking database and application access

I/O Setting Changes

Previously, configuration setting DIRECTIO controlled the default mode of I/O access at the server level. Now, DIRECTIO controls only the I/O mode used for databases when they are created. To change the I/O mode for an individual database, use Application Manager, and navigate to the Database Settings window, Storage tab.

Correct Structure Alignment

Header files contain new information and structures that enable DB2

OLAP Server to run more quickly and efficiently. You need not do anything for new or upgraded installations, the changes are made automatically. Two header files, esbapi.h and esbotl.h, are no longer needed, and therefore no longer shipped with Essbase. One new header file, esstsa.h, has been added.

SQL Interface Guide in HTML and PDF

The Essbase SQL Interface Guide is now available in HTML as well as PDF.

New Error Message Documentation

Help for error messages are now available from the Information Map, which you can launch from the **Start > Programs > IBM DB2 OLAP Server 8.1 > Documentation** menu.

The following changes are specific to Release 6.2 of Hyperion Essbase OLAP Server.

Application and database migration across OLAP Servers

You can migrate applications and databases from one server to another, regardless of platform. When you migrate, you can also migrate user/group security, including filter associations.

User and group migration across OLAP Servers

You can migrate DB2 OLAP Server users and groups from one server to another, regardless of platform. You can migrate individual users and groups, or you can migrate multiple users or multiple groups at one time.

Password propagation

You can change a password and then propagate that password to other OLAP Servers.

Single sign-on to DB2 OLAP Server enterprise environment

When you log in to Administration Services, the Administration Server handles your connections to OLAP Servers, applications, and databases.

Log analysis

Using Log Analyzer, you can filter, search, and analyze OLAP Server logs and application logs.

New MaxL Script Editor

A new MaxL Script Editor is integrated with the Administration Services Console. From this editor, you can create, edit, save, and execute MaxL statements and scripts. You can also use the editor to type and execute MaxL statements interactively, or to perform one DB2 OLAP Server operation at a time. Results from MaxL statements are displayed within the editor.

Background processing

You can execute calculations in the background while performing other tasks. You can also exit the console while the process continues to run. A message panel in the console window notifies you when the process has completed.

Multiple platform support

Both the middle tier Administration Server and the client console are Java products that run on multiple platforms. You can now manage DB2 OLAP Server with a user interface that runs on any platform supported by DB2 OLAP Server.

Printing of DB2 OLAP Server information

You can print the contents of many windows in the console. For example, you can print the contents of the Database Properties window or the Log Analyzer window.

Graphical view of DB2 OLAP Server

You can use the Graphical Enterprise view in the console to operate directly on DB2 OLAP Server objects and to create custom views of the DB2 OLAP Server environment.

Parallel data load

DB2 OLAP Server now supports parallel data loading. You can specify the number of data processing threads to be used in parallel in different stages of data load processing. Allocating processing threads across processors on multiple CPUs provides the performance advantages of true parallel operation.

Limits on query duration and time

When users create and execute a report or spreadsheet add-in query, they may unintentionally create a very large query. Some large queries may require too much time to execute or may access too many blocks to run efficiently.

To help administrators, DB2 OLAP Server now provides these new features:

- Administrators can limit the amount of time a query can run before completion.
- Administrators can limit the number of blocks that a query can access.
- Administrators can set these new limits at the server, application, or database level.

This new ability is available using new `essbase.cfgfile` settings.

New control of agent-server threads and user logins

Two new configuration settings provide more control over agent activity:

- AGTSVRCONNECTIONS specifies the number of server threads that the OLAP Server creates during startup to connect to the agent. This subset of server threads, along with the agent threads controlled by AGENTTHREADS, are needed when users connect from an agent to the server to in order to connect to an application and database. These threads are unrelated to the threads used direct client queries, which are specified by SERVERTHREADS or have default specification based on the number of licenses purchased.
- MAXLOGIN specifies the maximum number of users that can simultaneously log in to a single application at a time. This number includes all instances of the same user on the same computer and application. Both the server and agent use this setting. MAXLOGIN is unrelated to threads controlled by AGENTTHREADS, AGTSVRCONNECTIONS, or SERVERTHREADS.

Use these settings to match resources to load patterns:

- To support a small number of users, you can lessen the amount of memory that the OLAP Server uses by decreasing the value of MAXLOGIN.
- To support a large number of users trying to access the same application at the same time, you can increase the value of AGTSVRCONNECTIONS and AGENTTHREADS.
- To support many users issuing queries at the same time, you can increase the value of SERVERTHREADS.

Log delimiters

If you need to use external reporting tools or import log data into an OLAP database, DB2 OLAP Server now provides the ability to specify a delimiter between fields in DB2 OLAP Server log files:

- The `essbase.cfg` configuration file now supports an optional setting `DELIMITEDMSG` and `DELIMITER`. When `DELIMITEDMSG` is set to `TRUE`, DB2 OLAP Server inserts a tilde (~) between each field of a message, and writes each entry on a single line. If `DELIMITER` is set to one of three symbols, that symbol is used instead of the tilde.
- Log files contain standardized fields and field ordering. For example, an entry with a date and time stamp uses the stamp as the first field in the message.

New calculation functions

DB2 OLAP Server provides three new calculation functions in this release:

- Use `@PRIORS` and `@NEXTS` to skip `#MISSING` values, zero (0) values, or both, or neither during a calculation. `@PRIOR` and `@NEXT`, which do not skip, are still available.

- Use @NAME to pass a member name, as a string, to another function.
- Use @ALIAS to pass a member alias, as a string, to another function.

See the *Technical Reference* in the docs directory of your DB2 OLAP Server installation for details.

New API functions

DB2 OLAP Server Version 8.1 has two new API functions. One is a list of native and custom-defined calculation functions, and the other a new function to create a filter. Both functions are documented in this guide.

External authentication

DB2 OLAP Server now supports external authentication of users against a non-DB2 OLAP server. For this release, DB2 OLAP Server installs a protocol that can authenticate users against an LDAP V3 server.

To enable external authentication, administrators use this procedure:

1. Set the AUTHENTICATIONMODULE configuration setting in the server and client configuration file `essbase.cfg`.
2. Restart OLAP Server.
3. Create the user in DB2 OLAP Server with a reference to the LDAP protocol, using either Application Manager or DB2 OLAP Server Administration Services.

For instructions about how to set the configuration parameter, see the *Technical Reference* in the docs directory of your DB2 OLAP Server installation.

Export/Import LROs

The ability to export and re-import linked reporting object (LRO) information from a database has been added to improve backup capabilities and calculation performance.

You can use export LRO and import LRO(MaxL) to perform these tasks:

- Export the catalog that maps LROs to data cells, from a database to a single file on the server
- Load a file containing the exported LRO information.

Export of databases larger than 2 GB

Some file management systems do not support ASCII files larger than 2 GB. To avoid exceeding this limit, if DB2 OLAP Server anticipates that an export file will be larger than 2 GB, it breaks the file into two or more export files, as needed. Use the ESSCMD commands,

EXPORT and PAREXPORT, and the MaxL statement, export to export databases. This new feature works the same on all operating systems.

New CALCMODE configuration setting

You can now set CALCMODE to BLOCK or BOTTOMUP using the configuration setting CALCMODE in the server configuration file `essbase.cfg`, and give it server, application, or database scope instead of using the @CALCMODE function. For more information, see the configuration settings in *Technical Reference* in the docs directory of your DB2 OLAP Server installation.

Improved functionality

DB2 OLAP Server Version 8.1 provides these improvements to existing functionality:

- Improved Handling of Sessions and Requests
- Multiple DB2 OLAP Server Agents on a Single Server Computer
- Case Sensitivity Improvement
- Improved Add-in Support of Excel in Browsers
- Member Set Function Support in FIX
- New Dimensions Sparse by Default
- Custom-Defined Functions and Macros
- Attribute Association Memory Use
- Buffered I/O Default

Improved handling of sessions and requests

System administrators can log out all users of a particular application or database, or log out instances of particular users from a selected scope: an application, a database, or the entire OLAP Server system. If a user request is active when the administrator tries to log off the user, the administrator can force the user off. Using `force safely` terminates the request, and then logs the user off.

Additionally, administrators may terminate specific requests that have become problematic. For example, if a user loses the connection to the OLAP Server during a request (for example, a calculation), the abandoned request may cause the application to stop responding to subsequent requests.

Each user session has a unique ID number, which can be displayed and used by the administrator to terminate the current request associated with that session. When the administrator terminates a request, the user session remains active. The improved handling of sessions and requests can be found in any of these areas:

- Application Manager (Security > Connections)

- The MaxL language interface (**alter system kill request** and **alter system logout session**)
- The DB2 OLAP Server Administration Services Console

The improved handling of sessions and requests is also available to API programmers, who can incorporate the following request management functions:

- EssListRequests()
- EssKillRequest()

Multiple DB2 OLAP Server agents on a single server computer

Three new settings in the DB2 OLAP Server configuration file enable a single computer to run multiple DB2 OLAP Server agents at the same time. These settings provides two important improvements:

- Increased service to clients.
- DB2 OLAP Server administrators can research new systems without the need to use a different server computer. Previous releases required a single communication port per DB2 OLAP Server agent, per OLAP Server. This constraint has been removed.
- These settings must be in `essbase.cfg` configuration files on both the client and server.

The new configuration settings are: AGENTPORT, SERVERPORTBEGIN, SERVERPORTEND, and PORTINC.

Case sensitivity improvement

Previous releases of DB2 OLAP Server automatically changed the case of application and database names when they were created: the first letter was converted to uppercase, and other letters were all lowercase. With this release, DB2 OLAP Server creates the database name exactly as it is entered. Some backward compatibility with previous clients is provided:

- If a client prior to Version 7.1 FixPak 8 (Application Manager for example) is used to create a database or application name, the name is created with the first letter uppercase and all other letters lowercase.
- If scripts specify the incorrect case for an application or database name, DB2 OLAP Server searches for different versions of case for that object.

Some error messages have been rewritten to provide clarity. With this release, behavior on NT and UNIX platforms is identical.

Improved add-in support of of Excel in browsers

DB2 OLAP Server supports the ability access the DB2 OLAP Server Spreadsheet Add-in features when Microsoft Excel is invoked from

Internet Explorer. If you drill down in an Excel spreadsheet accessed using Internet Explorer, Excel refreshes the browser and displays the requested data automatically. Users need not perform any additional steps to refresh the browser view.

Member set function support in FIX

All functions that return a member or set of members are now supported when they occur within FIX statements. Member and member set functions in FIX is also supported for partition definitions and security filters.

New dimensions sparse by default

When you add a dimension to an existing outline, DB2 OLAP Server makes it a sparse dimension. Prior to Version 8.1, DB2 OLAP Server made new dimensions dense by default.

Custom defined functions and macros

Several changes have been made to custom-defined functions (CDF) and custom-defined macros (CDM):

- You no longer need to install the Java Virtual Machine (JVM) to create or use CDMs. CDFs still require the JVM.
- DB2 OLAP Server no longer validates function records before completing the creation process. Errors are detected only when an application loads the CDF. Typically this occurs during server startup and when the MaxL **refresh custom definitions** statement is executed. The DB2 OLAP Server agent checks the function for a valid name, but does no other validation. This means that there is no longer a need to start an application in order to create a function.
- You can now use MaxL to create and drop function records without loading or unloading an application.
- The rules for specifying Java classes and methods are now stricter.
- MaxL and Application Manager must be Version 8.1 to create or delete CDFs and CDMs on the Version 8.1 server. For details, see the *Database Administrators Guide* chapters about CDFs and CDMs.

Attribute association memory use

DB2 OLAP Server has improved memory management for processing attributes requiring less memory. This improvement is most noticeable for applications using many attributes.

Buffered I/O default

Version 8.1 includes buffered I/O as the default. If you want to use direct I/O, as was the default for Release 6.0 and 6.1, you can use the configuration setting `DIRECTIO TRUE`.

Because buffered I/O may affect some DB2 OLAP Server caches, you may need to adjust cache size as well as change the DIRECTIO configuration setting.

DB2 OLAP 8.1 clients not available for Windows 95

DB2 OLAP clients such as Application Manager and the Spreadsheet Add-in for Version 8.1 are not available for Windows 95.

What's new in DB2 OLAP Integration Server 8.1

This section provides an overview of additional significant changes in DB2 OLAP Integration Server Version 8.1. These changes are specific to Release 6.5 of Hyperion Essbase Integration Services.

Hybrid Analysis

Relational databases can store several terabytes of data and thus they have nearly unlimited scalability. DB2 OLAP Integration Server V8.1 includes Hybrid Analysis which offers you a method of integrating your OLAP database with a relational database and thereby take advantage of the scalability of the relational database. With Hybrid Analysis, you can support both multidimensional databases and relational databases, operate with almost no limitation on outline sizes, and provide a rapid transfer of data between OLAP databases and relational databases.

How Hybrid Analysis Works:

- Hybrid Analysis integrates an OLAP database with a relational database by allowing a portion of an Essbase cube to reside in the relational database. This Essbase portion consists of the lowest level members of the dimensions defined for Hybrid Analysis and the associated data of these members.
- Using Essbase Integration Services Console, the data in the relational database is mapped to hierarchies in the Essbase database. Hybrid Analysis thus eliminates the need to load and store members and their data within the OLAP database itself. It is this feature which combines the efficiency of mass data scalability with the sophisticated data analysis and rapid reporting of DB2 OLAP Server.

Using Hybrid Analysis:

- When using Hybrid Analysis, you create combinations of multidimensional and relational data storages. You define your Hybrid Analysis data storage options through Integration Services Console at the member level in the metaoutline.
- In the appropriate metaoutline dialog boxes, you perform the tasks necessary for using Hybrid Analysis:

- Specifying the relational data source for the metaoutline
- Defining the hierarchy for the applicable dimension
- Building the metaoutline down to the appropriate level
- Performing the member and data loads

When you save a metaoutline as Hybrid Analysis-enabled, the console warns you if the metaoutline contains dimensions or transformations that are likely to produce incorrect member names or data values within the Hybrid Analysis data storage.

You query the data from the relational database in much the same manner you query data from an OLAP database, that is, by using reporting tools such as Outline Editor, Essbase Spreadsheet Add-in, Report Writer, and third-party tools.

You can create a Hybrid Analysis storage using the TBC sample application.

Multiple Data Source Connectivity

You can access data from different data sources, primary and secondary, each with its own icon. Multiple data source connectivity gives you considerable flexibility in creating OLAP models and metaoutlines especially in the following functions:

- Creating fact tables and dimensions
- Building logical join relationships with fact tables
- Changing to another data source for a specific dimension
- Connecting to primary and secondary data sources
- Editing the properties of a data source
- Creating new models and metaoutlines
- Opening existing models and metaoutlines

Tables Grouped by Data Source and Owner

With multiple data source connectivity, relational tables are listed, categorized, and ordered by the name of the source or owner and by table type. In the left frame of the OLAP Model main window, tables with the same data source and owner are grouped together.

Verification of Formulas

DB2 OLAP Integration Server provides a rapid method of verifying and editing static formulas before using the formulas in DB2 OLAP Server. Formerly, when a static formula was entered in DB2 OLAP Integration Server, the formula was passed directly to DB2 OLAP Server without any verification. This often caused the formula to be rejected by DB2 OLAP Server because of simple errors such as

misspellings or missing semicolons. Verifying the formula beforehand can save you significant amounts of time during member and data loads.

Formula verification takes place in the Formula tab of the Member Properties Dialog Box in the metaoutline. During verification, the cursor stops at the location of each error. After correcting the error, you can either start the verification process at the beginning of the formula or choose to find the next error in the formula.

Automatic Migration to Version 8.1

In DB2 OLAP Integration Server Version 8.1, when you update an OLAP metadata catalog you created using Version 7.1, DB2 OLAP Integration Server runs a set of scripts that migrate your 7.1 catalog to version 8.1.

Native Driver Support on Oracle

In addition to the Open Database Connectivity (ODBC) drivers already supported, DB2 OLAP Integration Server 8.1 supports Oracle native drivers. This support eliminates the need for database client software and thus significantly facilitates the installation of DB2 OLAP Integration Server on the UNIX operating system. Support of Oracle native drivers also enhances performance on UNIX platforms.

Automatic Configuration Settings for Command Line Parameters

In Version 8.1, you can store server configuration file settings for command line parameters in a single file called `eis.cfg`. The DB2 OLAP Integration Server runs `eis.cfg` which in turn automatically sets the configurations for you. This process eliminates the need for manually entering configurations each time you perform a member or data load.

The following changes are specific to Release 6.2 of Hyperion Essbase Integration Services:

New Client Console with Dynamic Intelligent Help Window

DB2 OLAP Integration Server introduces the new Essbase Integration Services Console and companion Intelligent Help window. The new dockable Intelligent Help window contains dynamic links to key program functions to automate the major operations necessary for moving data from a relational database to online analytical processing (OLAP). This convenient new feature provides the following enhanced installation and client component functionality:

- Automatically detects the OLAP model fact table, dimensions, and hierarchies to create an OLAP model.
- Automatically detects dimensions and hierarchies in an OLAP model to create a basic metaoutline.

OLAP Metadata Catalog and Sample Application Creation

When you first start DB2 OLAP Integration Server Console after installing DB2 OLAP Integration Server, the system displays a new OLAP Metadata Catalog Setup dialog box. For first-time installations, you can click a button in this dialog box to create the OLAP Metadata Catalog. For existing users, if the system detects a previous version of the catalog, your existing catalog is updated to the current version. With Version 8.1, no other migration procedures are required, other than to update your existing OLAP Metadata Catalog. You must have an open database connectivity (ODBC) connection defined for the OLAP metadata catalog before you begin the automatic catalog creation process.

To create the OLAP Metadata Catalog, you first enter the DB2 OLAP Integration Server name and then enter the ODBC Data Source Name (DSN) for the catalog that you want to create. The DSN defines the database where you want to create the catalog. You then enter your user name and password for the DSN and click the OK button. You can choose not to show the OLAP Metadata Catalog Setup dialog box at each startup by clicking a check box. If you disable the option to view the OLAP Metadata Catalog Setup dialog box each time that you open a new session of Integration Services Console, you can select Tools > Create Catalog to view the dialog box again.

After the OLAP Metadata Catalog has been automatically created or updated from a previous version, you can access a new Sample Application Setup dialog box by selecting Tools > Create Sample. This dialog box enables automatic loading of the TBC sample application, including the sample data, sample OLAP model, and sample metaoutline. You must define an ODBC connection for the TBC sample application before you begin the automatic creation process.

To create the sample application, you must first connect to the OLAP Metadata Catalog where you want to store the sample data source, OLAP model, and metaoutline. You then enter the appropriate ODBC Data Source Name and password for the DSN user name TBC (tbc for Informix). DB2 OLAP Integration Server then creates the sample application automatically. The system displays a warning message if it detects that duplicate tables exist from a previous sample application. If a previous sample application is detected, the system will delete and replace the existing tables, or you must provide a new data source name before continuing.

After the OLAP Metadata Catalog has been created and, optionally, the sample TBC application has been created, the system displays the current Login dialog box.

Automatic Detection of Fact Table and Dimensions

After you click the OLAP Model icon for first-time model creation and log in to the appropriate data source, you can choose to have the OLAP model fact table and all related dimensions created automatically.

A quick-access link to the automatic detection option for creating a fact table is provided in the inline text of the Intelligent Help window (see Intelligent Help on page 19). In addition, a Tools menu option (Tools > Create Fact Table) is available from the OLAP Model main window.

When you select the fact table automatic-detection option using either method, the system displays a Create Fact Table dialog box. This dialog box shows the fact tables detected in the current data source to which you are connected, along with any existing fact tables in the OLAP Metadata Catalog that are present in the current data source. You can select either from a list of fact tables in existing OLAP models or from a list of potential fact table candidates to define the fact table to use. Check boxes in the Create Fact Table dialog box enable you to create the Time and Accounts dimensions automatically. If you prefer, you can create the Time or Accounts dimensions manually at a later time.

For automatic detection of dimensions, you can choose the automatic-detection link in the Intelligent Help window or select Tools > Create Dimensions to initiate the dimension-creation process. The system begins by looking for all tables joined to the fact table with primary-foreign key relationships. Next, it searches for all tables joined to the previously detected tables and adds them to the OLAP model schema. In a database where no primary-foreign key relationships are defined, the system begins by searching for all tables joined to the fact table and then uses the column name and type to determine join keys. If two tables have the same column name and type, the system assumes that the two tables are joined on that column. To reduce the possibility of creating invalid dimensions, the system does not search for join keys in any tables that are not joined to the fact table.

Automatic Detection of Hierarchies

Like the Create Fact Table and Create Dimensions options, a new option for automatic detection of hierarchies is provided with Version 8.1. Use the automatic-detection link to this option provided in the text of the Intelligent Help window (see Intelligent Help on page 19). Menu commands on the Tools menu in the OLAP Model and OLAP Metaoutline main windows also provide access to this option. From the OLAP Model main window, select Tools > Create Hierarchies;

from the OLAP Metaoutline main window, select Tools > Get Existing Hierarchies. You can choose to create a single hierarchy or multiple hierarchies.

When you select the automatic hierarchy detection option, using either method, DB2 OLAP Integration Server searches for and detects hierarchies that exist in the OLAP Metadata Catalog. The system then displays the detected hierarchies in a hierarchical tree format that includes the dimension name, the hierarchy names, and the associated member names.

Automatic hierarchy detection does not detect any filters contained within the hierarchies. In addition, the system compares the hierarchical structure of the data source with that of the OLAP model dimensions and deletes any invalid columns from the detected hierarchies. For example, if the OLAP model Product dimension contains Category, Family, and SKU columns, but the data source does not contain a Category column, the hierarchy that is returned contains only Family and SKU columns.

Intelligent Help

This feature enhances user convenience and ease of operation. When you open the OLAP Model or OLAP Metaoutline main window to create or modify an OLAP model or a metaoutline, DB2 OLAP Integration Server Console provides a new option known as Intelligent Help to guide you through the process. Intelligent Help is displayed in a separate window that provides numbered procedures, along with links to new automatic detection options and frequently used functions.

Process buttons along the top of the window contain labels for each main group of related tasks. For example, the process buttons for the OLAP Model tasks groups contain the following labels:

- Fact
- Dimension
- Table and Column properties
- Hierarchy
- Finish

The operations that you need to perform to create an OLAP model fall into these main groupings.

Below the task-group process buttons, the Intelligent Help window consists of a right and left frame:

- The left frame lists the major tasks to be performed for each of the process groups.

- The right frame contains detailed, numbered procedures for performing the tasks itemized in the left frame. Links to automatic detection functions and to user interface dialog boxes are provided in this frame, where appropriate.

The Intelligent Help window is a dockable, relocatable window that you can move and resize as you prefer. Integration Services Console displays the Intelligent Help window automatically the first time that you create or open an OLAP model or metaoutline. You can click an icon to undock the Intelligent Help window and move it away from the console. You can click a check box to show or hide the Intelligent Help window at startup, and you can click a Close icon to close the window at any time during a console session.

Both the toolbar and the Tools menu in the OLAP Model and OLAP Metaoutline main windows provide options for selecting Intelligent Help. The system remembers whether the Intelligent Help window was open or closed during your last work session. If the Intelligent Help window is open when you close an OLAP model, the window is displayed automatically when you reopen the model to work on it at a later time.

As always, online help is available to provide in-depth conceptual information and dialog box entry details.

Load optimization

DB2 OLAP Version 8.1 provides five dynamic new load optimization options for improving member and data load performance:

Member load options

- **Use Distinct Clause:** If the leaf node of a given dimension should be distinct, you can choose to add the Distinct clause to the Select statement. During a member load, only the distinct combination of columns in the Select statement is returned, resulting in a quicker member load. You might want to use this option, for example, if you have created a Time dimension from the fact table.
- **Use Outer Join:** Use of this feature enables you to view all columns in your DB2 OLAP Server database, whether or not the columns between two joined tables are identical.

The result of a natural (inner) join between two tables includes records where the joined columns match. In any row from Table A in which a join column does not exist in Table B, that row is dropped when data is loaded. In an outer join, however, when an unmatched column is encountered during a data load, the unmatched row is included in the results. The column data of the unmatched portion is replaced with a <NULL> value.

- **Allow Duplicate Shared Members:** Use of this feature enables the same member to be shared multiple times under the same parent, resulting in faster creation of the DB2 OLAP Server outline.

Data load options

- **Remove Extra Joins:** This feature removes extra joins whether or not a surrogate key is used for a data load. If a surrogate key is used in a join, the surrogate key is loaded into DB2 OLAP Server as an alias, and the join is eliminated in the data load.
- **Use User-defined SQL:** This option enables you to use an edited version of the standard SQL generated by DB2 OLAP Integration Server.

A new OLAP metaoutline SQL Override dialog box displays standard data load Select statements. You can select the statements that you want to edit. The two-frame dialog box shows the current metaoutline in the left frame and three windows in the right frame. The top right-frame window lists the Current Selected SQL and is the window in which you edit Select statements. The center window displays the User-Defined SQL List that you create. The bottom window displays the Standard Data Load SQL generated by DB2 OLAP Integration Server.

When you have completed editing of standard SQL statements, you can direct the system to use the edited version for a data load by checking the Use User-defined SQL for Data Load option in the OLAP Metaoutline Properties dialog box, Load Optimization tab. For rules and guidelines related to editing the standard data load SQL generated by DB2 OLAP Integration Server, click the Help button in the OLAP Metaoutline SQL Override dialog box.

Sort on alternate column

This feature enables you to sort on a column that resides in the relational data source, but which is not used in the OLAP metaoutline. Sorting on a nondisplayed column is enabled in both recursive builds and generation builds. The dialog boxes that you use to specify ordering and filtering display identical column lists for selection. To prevent multiple selections of the same column, a column that has been selected at any generation of ordering is no longer available for sort selection.

Loading of DB2 OLAP Server Properties from Database Columns

This new feature enables the following DB2 OLAP Server properties to be loaded from database columns in the same way that Consolidation Attributes and UDAs currently are loaded from DB2 OLAP Integration Server into DB2 OLAP Server:

- Two-pass calculation
- Data storage attribute

- Time balance
- Skip type
- Variance reporting
- Formulas

Inclusion of this feature follows the DB2 OLAP Server Administration Services Console model, enabling you to define major DB2 OLAP Server settings from within DB2 OLAP Integration Server. You define these member settings using the metaoutline Member Properties dialog box, Member Info tab. When property values are combined in one database column, you can separate them with a comma or a space (for example: +, Expense, Last, Both). When you use a word to define a property, only the first letter is used during a member load; for example, (O)nly or (E)xpense or (T)wo Pass Calc.

For details on allowable values for properties loaded from database columns, click the Help button in the Member Info tab of the metaoutline Member Properties dialog box.

XML Import/Export

DB2 OLAP Version 8.1 XML Import/Export enables you to import data into and export data from the OLAP Metadata Catalog. You can now quickly transfer OLAP models and metaoutlines from one OLAP Metadata Catalog to another through use of this feature. You can also use an XML-based editor to rename the OLAP model associated with a metaoutline and import the revised data into the OLAP Metadata Catalog. Be sure to use an XML editor that supports UTF 8 UNICODE.

Extended Markup Language (XML) consists of two parts:

- **Data Type Definition (DTD):** This part describes the allowable structure of XML documents. A DTD defines the major elements within a document, the child elements, and related attributes. It can constrain the pieces of data that occur in a document, the hierarchy of the data that it includes, and the number of times each piece of data occurs. DB2 OLAP Integration Server provides a predefined DTD to ensure that you use the same data format when you create XML documents.
- **XML documents:** This part specifies how to use the tagged markup to indicate the meaning of data, somewhat like a card catalog. In DB2 OLAP Integration Server, the data in an XML document describes the dimensions, members, hierarchies, and settings in OLAP models and metaoutlines. This information is stored in tables in the OLAP Metadata Catalog.

To use the new XML Import/Export feature, DB2 OLAP Integration Server provides a new dialog box named the XML Import/Export

dialog box. Select the File > XML Import/Export command from the OLAP Model or OLAP Metaoutline main window to access the dialog box.

You can then select the Import tab and enter the necessary information to open either an OLAP model or metaoutline from an XML file. You can also select the Export tab to save an OLAP model or metaoutline as an XML file. Click the Save As XML File button, enter an XML file name, and navigate to the directory where you want to save the file.

Incremental Update to Data

This feature enables you to add to and subtract from DB2 OLAP Server data during member loads and data loads. By default, DB2 OLAP Integration Server updates all dimensions and members. Incremental updating of an existing DB2 OLAP Server outline is faster than updating all dimensions and members. The Essbase Integration Services Console includes an Incremental Update dialog box to use for selecting update options.

For member load, you have the following options:

- Preserve all data (this is the default setting): preserves all existing data that applies to the changed outline when restructuring occurs.
- Preserve input data: preserves only those blocks containing data that is loaded.
- Preserve level 0 data: preserves data only for level zero members.
- Discard all data: clears all data from the database.

For data load, you have the following options:

- Overwrite (this is the default setting): replaces the values in the database with the values in the data source.
- Add: adds values in the data source to the existing values in the database.
- Subtract: subtracts the values in the data source from the existing values in the database.

For both member and data loads, you have the following Attribute Dimension Restriction options:

- Delete all dimensions (this is the default setting): deletes all attribute dimensions associated with any base dimension selected for the incremental update.
- No update to existing dimensions: keeps attribute dimensions unchanged while updating other areas.
- Update existing dimensions: updates all changes to attributes.

Member Load and Data Load Filter Options

This feature enables you to limit filters to a member load or a data load, or to apply filters to both member and data loads.

The Essbase Integration Services Console includes a revised DB2 OLAP Server Application and Database dialog box with a simple drop-down list from which to select or deselect filters.

Improved SQL Drill-Through

This feature takes advantage of the new data load optimizations by including join key optimization in the SQL for drill-through. In join key optimization, the join between a fact table and a dimension table is removed if the join column is the same as the column selected. This results in faster drill-through queries.

Windows Service Utility Support

This new utility adds the Windows NT and Windows 2000 service for DB2 OLAP Integration Server, accepting parameters supported by the `olapisvr` command. These parameters enable you to:

- View a list of available switches for starting DB2 OLAP Integration Server.
- Set the buffer size that DB2 OLAP Integration Server uses when transferring rows from the data source to the DB2 OLAP Server database.
- Set the number of records that DB2 OLAP Integration Server commits to DB2 OLAP Server during a data load.
- Give the log file a different name.
- Set the level of detail in the messages logged by DB2 OLAP Integration Server.
- Set the DB2 OLAP Server message database file to a different name.
- Specify the number of threads for executing SQL, retrieving data, transforming data, and performing a data load.
- Set the TCP port number at which DB2 OLAP Integration Server connects with a client.
- Set the number of network listeners that DB2 OLAP Integration Server starts.

Improved Access to the Server Log File

To assist users in analyzing and debugging member and data loads, DB2 OLAP Integration Server now provides two methods of accessing the Server log file:

- During a member or data load, or at any other time, you can select Tools > View Log File to view the Server log file.

- At the completion of a member or data load, a View Log File option button is enabled that you can select to view the Server log file.

In both cases, DB2 OLAP Integration Server retrieves the log data and displays it in the Server Log File window with a date and time stamp for each activity that has occurred during the load. For very large member or data loads, the server retrieves only the last 1 MB of the log file. You can copy the log file and paste it to any text editor, such as Notepad, to print it out for review and analysis.

New Sample Application

DB2 OLAP Integration Server provides a new sample application based on a fictitious company named The Beverage Company (TBC). The TBC sample application demonstrates creating a Measures dimension recursively, using a sort on an alternate column in the relational data source, and loading metadata from database columns. Additionally, the sample application includes UDAs and Date Time Series data, additional years, new drill-through paths, and a new Time table to demonstrate data load incremental updates.

The TBC sample application includes the following components:

- Sample OLAP Metadata Catalog (TBC_MD)
- Sample data source (TBC)
- Sample OLAP model (TBC Model) and sample OLAP metaoutline (TBC Metaoutline)

For the first time, the sample application OLAP model and OLAP metaoutline are provided in Java-supported XML format.

The Essbase Integration Services Console provides a new Sample Application Setup dialog box that is displayed automatically after installation of DB2 OLAP Integration Server. With the completion of a few simple entries in the dialog box, you can now load the sample application automatically.

This automated feature enables you to create a functional sample application that is ready to use within minutes. You must set up the database for the OLAP Metadata Catalog and the TBC data source, and define open database connectivity (ODBC) connections beforehand. When setting up ODBC connections, the user name that you use when defining the Data Source Name (DSN) for the data source must be TBC (tbc for Informix), because the sample tables were created using this name.

Teradata support

DB2 OLAP Integration Server, Version 8.1, now supports Teradata, a database from NCR Corporation, using NT or UNIX operating

systems. Teradata is supported both as a relational data source and as an OLAP Metadata Catalog storage database.

UNIX ODBC Configuration Utility

DB2 OLAP Integration Server provides a new utility for UNIX users to verify, add, or delete ODBC connections. The `odbcconfig` utility enables you to edit ODBC configuration information in the `odbc.ini` file. This new utility also displays current environment settings related to relational database management system (RDBMS) access. You can use the `odbcconfig` utility for diagnostic testing of environment settings.

The `odbcconfig` utility provides the following menu options to aid you in configuring and troubleshooting ODBC connections:

- **List:** Use this option to list all defined DSNs for the OLAP Metadata Catalog and source databases and verify that they are correct.
- **Add:** Use this option to add a DSN for the OLAP Metadata Catalog and source databases.
- **Delete:** Use this option to delete a DSN for the OLAP Metadata Catalog and source databases.
- **Test:** Use this option to test that the ODBC connections that you created are valid.

Chapter 2. Before you install

This chapter describes the overall installation process and steps to take before installing DB2 OLAP Server Version 8.1

Read the installation checklist

This checklist summarizes the major steps of installing DB2 OLAP Server and DB2 OLAP Integration Server:

- Verify that you have the hardware and software required by DB2 OLAP Server and DB2 OLAP Integration Server. See Chapter 3, “Hardware and software Requirements” on page 31.
- Determine what products and features you purchased and are entitled to install.
- Determine what free features and options you want to install:
 - IBM DB2 OLAP Server Miner is a no-cost add-on feature of DB2 OLAP Server that automatically mines through large volumes of OLAP data and identifies and reports on specific elements that represent the most outstanding, and unexpected, pieces of information. You can expose problems and opportunities that were previously hidden and act on them quickly. For information about installing OLAP Miner, see “Installing DB2 OLAP Server Miner on Windows” on page 49 and “Installing DB2 OLAP Server Miner on AIX” on page 65.
 - You can place the installation files for the client components on a network so you users can install the clients without having a CD-ROM. For more information, see “Setting up client installation from a network file server” on page 50
 - If you choose the **Typical** or **Compact** options during installation, DB2 OLAP Server installs only a subset of the documentation. To install all the documentation, choose **Custom** and select the books you want.
- If you have an earlier version of DB2 OLAP Server on your computer, prepare for migration:
 - Back up your entire previous installation directory, or at least following files:
 - Your data
 - Essbase.sec
 - license.*
 - Migrate your Relational Storage Manager (RSM) applications to the Multidimensional Storage Manager (MSM) before installing Version 8.1.

CAUTION:

The RSM is withdrawn from Version 8.1. If you are upgrading from DB2 OLAP Server Version 7.1 or earlier, and you have applications that use the RSM, you must migrate them to the MSM in Version 7.1 before installing Version 8.1, or you will lose data. See “Migrate from RSM to MSM before upgrading to Version 8.1” on page 131 for more information.

- If you installed the OLAP Miner at the Version 7.1 level, back up the OLAP Miner configuration files for both the server and clients. The OLAP Miner server configuration file is `omserver.cfg` and the file for the clients is `omclient.cfg`.
- Stop any OLAP services that are running.
- Install the server components and client components. See either Chapter 4, “Installing on Windows” on page 45 or Chapter 5, “Installing on AIX, Solaris Operating Environment, and HP-UX” on page 61. If you want to make the Windows clients available on a LAN so your users can download and install them without a CD-ROM, see “Setting up client installation from a network file server” on page 50.
- Install Java for both DB2 OLAP Server and DB2 OLAP Integration Server. See Chapter 6, “Installing and configuring Java” on page 73.
- If you want to install DB2 OLAP Server Miner, see “Installing DB2 OLAP Server Miner on Windows” on page 49 or “Installing DB2 OLAP Server Miner on AIX” on page 65.
- If you installed the SQL Interface, configure it and set up ODBC using the instructions in Chapter 7, “Configuring the SQL Interface” on page 85
- If you installed DB2 OLAP Integration Server, create metadata catalogs. See Chapter 9, “Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server” on page 105.
- If you installed DB2 OLAP Integration Server, configure your relational data sources and set up ODBC by following the instructions in Chapter 8, “Configuring data sources for DB2 OLAP Integration Server” on page 97.
- Set up the sample applications. See Chapter 10, “Setting up the sample applications” on page 111.
- Migrate your applications to Version 8.1, if any. See Chapter 11, “Upgrading and migrating from earlier versions” on page 131.

Prepare to upgrade DB2 OLAP Integration Server

If you are upgrading to DB2 OLAP Integration Server Version 8.1 from a previous release, you should assess several system environment and data migration issues before installing the new software.

The following tasks cannot be completed until after you have installed Essbase Integration Services:

- On UNIX systems, you cannot update environment variables until after the installation process has completed and the environment variable script files you need have been created (see “Updating the environment settings” on page 66).
- 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 110).

Preserving an existing ODBC Driver

DB2 OLAP Integration Server architecture concentrates ODBC driver configuration solely on the computer that runs DB2 OLAP Integration Server. With this architecture, you do not have to configure ODBC drivers for client computers that run Essbase Integration Services Console or worry about upgrading ODBC drivers on clients.

If you use the MERANT Version 3.6 ODBC drivers supplied with earlier versions, you may want to preserve the older drivers for compatibility with other applications. In general, however, you should upgrade the ODBC drivers to the versions provided with the latest release of Essbase Integration Services. DB2 OLAP Integration Server Version 8.1 supports MERANT Version 4.0 ODBC drivers.

DB2 OLAP Integration Server Version 8.1 does not support MERANT Version 3.6 ODBC drivers.

The directory structure provided with DB2 OLAP Integration Server Version 8.1 lets you to retain and continue to use older versions of MERANT drivers without the need to install DB2 OLAP Integration Server to a new location.

Migrating existing OLAP Metadata Catalogs

If you have an existing OLAP Metadata Catalog from an earlier version of DB2 OLAP Integration Server, 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 Essbase Integration Services setup program installs the scripts that are needed for the upgrade (see “Upgrading the OLAP Metadata Catalog” on page 110).

CAUTION:

If you have installed the sample application from a previous release of Essbase Integration Services, 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:

1. Complete the upgrade steps for the existing OLAP Metadata Catalog described in “Upgrading the OLAP Metadata Catalog” on page 110.
2. 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 release. The new release of OLAP Metadata Catalog is not compatible with previous releases of DB2 OLAP Integration Server. In addition, do not attempt to use the Version 8.1 OLAP Metadata Catalog with previous releases of the software; doing so could result in corrupted catalog data.

Check compatibility with earlier versions

It is recommended that you install the Version 8.1 OLAP server component with Version 8.1 client products. Client products include the Application Manager, Spreadsheet Add-ins for Microsoft Excel and Lotus 1–2–3, and the Runtime Client.

There are limitations when you mix different versions of clients and servers in the same environment:

Newer clients with older servers

Using a client that is newer than the server is not recommended or supported.

Older clients with newer servers

Using a client that is older than the server version is not supported. However, some clients of previous releases might work with the Version 8.1 server, although with feature limitations.

Be careful when modifying Version 8.1 databases or outlines with earlier releases of Application Manager. If you save a Version 8.1

database that contains new Version 8.1 features with an earlier release of the Application Manager, the new features may be stripped from the database.

Depending on the service level of an older client, it might not be able to connect with a newer server. Additionally, the newer server might issue messages that are incorrect in the context of the older client.

One client with many servers

If you need to connect a client to several OLAP servers on different machines or operating systems, it is recommended that you make them all the same version and service level. Your client should at least be at the same as the lowest level server. If you experience problems, you might need to use different client machines to match up with the appropriate host, or upgrade all clients and servers to the same service level.

Different versions of clients on the same client system

If you install more than one client component on the same computer, such as Spreadsheet Add-in and Application Manager, make sure that all of the components are on the same release level. Version 8.1 clients share.dll files that may not work with earlier client components.

IBM components with Hyperion components

IBM does not support mixing OLAP clients and servers from IBM with OLAP clients and servers from Hyperion Solutions. There are some differences in feature mixes, levels, and some issues in licensing that may cause problems, even though mixing IBM with Hyperion products might appear to work in some situations.

Chapter 3. Hardware and software Requirements

This chapter describes the hardware and software that you need to run DB2 OLAP Server on various operating systems.

Use the following tables to verify that DB2 OLAP Server Version 8.1 supports the platforms where you plan to install client and server software.

- Server Platform System Requirements
- Client Platform System Requirements
- API Client Development Platform System Requirements
- Supported ODBC Drivers and Relational Data Sources
- Supported JDBC Drivers for XML Import/Export in OIS

Server hardware and software requirements

Table 2 contains a summary of the system requirements of the server components of DB2 OLAP Server and DB2 OLAP Integration Server:

Table 2. Summary of server requirements

Server Operating Systems	Supported Versions
Microsoft Windows See Table 3 on page 32 for more details.	Pentium or later computers for the following versions of Windows: <ul style="list-style-type: none">• Windows XP• Windows 2000• Windows NT 4.0; Service Pack 5 is required by DB2 OLAP Integration Server Windows NT 3.5.1 is not supported.
Solaris Operating System See Table 4 on page 33 for more details.	Sun Solaris Operating System 7 and 8 (SUN OS 5.7, and 5.8) on Sun SPARC or ULTRASPARC computers
AIX See Table 5 on page 34 for more details.	4.3.3 or 5L on RS 6000 PowerPC computers
HP-UX See Table 6 on page 36 for more details.	11.0 or 11i on PA-RISC computers.

DB2 OLAP Integration Server Version 8.1 requires, as a minimum, FixPak 8 of DB2 OLAP Server Version 7.1. If you want to you Hybrid Analysis, you need to install DB2 OLAP Server Version 8.1.

Windows server requirements

DB2 OLAP Server for Windows platforms requires a system that meets specific requirements. These requirements are described in Table 3.

Table 3. Windows system requirements

Component	Requirements
Microprocessor	Pentium or higher
RAM	64 MB or greater Member Load formula for DB2 OLAP Integration Server: 10 MB + (700 Bytes x number of members)
Microsoft Windows version	<ul style="list-style-type: none"> • Windows XP • Windows 2000 • Windows NT 4.0; Service Pack 5 is required by DB2 OLAP Integration Server <p>Windows NT 3.5.1 is not supported</p>
Disk space	<ul style="list-style-type: none"> • For DB2 OLAP Server, approximately 65 to 85 MB for the server software and the sample applications • For DB2 OLAP Integration Server: <ul style="list-style-type: none"> – 20 MB for server software – 1 MB for OLAP Metadata Catalog (as shipped) – 30 MB for OLAP Metadata Catalog in the relational database – 30 MB for the sample application (as shipped) – 20 MB for the sample application without Hybrid Analysis data (on the RDBMS server computer) – 40 MB for the sample application with Hybrid Analysis data (on the RDBMS server computer) – 22 MB for Java and Jar files – Totals: <ul style="list-style-type: none"> - 73 MB total (as shipped) - 92 MB total (set up, without Hybrid Analysis sample) - 112 MB total (set up, with Hybrid Analysis sample) <p>Plus, up to 15 MB for optional documentation.</p>
Network protocol	Named Pipes or TCP/IP. For OIS, TCP/IP only.

Table 3. Windows system requirements (continued)

Component	Requirements
Java platform	Java Runtime Environment version 1.3 or later, for both DB2 OLAP Server and DB2 OLAP Integration Server

Solaris Operating System server requirements

DB2 OLAP server for the Solaris Operating System requires a system with the minimal hardware and software requirements shown in Table 4.

Table 4. Solaris system requirements

Component	Requirement
Server platform	Sun SPARC or ULTRASPARC machines
Solaris Operating System version	Sun Solaris Operating System 7 and 8 (SUN OS 5.7, and 5.8)
RAM	<p>For DB2 OLAP Server: 128 MB RAM (256 or more MB recommended, especially for large databases)</p> <p>For DB2 OLAP Integration Server:</p> <ul style="list-style-type: none"> • 256 MB for the server software • The minimum ulimit settings are: Stack 50 MB; Data 100 MB; Memory 100 MB • Member Load formula: 50 MB + (700 Bytes x number of members)
Swap space	At least two times the amount of physical memory is recommended when physical memory is less than .5 GB. Contact your system administrator or hardware vendor for optimum swap configuration.

Table 4. Solaris system requirements (continued)

Component	Requirement
Disk space	<p>For DB2 OLAP Server:</p> <ul style="list-style-type: none"> • 75 MB • 38 MB for server software • 15 MB for sample applications as shipped; 37 MB after they are fully loaded with default settings <p>For DB2 OLAP Integration Server:</p> <ul style="list-style-type: none"> • 20 MB for server software • 1 MB for OLAP Metadata Catalog (as shipped) • 30 MB for OLAP Metadata Catalog in the relational database • 30 MB for the sample application (as shipped) • 20 MB for the sample application without Hybrid Analysis data (on the RDBMS server computer) • 40 MB for the sample application with Hybrid Analysis data (on the RDBMS server computer) • 22 MB for Java and Jar files • Totals: <ul style="list-style-type: none"> – 73 MB total (as shipped) – 92 MB total (set up, without Hybrid Analysis sample) – 112 MB total (set up, with Hybrid Analysis sample)
Network protocol	TCP/IP (included with Solaris)
Thread type	POSIX Kernel Threads (included with Solaris)
Java platform	<p>For DB2 OLAP Server: Java Runtime Environment version 1.3 or later</p> <p>For DB2 OLAP Integration Server: Java Runtime Environment version 1.3.0_02 or later</p>

AIX server requirements

The OLAP server for AIX requires a system with the minimal hardware and software requirements shown in Table 5.

Table 5. AIX system requirements

Component	Requirement
Server platform	RS 6000 PowerPC machines
AIX version	4.3.3 for 5L

Table 5. AIX system requirements (continued)

Component	Requirement
RAM	<p>For DB2 OLAP Server: 128 MB RAM (256 or more MB recommended) Larger databases have greater memory requirements.</p> <p>For DB2 OLAP Integration Server</p> <ul style="list-style-type: none"> • 256 MB for the server software • The minimum ulimit settings are: Stack 50 MB; Data 100 MB; Memory 100 MB • Member Load formula: 50 MB + (700 Bytes x number of members)
Swap space	<p>At least two times the amount of physical memory is recommended when physical memory is less than .5 GB. Contact your system administrator or hardware vendor for optimum swap configuration.</p>
Disk space	<p>For DB2 OLAP Server:</p> <ul style="list-style-type: none"> • 75 MB • Server software uses 38 MB • Sample applications use 15 MB as shipped and use 37 MB after they are fully loaded with default settings. <p>For DB2 OLAP Integration Server:</p> <ul style="list-style-type: none"> • 20 MB for server software • 1 MB for OLAP Metadata Catalog (as shipped) • 30 MB for OLAP Metadata Catalog in the relational database • 30 MB for the sample application (as shipped) • 20 MB for the sample application without Hybrid Analysis data (on the RDBMS server computer) • 40 MB for the sample application with Hybrid Analysis data (on the RDBMS server computer) • 22 MB for Java and Jar files • Totals: <ul style="list-style-type: none"> – 73 MB total (as shipped) – 92 MB total (set up, without Hybrid Analysis sample) – 112 MB total (set up, with Hybrid Analysis sample)
Network protocol	TCP/IP (included with AIX)
Thread type	POSIX Kernel Threads (included with AIX)

Table 5. AIX system requirements (continued)

Component	Requirement
Java platform	For DB2 OLAP Server: Java Runtime Environment version 1.2.2 or later For DB2 OLAP Integration Server: Java Runtime Environment version 1.3.0 or later

HP-UX server requirements

The OLAP server for HP-UX requires a system with the minimal hardware and software requirements shown in Table 6.

Table 6. HP-UX system requirements

Component	Requirement
Server platform	PA-RISC machines
HP-UX version	11.0, plus the latest HP-UX Quality Pack, available from the HP Web site (www.software.hp.com/). Search for: hp-ux quality pack bundles
RAM	For DB2 OLAP Server: 128 MB RAM (256 or more MB recommended) Larger databases have greater memory requirements. For DB2 OLAP Integration Server <ul style="list-style-type: none"> • 26 MB for the server software • The minimum ulimit settings are: Stack 50 MB; Data 100 MB; Memory 100 MB • Member Load formula: 50 MB + (700 Bytes x number of members)
Swap space	At least two times the amount of physical memory is recommended when physical memory is less than 0.5 GB (500 MB). Contact your system administrator or hardware vendor for optimum swap configuration.

Table 6. HP-UX system requirements (continued)

Component	Requirement
Disk space	<p>For DB2 OLAP Server:</p> <ul style="list-style-type: none"> • 75 MB • Server software uses 38 MB • Sample applications use 15 MB as shipped and use 37 MB after they are fully loaded with default settings. <p>For DB2 OLAP Integration Server:</p> <ul style="list-style-type: none"> • 20 MB for server software • 1 MB for OLAP Metadata Catalog (as shipped) • 30 MB for OLAP Metadata Catalog in the relational database • 30 MB for the sample application (as shipped) • 20 MB for the sample application without Hybrid Analysis data (on the RDBMS server computer) • 40 MB for the sample application with Hybrid Analysis data (on the RDBMS server computer) • 22 MB for Java and Jar files • Totals: <ul style="list-style-type: none"> – 73 MB total (as shipped) – 92 MB total (set up, without Hybrid Analysis sample) – 112 MB total (set up, with Hybrid Analysis sample)
Network protocol	TCP/IP (included with HP-UX)
Thread type	POSIX Kernel Threads (included with HP-UX)
Java platform	<p>For DB2 OLAP Server: Java Runtime Environment version 1.2.2 or later</p> <p>For DB2 OLAP Integration Server: Java Runtime Environment version 1.3.0.01 or later</p>

Client hardware and software requirements

The system requirements of the Application Manager, the Runtime Client, the Spreadsheet Add-ins for Excel and Lotus 1-2-3, and the OIS Essbase Integration Services Console are described in Table 7 on page 38. API client requirements are described in “API client hardware and software requirements” on page 39.

Table 7. Client requirements for DB2 OLAP Server and DB2 OLAP Integration Server:

Component	Requirement
Microsoft Windows version	<ul style="list-style-type: none"> • Windows XP • Windows ME • Windows 2000 • Windows NT 4.0; Service Pack 5 is required by DB2 OLAP Integration Server • Windows 98 <p>Windows 95 and Windows NT 3.5.1 are not supported</p>
Microsoft Excel version	Excel 2000, Excel XP
Lotus 1-2-3 version	Lotus 1-2-3 97, plus 9.7, 9.6 and 9.5
Microprocessor	Pentium or better or equivalent
Display	Resolution of at least 640 x 480 (800 x 600 or more recommended)
RAM	<ul style="list-style-type: none"> • 16 MB for DB2 OLAP Server Spreadsheet Add-in and Application Manager • 32 MB for the Runtime Client • 64 MB for the OIS Essbase Integration Services Console
Disk space	<ul style="list-style-type: none"> • 24 MB for Application Manager • 18 MB for the Spreadsheet Add-in for Lotus 1-2-3 • 17 MB for the Spreadsheet Add-in for Excel • 9 MB for the Runtime Client • 45 MB for the OIS Essbase Integration Services Console • 75 MB for the documentation, if you install all of it
Network protocol	TCP/IP or Named Pipes. For OIS, TCP/IP only.

API client hardware and software requirements

The system requirements of the API client development platforms on Windows and UNIX platforms are described in Table 7.

Table 8. API client development platforms

Platform	Supported versions
Windows version	<ul style="list-style-type: none">• Windows XP on Pentium or better, or equivalent computers• Windows ME on Pentium or better, or equivalent computers• Windows NT 4.0 on Pentium or better, or equivalent computers• Windows 2000 on Pentium or better, or equivalent computers
Solaris Operating System version	Sun Solaris Operating System 7 and 8 (SUN OS 5.7, and 5.8) on Sun SPARC or ULTRASPARC computers
AIX version	4.3.3 and 5L on RS 6000 PowerPC computers
HP-UX version	11.0 and 11i on PA-RISC computers
RAM	<ul style="list-style-type: none">• 64 MB RAM (128 or more MB recommended) for Windows 2000, Windows NT , or Windows 98• Physical memory: 64 MB (256 MB recommended) for AIX, HP-UX, and Solaris <p>Swap space: At least two times the amount of physical memory is recommended when physical memory is less than .5 GB. Contact your system administrator or hardware vendor for optimum swap configuration.</p>
Disk Space	<ul style="list-style-type: none">• 15 MB for 32-bit DB2 OLAP Server API on Windows XP, Windows 2000, Windows NT , on Pentium• 10 MB for AIX, HP-UX, and Solaris

Supported ODBC drivers and relational data sources

In most cases, DB2 OLAP Server and DB2 OLAP Integration Server requires the minimal levels of ODBC drivers. If you install both products, check the following tables to make sure you use an ODBC driver that meets the minimum requirements of both products.

For DB2 OLAP Server

MERANT (formerly INTERSOLV) ODBC driver version 4.0 is provided with DB2 OLAP Server Version 8.1 SQL Interface. Additionally, DB2 OLAP Server also supports some non-MERANT Windows NT drivers, which are listed in

Table 9. These non-MERANT drivers are not provided with SQL Interface.

Table 9. Supported ODBC drivers for DB2 OLAP Server

Relational Database Server (Relational Data Source)	ODBC drivers per supported operating system			
	Windows NT 4.0, Windows 2000, Windows XP	Solaris 7, 8 (Sun OS 5.7, or 5.8)	AIX 4.3.3, AIX 5L	HP-UX 11.0 ¹ , HP-UX 11i
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
DBASE V	Microsoft Driver	MERANT 4.0	Not supported	MERANT 4.0
Informix 9.x (using Informix client)	MERANT 4.0	MERANT 4.0 (ESQL 9.2)	MERANT 3.6 (ESQL 9.2)	Not supported
Oracle 8i (SQL Net 8)	MERANT 4.0	MERANT 4.0	MERANT 3.6 ²	MERANT 4.0
Oracle 9i (Oracle Net)	MERANT 4.0	MERANT 4.0	MERANT 3.6 ²	MERANT 4.0
MS SQL Server 7.0	MS SQL Server 7.0 ODBC	Not supported	Not supported	Not supported
MS SQL Server 2000	MS SQL Server 2000 ODBC	Not supported	Not supported	Not supported
Sybase 11.x (Adaptive Server)	MERANT 4.0	MERANT 4.0	MERANT 3.6 ²	Not supported
Sybase 12	MERANT 4.0	MERANT 4.0	MERANT 3.6 ²	Not supported
Access 2000	Microsoft Driver	Not supported	Not supported	Not supported
Access 2002	Microsoft Driver	Not supported	Not supported	Not supported

Notes:

1. MERANT 4.0 ODBC drivers are supported on HP-UX 11.0 computers when using 32-bit relational database client software only.
2. For more information about using ODBC 3.6, see "Preserving an existing ODBC Driver" on page 27.

For DB2 OLAP Integration Server

DB2 OLAP Integration Server supports the ODBC drivers for the relational database management systems (RDBMS) and operating systems shown in Table 10.

Table 10. Supported ODBC drivers for DB2 OLAP Integration Server

Relational Database Server (Relational Data Source)	ODBC drivers per supported operating system			
	Windows NT 4.0, Windows 2000, Windows XP	Solaris Operating System 7, 8 (Sun OS 5.7, 5.8)	AIX 4.3.3, AIX 5L	HP-UX 11.0, HP-UX 11i ¹
DB2 UDB 6.1 (DB2 UDB 6.1)	DB2 6.1 ODBC ²	DB2 6.1 ODBC	DB2 6.1 ODBC	DB2 6.1 ODBC
DB2 UDB 7.1 (DB2 UDB 7.1)	DB2 7.1 ODBC ²	DB2 7.1 ODBC	DB2 7.1 ODBC	DB2 7.1 ODBC
DBASE V	Not supported	Not supported	Not supported	Not supported
Informix 9.2 (Connect 2.4)	MERANT 4.0 ²	MERANT 4.0	Not supported	MERANT 4.0
Oracle 8i ² (Net8)	MERANT 4.0 ²			
	Oracle ODBC ²	MERANT 4.0	MERANT 4.0	MERANT 4.0
	OCI direct connection ²	OCI direct connection	OCI direct connection	OCI direct connection
Oracle 9i ² (Oracle Net)	MERANT 4.0			
	Oracle ODBC	MERANT 4.0		
	OCI direct connection	OCI direct connection	Not supported	Not supported
Oracle 9i ² (Net8)	MERANT 4.0 ²			
	Oracle ODBC ²	MERANT 4.0	MERANT 4.0	MERANT 4.0
	OCI direct connection ²	OCI direct connection	OCI direct connection	OCI direct connection
MS SQL Server 7.0 (no client required)	MS SQL Server 3.7 ODBC (included in MDAC 2.5)	Not supported	Not supported	Not supported

Table 10. Supported ODBC drivers for DB2 OLAP Integration Server (continued)

	MS SQL Server 3.7 ODBC (included in MDAC 2.5)			
MS SQL Server 2000 (no client required)	MS SQL Server 2000.80	Not supported	Not supported	Not supported
Sybase 11.x (Adaptive Server)	Not supported	Not supported	Not supported	Not supported
Sybase 12 (On UNIX, Open Client 11.1; On Windows, no client required)	MERANT 4.0 Wire Protocol	MERANT 4.0 Wire Protocol	Not supported	Not supported
Access 2000	Not supported	Not supported	Not supported	Not supported
Access 2002	Not supported	Not supported	Not supported	Not supported

Notes:

1. MERANT 4.0 ODBC drivers are supported on HP-UX 11.0 computers when using 32-bit relational database client software only.
2. This RDBMS is not supported on Windows XP.

DB2 OLAP Integration Server provides MERANT 4.0 ODBC drivers for AIX, HP-UX, Solaris Operating System, Windows NT, Windows 2000, and Windows XP.

ODBC drivers used with DB2 OLAP Integration Server must be thread-safe. All drivers listed in Table 10 on page 41 are thread-safe. HP-UX requires kernel-threads. Solaris Operating System requires Solaris threads, and AIX requires kernel-threads with pthread API.

Supported JDBC Drivers for XML Import/Export in OIS

DB2 OLAP Integration Server requires JDBC drivers to support the XML Import/Export function.

- JDBC Type 1 is compatible with all RDBMSs supported by DB2 OLAP Integration Server. It is installed when you install the Java Runtime Environment (included as an option in the Windows installation; included as a separate installation for UNIX).
- The JDBC Type 3 driver works on all platforms.

- The JDBC Type 4 driver is required by DB2 OLAP Integration Server if you are using Oracle with OCI. The JDBC Type 4 driver for Oracle works on all platforms.

Chapter 4. Installing on Windows

This chapter tells you how to install server and client components for DB2 OLAP Server Version 8.1 on Windows operating systems.

Installing DB2 OLAP Server

Make sure you consider the following important changes in how DB2 OLAP Server is installed in version 8.1.

Do not install in a directory whose name includes blanks

If you install Version 8.1 on a Windows computer on which you have an earlier version of DB2 OLAP Server, and if you accepted the default installation directory name when you installed the earlier version, then some functions in version 8.1 might fail. In earlier versions, the default installation directory name contained blanks, which causes the failure. Additionally, other functions might produce unpredictable results if the name of the installation directory contains underscores, dashes, or other special characters. If you are installing DB2 OLAP Server version 8.1 as a first-time installation, you are prompted for an installation directory. The new default directory for version 8.1 has the correct naming scheme, but if you choose to store DB2 OLAP Server in a different directory, it is strongly recommended that your directory name does not contain blanks, underscores, dashes, or other special characters.

If you install Version 8.1 into a different directory than your Version 7.1 installation, remember to update your environment settings to point to the new installation directory.

The Security Migration Tool is the easiest way to upgrade to version 8.1

If you have an earlier version of DB2 OLAP Server and want to install version 8.1 on a different computer, you can use the Security Migration tool to migrate your applications, data, and user information. See “Security Migration Tool” on page 143 for more information.

Upgrading to version 8.1 on the same Windows computer

If you have an earlier version of DB2 OLAP Server on your Windows computer, and you want to install version 8.1 on that computer, it is recommended that you export your data and application files, uninstall the previous edition, install version 8.1 (in a directory whose name does not contain blanks, underscores, dashes, or other special characters), and then import your application files and data into the new version.

You can install version 8.1 on top of an earlier version, and your existing OLAP applications will be available to the new or upgraded version. but results might be unpredictable. When you install a new version to the same directory, you should take the following steps as a precaution:

- Back up the following files:
 - Your data
 - Essbase.sec
 - license.*
- Stop any OLAP services.

The Relational Storage Manager is not available in version 8.1

The Relational Storage Manager (RSM) is withdrawn from Version 8.1. If you are upgrading from DB2 OLAP Server Version 7.1 or prior, and you have you have applications that use the RSM, you must migrate them to the Multidimensional Storage Manager (MSM) before installing DB2 OLAP Server Version 8.1, or you will lose data. See “Migrate from RSM to MSM before upgrading to Version 8.1” on page 131 for more information.

During installation you are asked what edition and options you purchased. This information is used to build a license key for DB2 OLAP Server.

Installation steps

To install and set up DB2 OLAP Server on Windows:

1. Insert the DB2 OLAP Server CD into the CD-ROM drive to start the installation program. If the program does not start automatically, you can run `setup.exe` from the root directory.
2. Click **Next**. To cancel installation at any time, click **Cancel**. To display Help at any time, press F1.
3. Press **Accept** to accept the Software License Agreement.
4. Next, select the installation options that you want to install, and click **Next**.
5. When prompted, select the edition of DB2 OLAP Server that you purchased, and click **Next**.

If you are performing a Client Only install (such as for AS/400), or if you are installing on a Windows 9X platform, then Personal Edition will be selected by default and this screen will not be presented.

6. On the Select Installation Options screen, you select which server and client components you want. You also have the option of loading an installable image of the clients onto a file server and make them available to your enterprise. To activate this feature, follow the directions in “Setting up client installation from a network file server” on page 50.

7. In the Select Options Purchased screen, select the optional components that you purchased. This panel can be filled in if you are migrating from a previous OLAP Server release and the previous license information can be read. Click **Next**.
8. In the Choose Destination Location panel, enter the drive and directory where you would like to install DB2 OLAP Server, and click **Next**. If you are doing a network installation, network drives will also be displayed in the Drive drop-down menu. IBM strongly recommends that you install DB2 OLAP Server and its features into a new separate directory; do not install into directory in which a product other than DB2 OLAP Server and its features has been installed.
9. Select the type of installation you want: **Typical**, **Compact**, or **Custom**. If you choose the **Typical** or **Compact** options, DB2 OLAP Server installs only a subset of the documentation. To install all the documentation, choose **Custom** during installation and select the books you want. Click **Next**.
10. Enter the user ID and password you want to designate as the DB2 OLAP Server super user, and click **Next**. The user ID must exist on the computer.
11. Select the network protocol for your computer, and click **Next**. DB2 OLAP Integration Server only supports TCP/IP.
12. Enter a name for the folder into which DB2 OLAP Server is installed, or accept the default. Click **Next**.
13. Click next to start copying the files to your computer. You might need to restart your computer after the files are copied.

Security

The `essbase.sec` file stores information about users, passwords for native security, groups, privileges, and applications. Each time that you successfully start the server component, `essbase.exe`, DB2 OLAP Server makes a backup copy of the security file (`essbase.bak`). If you attempt to start the Agent and a password prompt is not displayed or your password is rejected, no backup file is created. You can restore from the last successful startup by copying `essbase.bak` to `essbase.sec`. Both files are in the `bin` directory where you installed the server component.

For information about migrating your security information when you upgrade to Version 8.1, see “Security Migration Tool” on page 143.

Network Protocol Files

Network protocol files implement the TCP/IP or Named Pipes protocol that you select when you install OLAP Server or client software. The network protocol file for server software is `essnet.dll`. Additionally, the client software (for example, Essbase Spreadsheet Add-in) is installed with a separate `essnet.dll`. Network protocol files reside in the `bin` directory with the DB2

OLAP Server program files. If you have more than one server computer and the servers use different network protocols, you must install the client software in multiple directories, one for each network protocol. For example, you can install Spreadsheet Add-in for a Named Pipes server in `n:\ibm\db2olap` and Spreadsheet Add-in for a TCP/IP server in `t:\ibm\db2olap`. The two installations use different `essnet.dll` files.

Installing DB2 OLAP Integration Server on Windows

DB2 OLAP Integration Server is an add-on feature of DB2 OLAP Server. You can install it when you install DB2 OLAP Server, or you can install it separately.

To install the DB2 OLAP Integration Server separately:

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 DB2 OLAP Integration Server meet system requirements.
2. Check for any migration issues. If you have existing applications that use the Relational Storage Manager, you must migrate them to the Multidimensional Storage Manager before installing DB2 OLAP Server Version 8.1.
3. Install DB2 OLAP Server if you have not already done so.
4. Install the server component of DB2 OLAP Integration Server from the DB2 OLAP Server CD-ROM.
5. Configure the Java Runtime Environment; see “Setting up Java on Windows” on page 82 for more information.
6. Install the client component (the Essbase Integration Services Console) from the DB2 OLAP Server CD-ROM.
7. Configure a data source using ODBC. See Chapter 8, “Configuring data sources for DB2 OLAP Integration Server” on page 97. On the computer running DB2 OLAP Integration Server, create Open Database Connectivity (ODBC) connections to an OLAP Metadata Catalog data source and to the relational data source that you use for creating OLAP models and metaoutlines.
8. Create an OLAP Metadata catalog. See Chapter 9, “Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server” on page 105 for more information.
9. Set up the sample applications. See “Samples for DB2 OLAP Integration Server” on page 117.

Installing DB2 OLAP Server Miner on Windows

DB2 OLAP Server Miner, also called OLAP Miner, is a new no-charge feature of DB2 OLAP Server that brings together IBM's data mining and OLAP technologies. OLAP Miner applies a sophisticated data mining algorithm called Deviation Detection to the data stored in OLAP cubes. The algorithm scans slices of cubes, identifying values that deviate from the norm. The results of a data mining run can be used by the business analyst to focus on the interesting or surprising values in context of the source OLAP cube. You can still use the usual slice, dice, and drill operations to explore the data, but now you let the values discovered by OLAP Miner drive your analysis. See the *OLAP Miner User's Guide* for information about what OLAP Miner can do for you.

OLAP Miner requires the DB2 OLAP Server API to be installed on the computer on which you install the OLAP Miner server. Java is a requirement for both DB2 OLAP Server and OLAP Miner.

OLAP Miner has a client component that runs on Windows and a server component that runs on both Windows and AIX. For information about installing the OLAP Miner server component on AIX, see "Installing DB2 OLAP Server Miner on AIX" on page 65.

On Windows, you can install the OLAP Miner client on the same computer as the OLAP Miner server or on another computer.

To install OLAP Miner on Windows:

1. Insert the DB2 OLAP Server installation CD into your CD-ROM drive.
2. Find the `OLAP_Miner\win32` directory on the CD.
3. Run the `setup.exe` file to begin the installation program.
4. The installation program prompts you for the following information:
 - If you are installing the OLAP Miner server, you need
 - The alias name of the OLAP server you will examine with OLAP Miner.
 - The host address (or IP address) of the server component of DB2 OLAP Server. The address is added to the `bin\omclient.cfg` file.
 - If you are installing the OLAP Miner client, you need the host address (or IP address) of the computer on which the OLAP Miner server is installed.

You can set up a silent installation so your users do not have to type responses to the prompts of the installation program. To set up a silent installation:

1. Record your responses to the OLAP Miner installation program prompts. To record your responses, run the following command:

```
setup.exe -r
```

This will record your responses to the installation prompts into a file called `setup.iss`, which is stored in the main Windows directory. On Windows NT and Windows 2000 the file is stored in `clinet`, and on Windows 95 and Windows 98, the file is stored in `C:\Windows`.

2. Copy the `setup.iss` file into a temporary directory on the computer on which you are installing OLAP Miner.
3. To run the silent installation, run the `setup.exe` command using the `-s` parameter (for silent installation) and the `-f1` parameter (which indicates the location of the `setup.iss` file). For example, if you stored the `setup.iss` file in the `C:\TEMP` directory, enter the following command:

```
setup.exe -s -f1C:\TEMP\setup.iss
```

Setting up client installation from a network file server

DB2 OLAP Server Version 8.1 offers your enterprise the option of installing the client components from a shared network file server. The alternative is to install the clients locally, one at a time, directly from the CD.

Installing clients over a network has the following advantages over locally installed clients:

- The users in your enterprise can all use the same level of client.
- You can deploy clients to larger numbers of users at the same time, which is often faster than passing a CD from user to user.
- You can apply FixPaks to the common client code and have your users upgrade together.
- The network clients take up less disk space than locally installed clients.

Although network clients offer the same function as locally installed clients, there are important differences between them:

Locally installed clients

Locally installed clients are installed directly from the CD. They include all the client executable files. The installation program adds Start menu shortcuts, updates the Windows registry, and updates environment variables. Because client operations are completely local and do not load or interpret code over a network, local clients might offer better performance than network clients.

Network clients

Network clients are installed from a file server. They do not include

any executable files, and therefore take up less room and store fewer files than locally installed clients. The installation program adds **Start** menu shortcuts, updates the Windows registry, and updates environment variables. Network clients all use the same language and connect to the server using the same network protocol.

The network client feature installs clients of DB2 OLAP Server and DB2 OLAP Integration Server. OLAP Miner clients are not enabled for network installation; they must be installed directly from the CD.

There are two major steps in setting up a network installation, and they are performed by two different users:

1. The system administrator runs the DB2 OLAP Server installation program to load the client image onto the file server. The computer from which the system administrator does this is called the image owner computer. The client image consists of executable files plus a second installation program that your network client users can run to install the network clients. See “Loading the client image onto the network”.
2. The network client users install the network clients by running the second installation program. See “Installing the clients from the network” on page 53.

Loading the client image onto the network

The instructions in this section are intended for the system administrator, who loads the common client image onto the network.

Before you begin, you need to designate one client computer as the client image owner, and decide which user ID you will use to load the client image:

- The image owner computer can be any computer on the network that satisfies the following criteria:
 - You cannot have locally installed clients or the server components on the image owner computer.
 - The image owner computer should not be one that is likely to be removed from the system. If the image owner computer crashes or is removed from the system, the network client users can continue running the clients, but the system administrator take extra steps to apply FixPaks. See “Maintaining the common clients” on page 54 for more information.
You can have the server components on the same file server as the common client image as long as they are not installed in the same directory.
- The user ID with which the image owner computer accesses the file server must have administrator authority and read/write privileges on the file server.

To load the client image onto the network:

1. Connect to the file server. Whatever drive letter you assign will be the same drive letter that the network client users must use when they connect to the file server. For example, if you designate the file server as the p: drive, the network client users have to connect the file server as their p: drive.
2. Insert the DB2 OLAP Server CD into the CD drive on the computer that you designate as the client image owner computer. If the installation program does not start automatically, run the setup.exe file in the \Win32 directory of the CD.
3. On the IBM DB2 OLAP Server Setup panel, click **Next**.
4. On the Software License Agreement panel, read the licence terms. If you accept them, click **Accept**.
5. On the Select Installation Options screen, select **Client**. A box called **Store executables on network** appears. Click it and then click **Next**.
6. On the Select Options Purchased panel, select the optional features your enterprise has purchased and click **Next**.
7. On the Choose Destination Location screen, select the drive letter of the file server from the **Drive** box.
8. In the **Directory** field, either accept the default value or click **Browse** to select a different path name. Click **Next**.
9. On the Select Installation Type panel, select either **Typical**, **Compact**, or **Custom**. It is recommend that you select **Custom** so you can select which online books are installed with the network client. If you choose **Typical** or **Compact**, the installation program only installs a small subset of the documentation. Click **Next**.
If you selected **Custom**, the Select Components panel appears. Open the Client and Server Documentation item in the tree view and select the documentation you want to make available to the network client users. It is recommended that you select all the documentation the network client users will need. Click **Next**.
10. On the OLAP User Information panel, enter the user ID and password of the super user, which is the user ID that you use to connect the client image owner computer to the file server. This user ID must have administrator authority and read/write privileges on the file server. Click **Next**.
11. On the Select Folder panel, either accept the default name of the folder or enter a new one. Click **Next**.
12. On the Start Copying Files panel, click **Next**. The network client image is copied to the file server.

The next step is to inform the network client users how they can install the network clients. They need the following information:

- The drive letter that you selected when loading the client image onto the file server. The network client users must select the same drive letter when they connect to the file server.
- The directory on the file server that contains the installation program for the network clients.
- The path of the OLAP server components. The path is set in the ARBORPATH variable.

The next section, “Installing the clients from the network”, describes how this information is used when installing the network clients.

Installing the clients from the network

The instructions in this section are intended for the network client user, who installs the DB2 OLAP Server clients from the file server. The examples in this section assume the system administrator installed the network client image on the p: disk and accepted the default value for the installation directory (p:\ibm\db2olap).

If you already installed DB2 OLAP Server directly from a CD, you cannot run this installation program.

Before you begin, you need the following information from your system administrator:

- The drive letter you must use to connect to the file server, and the directory in which the system administrator loaded the installation program.
- The path of the OLAP server components.
- The language of the OLAP clients. The system administrator selects which language the clients are in, and you must select the same language. If you install the clients, your computer must be configured in the same language, or results might be unpredictable.

To install the clients from the network:

1. Connect to the file server where the system administrator loaded the client image package. Assign the same drive letter to the file server that the system administrator used. You must always use the same drive letter when you connect to the file server, to the OLAP clients will not work.
2. Change directories to the location of the installation program, such as the p:\ibm\db2olap\install\ directory.
3. Run the netinst.exe command to start the the installation program.
4. On the Location to Save Files panel, select the directory where you want to install the OLAP clients and click **Next**.
5. On the OLAP Network Installation panel, click **Next**.

6. On the Choose Destination Location panel, enter the path from which you are installing the OLAP clients. It is the same as the location of the `netinst.exe` command.
7. On the OLAP Network Installation Language Selection screen, select the language that your administrator supplies to you.
8. On the Start Copying Files panel, click **Next**. After the installation program completes, reboot your computer.

Whenever you connect to the file server to use the OLAP clients, use the same drive letter you used when you installed them.

The installation program creates a log file called `olap81cli.log` that contains information about the components installed.

To uninstall the clients, the run `netinst.exe` again and select the Remove option. The program removes the shortcuts and updates to the Windows registry and environment variables on the client computer. Uninstalling the network clients from your client computer does not affect any files on the server or the image owner computer.

Maintaining the common clients

IBM FixPaks for DB2 OLAP Server can apply service upgrades to the client image stored on the file server.

The following steps describe how to to apply FixPaks for common clients:

1. The system administrator logs on to the image owner computer.
2. The system administrator connects to the file server using the same user ID under which the common client image was originally loaded onto the file server.
3. The system administrator runs the FixPak installation program.
4. The network client users connect to the file server and run the updated `netinst.exe` file to apply the FixPak to their computers. If there are new client products in the client image, then they will be added to the user's computer. If clients were removed from the file server, then they will be removed from the user's computer.

If the image owner computer crashes or is removed from the system, the network client users can continue running the clients. However, the system administrator cannot apply FixPaks without first taking the following steps:

1. Designate a new client image owner computer.
2. Remove the common client image from the file server.
3. Reinstall the original network client image from the new client image owner computer.
4. Apply the FixPak as described in this section.

Configuring Client and Server for Non-English DB2 OLAP Server

To use non-English versions of DB2 OLAP Server successfully, you must set the ESSLANG variable to be the same on both the client computers and the server computer.

Create the ESSLANG environment variable to point DB2 OLAP Server clients and servers to the correct Uniscape code page (installed in the ESSBASE\LOCALE directory). If no ESSLANG value is set, the default is United States English.

You can create the ESSLANG variable as a user variable or a system variable. If ARBORPATH is defined as a system variable, also create ESSLANG as a system variable. If ARBORPATH is defined as a user variable, create ESSLANG as a user variable.

ESSLANG settings should follow the format ESSLANG=*locale* where *locale* is a supported Uniscape GlobalC locale of the format:

Language_Territory.Code page name@Sort sequence

Examples of ESSLANG Settings:

```
ESSLANG=German_Germany.Latin1@Default
ESSLANG=French_France.Latin1@Default
ESSLANG=Spanish_Spain.Latin1@Spanish
ESSLANG=Italian_Italy.Latin1@Default
ESSLANG=Portuguese_Portugal.Latin1@Default
ESSLANG=Norwegian_Norway.Latin1@Danish
ESSLANG=Swedish_Sweden.Latin1@Swedish
ESSLANG=Danish_Denmark.Latin1@Danish
ESSLANG=Finnish_Finland.Latin1@Finnish
ESSLANG=Japanese_Japan.MS932@Binary
ESSLANG=CanadianFrench_French-Canada.Latin1@Default
ESSLANG=TraditionalChinese_Taiwan.MS950@Binary
ESSLANG=SimplifiedChinese_China.MS936@Binary
ESSLANG=Albanian_Albania.ISO-8859-2@Default
ESSLANG=Arabic_SaudiArabia.ISO-8859-6@Default
ESSLANG=Bulgarian_Bulgaria.ISO-8859-5@Default
ESSLANG=Byelorussian_Belarus.ISO-8859-5@Default
ESSLANG=Catalan_Catalonia.Latin1@Default
ESSLANG=Croatian_Croatia.ISO-8859-2@Croatian
ESSLANG=Czech_CzechRepublic.ISO-8859-2@Czech
ESSLANG=Dutch_Netherlands.Latin1@Default
ESSLANG=Estonian_Estonia.ISO-8859-4@Estonian
ESSLANG=Farsi_Iran.ISO-8859-6@Default
ESSLANG=Greek_Greece.ISO-8859-7@Default
ESSLANG=Hebrew_Israel.ISO-8859-8@Default
ESSLANG=Hungarian_Hungary.ISO-8859-2@Hungarian
ESSLANG=Korean_Korea.MS949@Binary
ESSLANG=Latvian_Latvia.ISO-8859-4@Latvian
ESSLANG=Lithuanian_Lithuania.ISO-8859-4@Lithuanian
ESSLANG=Macedonian_Macedonia.ISO-8859-5@Default
```

```
ESSLANG=Romanian_Romania.ISO-8859-2@Romanian
ESSLANG=Russian_Russia.ISO-8859-5@Default
ESSLANG=Serbian_Yugoslavia.ISO-8859-2@Default
ESSLANG=CyrillicSerbian_Yugoslavia.ISO-8859-5@Default
ESSLANG=Slovak_Slovakia.ISO-8859-2@Slovak
ESSLANG=Slovenian_Slovenia.ISO-8859-2@Slovenian
ESSLANG=Thai_Thailand.MS874@Thai
ESSLANG=Turkish_Turkey.ISO-8859-9@Turkish
ESSLANG=Ukrainian_Ukraine.ISO-8859-5@Ukrainian
ESSLANG=Vietnamese_VietNam.MS1258@Vietnamese
```

Starting DB2 OLAP Server on Windows

DB2 OLAP Server is automatically started as a service on your workstation. If you should need to start it manually by using the command line, follow these steps:

1. At the command prompt, enter:

```
Essbase
```

If an error occurs during startup, check your path and environment settings. If your path and environment settings are correct, ensure that the appropriate communications protocol is installed and running on your Windows workstation.

2. The first time you use DB2 OLAP Server, you are prompted to enter the following information:

Company name

The company name that you want to use in your server license registration.

Your name

The name to use as the DB2 OLAP Server supervisor ID. Use this name for your initial login from the Application Manager.

System password

This password is required whenever you start DB2 OLAP Server. It is also required to gain access to the server from the Application Manager when you use the supervisor account.

Verification of the correctness of the information you entered

Enter Y to verify the entries, or N to retype your entries.

When you enter Y, the information will be saved. At subsequent logins, you will be prompted for the system password only.

DB2 OLAP Server displays this message when the server is started and ready for use:

```
Waiting for Client Requests. . .
```

Automating server startup

On Windows, you can place the OLAP server component in a startup folder so it will start automatically when Windows boots. You can also specify the server password, to avoid having to type it each time the OLAP Server starts.

To automate the OLAP server startup on Windows platforms:

1. To automate the OLAP Server startup on Windows platforms:
 - On Windows NT, select **Start > Settings > Taskbar Start Menu Programs** tab.
 - On Windows 2000, select **Start > Settings > Taskbar & Start Menu Advanced** tab.
2. Click Add and specify the location of the Essbase Agent program file, `essbase.exe`.

If you installed Essbase to the default directory, this is `c:\ibm\db2olap\bin\essbase.exe`. If you want the DB2 OLAP Server system password to be automatically entered each time DB2 OLAP Server starts, type the password after `essbase.exe` on the command line. For example:

```
c:\ibm\db2olap\bin\essbase.exe password
```

where `password` is your DB2 OLAP Server system password.

3. Click **Next**, and then double-click the StartUp or Startup folder.
4. Type Essbase, or your server name, and click **Finish**.

Automating application startup

2 You can automatically start up an application or database. When the OLAP server is started, DB2 OLAP Server starts any application or database that is defined with an automatic startup setting. You define startup settings in Essbase Application Manager (Application Settings dialog box and Database Settings dialog box) or in Essbase Administration Services (Application Properties window and Database Properties window).

Stopping DB2 OLAP Server

To stop the server from any workstation that is connected to it, use one of the following two methods:

- For DB2 OLAP Server, you can use Windows Services:
 1. Click **Start** → **Settings** → **Control Panel** → **Services**
 2. On the Services window, select the Essbase server.
 3. Click **Stop**.
- For DB2 OLAP Server, you can use the SHUTDOWNSERVER command in ESSCMD. See the *Database Administrator's Guide*, and the ESSCMD online help for more information.

Optimizing network throughput on Windows NT

On Windows NT servers, a network setting enables you to optimize the performance of networked applications. DB2 OLAP Server performance improves significantly on Windows NT with this setting enabled.

On Windows 2000, the default system setting is to optimize performance for networked applications. It is not necessary to change this setting.

To enable the Maximize Throughput for Network Applications setting:

1. Exit DB2 OLAP Server.
2. Start the Windows NT Control Panel.
3. Double-click the **Network** icon.
4. Select the **Services** tab.
5. In the **Network Services** list box, double-click **Server Services**. The Server dialog box is displayed. If a message is displayed on your screen indicating that you cannot configure the software component, then your Windows NT computer is probably a Windows NT Workstation configuration, not a Windows NT server configuration, and you cannot change the network settings.
6. Select **Maximize Throughput for Network Applications**.
7. Click **OK**. The **Services** tab of the Network dialog box is displayed.
8. Click **Close**.
9. Restart Windows NT.

Adding additional software or users later

The DB2 OLAP Server installation program prompts you for the edition and options that you have purchased, and uses that information to enable DB2 OLAP Server and add-on components.

If you install a license of DB2 OLAP Server and later want to add more features or additional users, you use one of the following methods:

- Select **Modify Component or License Information** from the **Start → Programs → IBM DB2 OLAP Server 8.1** menu.
- Run the SETUP.EXE program either from the root directory or from the CD-ROM.
- Use the Windows Add/Remove Software utility, which is available from the Control Panel.

Using agent commands

DB2 OLAP Server has a server process called the Agent. When you start the main server program (ESSBASE), the Agent becomes the active terminal, unless you start the server in the background. The Agent acts as a console for high-level server operations. Pressing the Enter key in the Agent terminal provides a list of all available commands.

Note: The Agent commands are available only when the server is started in the foreground.

After you have successfully started the server, install the Application Manager and verify that you can connect to the server from there.

The following table shows the commands that are used to manage DB2 OLAP Server.

When you start DB2 OLAP Server, the Agent window opens. The Agent window is the console for high-level server operations. Press **Enter** to view the following list of all available commands.

Table 11. Agent Commands

Command	Description
START <i>appname</i>	Starts the specified application.
STOP <i>appname</i>	Stops the specified application.
USERS	Displays a list of all users that are connected to the server, the total number of connections, and the number of ports that are available.
LOGOUTUSER <i>username</i>	Disconnects a user from the server and frees a port. This command requires the DB2 OLAP Server password.
PASSWORD	Changes the system password that is required to start DB2 OLAP Server.
VERSION	Displays the server version number.
HELP	Lists all valid commands and their respective functions.
PORTS	Displays the number of ports that are installed on the OLAP server and how many are in use. Number of available ports = number of licensed ports + 1 system administrator port.

Table 11. Agent Commands (continued)

Command	Description
DUMP <i>filename</i>	Dumps information from the DB2 OLAP Server security system to a specified file in ASCII format. If you do not supply a path with the file name, the file is saved to the \ ARBORPATH\bin directory (/ ARBORPATH/bin on UNIX systems). This command requires the DB2 OLAP Server password.
QUIT/EXIT	Shuts down all open applications and stops DB2 OLAP Server.

Chapter 5. Installing on AIX, Solaris Operating Environment, and HP-UX

This chapter tells you how to install server components for DB2 OLAP Server and DB2 OLAP Integration Server Version 8.1 on AIX, Solaris Operating Environment, and HP-UX operating systems.

Make sure you consider the following important changes in how DB2 OLAP Server is installed in version 8.1.

The Security Migration Tool is the easiest way to upgrade to version 8.1

If you have an earlier version of DB2 OLAP Server and want to want to install version 8.1 on the same UNIX computer, you can use the Security Migration tool to migrate your applications, data, and user information. See “Security Migration Tool” on page 143 for more information.

Upgrading to version 8.1 on the same UNIX computer

If you have an earlier version of DB2 OLAP Server on your UNIX computer, and you want to install version 8.1 on that computer, it is recommended that you export your data and application files, uninstall the previous edition, install version 8.1 (in a directory whose name does not contain blanks, underscores, dashes, or other special characters), and then import your application files and data into the new version.

You can install version 8.1 on top of an earlier version, and your existing OLAP applications will be available to the new or upgraded version. but results might be unpredictable. When you install a new version to the same directory, you should take the following steps as a precaution:

- Back up the following files:
 - Your data
 - Essbase.sec
 - license.*
- Stop any OLAP services.

The Relational Storage Manager not available in version 8.1

The Relational Storage Manager (RSM) is withdrawn from Version 8.1. If you are upgrading from DB2 OLAP Server Version 7.1 or prior, and you have you have applications that use the RSM, you must migrate them to the Multidimensional Storage Manager (MSM) before

installing DB2 OLAP Server Version 8.1, or you will lose data. See “Migrate from RSM to MSM before upgrading to Version 8.1” on page 131 for more information.

Installing DB2 OLAP Server on UNIX

The user under which you are installing DB2 OLAP Server must have both write and execute privileges in the directory where you are running the installation. If you install the DB2 OLAP Server as root, then there will be difficulties with other users accessing the server and updating the installation with IBM FixPaks, so it is recommended that you install under a user name other than “root.”

Ensure that you have adequate disk space available on your system. You can check this by running `df-k` to check the percentage of space available on the mount where you are installing DB2 OLAP Server.

During the installation, you will be asked to enter the DB2 OLAP Server license edition and options that you purchased. This information is used to build a license key for DB2 OLAP Server and to enable the products for which you have licenses.

Installation steps for DB2 OLAP Server

1. Log on to the server as a user ID who will be the administrator of the DB2 OLAP Server (do not use root).
2. If you are installing on a Solaris operating system, set the locale to c. When you have finished installing, make sure the locale is still c when you start the server for the first time. Using the correct locale ensures that the administration user ID will be configured correctly.
3. If your workstation has a CD-ROM drive, follow these steps:
 - a. Insert the DB2 OLAP Server CD into your CD-ROM drive.
 - b. Use the `ls` command to find `setup.sh` on the CD. The file name will be either uppercase or lowercase, depending on your operating system.
 - c. Enter `setup.sh` to run the setup script from the DB2 OLAP Server CD. If a multi-threaded environment is not detected, you will receive an error message.
4. If your workstation does not have a CD-ROM drive, you must create a temporary directory for use during the installation. To create a temporary directory, follow these steps:
 - a. Use FTP, NFS, or any similar utility to copy the DB2 OLAP Server installation files into your temporary directory.
 - b. Change to the temporary directory.
 - c. Use the `ls` command to find `setup.sh` in the temporary operating system subdirectory. For example, look in `/temp/aix` for the AIX

installation file. The file name will be either uppercase or lowercase, depending on your operating system.

- d. Enter `setup.sh` to run the setup script from the temporary directory.
5. If your operating system is not compatible with DB2 OLAP server, you will receive a notification from the installation, which will then exit. If a multi-threaded environment is not detected, you will receive an error message.

To install DB2 OLAP Server:

1. Enter the directory to which you want to install DB2 OLAP Server. This can be a pre-existing directory, or a directory that will be created by the install. IBM strongly recommends that you install DB2 OLAP Server and its features into a new separate directory; do not install into directory in which a product other than DB2 OLAP Server and its features has been installed.
2. If you have a current DB2 OLAP Server license in another directory that you want to use for the current installation, enter that directory during installation for licensing reference.
3. You will then be presented with the IBM License Agreement Utility, which will prompt you for the IBM license agreement approval. You will be asked to do the following:
 - a. Select the number that corresponds with the language in which you would like to read the IBM license.
 - b. Press Enter to view the license for approval.
 - c. Press "1" to accept the agreement, "2" to decline it, or "99" to return to the previous screen.

The IBM License Agreement Utility will then complete the licensing. The installation will continue to the next panel if the license was accepted, or exit the installation if the license was declined.
4. Select the license edition that you purchased.
5. Select the license options that you purchased, including the number of users licensed. It is important that you select all of the options that you purchased, as this will enable those options to be used by the DB2 OLAP Server user.
6. Select the products that you want to install.
7. After verifying that the summary is correct, select Continue.
8. Select Install to run the installation.

An Installation Status Report panel appears and provides you with updates as the installation progresses. If there are any errors during the installation process, you will be notified.

The installation file will include a log file (`olapinst.log`) located in `<ARBORPATH>/install`, as well as a response file (`olapisnt.rsp`) that can be

used for silent or unattended installations. The log file will include the license information and a list of the installed products. Additionally, it will include any installation errors if they occurred.

Installing DB2 OLAP Integration Server on UNIX

DB2 OLAP Integration Server is installed as an optional add-on for the DB2 OLAP Server.

To install the DB2 OLAP Integration Server as an add-on component:

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 DB2 OLAP Integration Server meet system requirements.
2. Check for any migration issues. If you have existing applications that use the Relational Storage Manager, you must migrate them to the Multidimensional Storage Manager before installing DB2 OLAP Integration Server Version 8.1.
3. Install DB2 OLAP Server if you have not already done so.
4. Install the server component of DB2 OLAP Integration Server from the DB2 OLAP Server CD-ROM, on Windows NT, 2000, or XP, or on UNIX. For UNIX installation, you must perform additional manual configuration and install the Java Runtime Environment or equivalent included on the installation CD-ROM. See Chapter 6, “Installing and configuring Java” on page 73 for more information.
5. Install the client component (the Essbase Integration Services Console) from the DB2 OLAP Server CD-ROM, on Windows 98, Windows NT, Windows ME, Windows 2000, or Windows XP.
6. Configure a data source using ODBC. See Chapter 7, “Configuring the SQL Interface” on page 85. On the computer running DB2 OLAP Integration Server, create Open Database Connectivity (ODBC) connections to an OLAP Metadata Catalog data source and to the relational data source that you use for creating OLAP models and metaoutlines.
7. Create an OLAP Metadata catalog. See Chapter 9, “Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server” on page 105 for more information.
8. Set up the sample applications. See “Samples for DB2 OLAP Integration Server” on page 117.

Updating the Environment for the Database Client

For the OLAP Integration Server, you need to set the environment variables required for database client software in the login script of the user who runs OLAP Integration Server. These environment settings are required for ODBC access to the databases in the RDBMS that 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 OLAP Integration Server software. For more information about how to set environment variables for the database client, see the database client installation documentation.

To verify that the database client software is set up correctly, log on as the user who runs OLAP Integration Server, and use a database utility to connect to the databases that you use with OLAP Integration Server.

Installing DB2 OLAP Server Miner on AIX

DB2 OLAP Server Miner, also called OLAP Miner, is a new no-charge feature of DB2 OLAP Server that brings together IBM's data mining and OLAP technologies. OLAP Miner applies a sophisticated data mining algorithm called Deviation Detection to the data stored in OLAP cubes. The algorithm scans slices of cubes, identifying values that deviate from the norm. The results of a data mining run can be used by the business analyst to focus on the interesting or surprising values in context of the source OLAP cube. You can still use the usual slice, dice, and drill operations to explore the data, but now you let the values discovered by OLAP Miner drive your analysis. See the *OLAP Miner User's Guide* for information about what OLAP Miner can do for you.

OLAP Miner requires the DB2 OLAP Server API to be installed on the computer on which you install the OLAP Miner server. Java is a requirement for both DB2 OLAP Server and OLAP Miner.

OLAP Miner has a client component that runs on Windows and a server component that runs on both Windows and AIX. For information about installing the OLAP Miner client component on Windows, see "Installing DB2 OLAP Server Miner on Windows" on page 49. The OLAP Miner server is not supported on Solaris Operating System or HP-UX.

To install the server component of OLAP Miner on AIX:

1. Insert the DB2 OLAP Server installation CD into your CD-ROM drive.
2. Find the `OLAP_Miner\aix` directory on the CD.
3. Run the `setup.sh` file to begin the installation program.
4. The installation program prompts you for the following information:
 - The alias name of the OLAP database you will examine with OLAP Miner.
 - The host address (or IP address) of the server component of DB2 OLAP Server. The address is added to the `bin\omclient.cfg` file.

Updating the environment settings

On UNIX, the installation program creates scripts you can use to apply the environment variables you need to run DB2 OLAP Server. You must run these scripts manually.

For all UNIX operating systems

Setting up the environment to run the DB2 OLAP Server

This section describes scripts that are created by the installation program and can be called directly from the `.profile` file in order to activate the environment variables.

To run the scripts, add the following command for each script to run in the `.profile` file:

```
. /full path/script name
```

where *full path* is the path where the script file is stored, and *script name* is the name of the script for your operating system.

After entering the statement, log out and then log back in to activate the new environment variables.

For DB2 OLAP Server

Environment variables for the DB2 OLAP Server (in the `ARBORPATH` directory) are located in the following scripts:

essbaseenv.doc

This file contains information and instructions about the scripts that set up the environment variables for DB2 OLAP Server and Java. For more information about installing Java, see Chapter 6, “Installing and configuring Java” on page 73.

essbaseenv.csh

This script sets up the DB2 OLAP Server environment variables in the C Shell or Bourne Shell.

essbasenv.sh

This script sets up the DB2 OLAP Server environment variables in the Korn shell.

essjava.csh

This script sets up the Java environment variables in the C Shell.

essjava.sh

This script sets up the Java environment variables in the Korn Shell.

You need to install Java before running these scripts to set the environment. For more information about installing Java for DB2 OLAP Server, see

Chapter 6, “Installing and configuring Java” on page 73 and “Updating Java variables for DB2 OLAP Server on UNIX” on page 75.

For DB2 OLAP Integration Server

Environment variables for the DB2 OLAP Integration Server (in the ARBORPATH/is/ directory) are located in the following scripts:

hisjava.csh

This script sets up the Java environment variables in the C Shell for DB2 OLAP Integration Server.

hisjava.sh

This script sets up the Java environment variables in the Korn Shell for DB2 OLAP Integration Server.

You need to install Java before running these scripts to set the environment. For more information about installing Java for DB2 OLAP Integration Server, see Chapter 6, “Installing and configuring Java” on page 73 and “Updating Java variables for DB2 OLAP Integration Server on UNIX” on page 79.

For Solaris Operating Environment only

Run the Bourne shell script `root.sh` from the root account. The script is in the `$ARBORPATH/bin` directory and performs additional installation tasks, such as setting sticky bits on binaries and shared libraries. For example, if you want to log on as the root user and run `root.sh`, type the following commands:

```
$su -  
PASSWORD: (enter your password)  
# cd /home/essbase/bin  
# sh root.sh  
# exit
```

The `root.sh` script asks whether you installed the SQL Interface module. If you installed it, `root.sh` prompts you for the full path and file name of the `libodbc.so` shared library. Enter the full path and file name, such as:

```
libodbc.so/home/essbase/dlls/libodbc.so.
```

For HP-UX only

Enter the following command in the `$ARBORPATH/bin` directory:

```
ln -f -s /opt/IBMdb2/V7.0/lib/libdb2.sl libesssql.sl
```

where `/opt/IBMdb2/V7.0/` is the installation directory for DB2 UDB.

The installation program does not update the environment files `.cshrc` or `.profile`. Because the format of these files varies depending on the shell you use, you need to update the appropriate file for your system manually.

Read the `hyperionenv.doc` text file in your installation directory for the environment setting for the OLAP server.

Note: If you are installing on Solaris, and you plan to use the Cache Memory Locking database setting, also run `root.sh` from the `ARBORPATH/bin` directory.

If you are installing SQL Interface on any UNIX platform, you must take additional steps to set your environment. See *Installing SQL Interface* for details.

Note: After installation and updating the environment, you must log out and log on again to enable the new settings to take effect.

Starting and shutting down the server

You can start the server in the foreground or in the background.

Before you can access the server, you must be logged on to the UNIX server on an account that has at least Read/Write/Execute access to the Essbase directories. Ideally, this is the same account that was used to install DB2 OLAP Server.

Starting the server in the foreground

To start the server in the foreground:

1. From the command prompt, type:
`$ARBORPATH/bin/ESSBASE`

or, if your path includes `$ARBORPATH/bin`, type:
`ESSBASE`

If you encounter any errors during startup, check your path and environment settings.

2. If this is a first-time installation, Essbase prompts you for the following items:
 - **Company name.** The name you type will be embedded in your server license registration.
 - **Your name.** The name you type will be designated as the Essbase system supervisor. Use this name for your initial logon from the Essbase client to the Essbase server.
 - **System password.** The password you choose is required every time you start the Essbase server.
 - You are asked to confirm your choices. Type **1** to verify your entries, or **2** to retype them.
3. If this is not a first-time installation, the system prompts you for the system password that you specified when you first ran the server. The Essbase Agent then displays the following message:

Waiting for Client Requests...

You can now use the Essbase Agent to perform high-level supervisor operations. See *Using Essbase Agent Commands* for a list of available commands.

Starting the server in the background

If you are starting the server for the first time after installing, you must start it in the foreground once before you can start it in the background. See *Starting Essbase in the Foreground*. If you installed Version 8.1 in the same directory as an existing installation, it is not necessary to start the server in the foreground before starting it in the background.

After you have started the server in the foreground and followed the prompts to enter your company name, Essbase administrator user name, and password, type EXIT to stop the server and follow the instructions to start the server as a background process.

To start the server as a background process:

1. If this is a first time installation, you must start Essbase in the foreground.
2. To start the server in the background on UNIX, or on Windows systems utilizing a UNIX-like shell such as MKS, enter the following command at the command prompt of your operating system:

```
$ARBORPATH/bin/ESSBASE password -b &
```

If your path already includes \$ARBORPATH/bin, type this instead:

```
ESSBASE password -b &
```

You can start the server without using the ampersand (&) at the end of the command, but if you do not type the &, the command prompt is not returned after the server is started.

Securing a password script

To hide the system password from the UNIX process listing created by the ps -ef utility on Solaris and HP-UX, you must create a simple shell script and use it to launch the Essbase Agent. Create a shell script named `essbase.secure`, consisting of the following lines:

```
#!/bin/sh
PASS=$1
ESSBASE -b -secure << EOF &
${PASS}
EOF
```

After creating this script, launch the Agent to start it in the background with password secured from process listing details:

```
essbase.secure password
```

Note: On AIX servers, there is no need to hide the Essbase system password because it is hidden automatically.

Shutting down the server

To shut down the server when Essbase is running in the foreground, type EXIT.

To shut down the server from any terminal connected to it, use the SHUTDOWNSERVER command in ESSCMD.

Adding additional software or users later

The DB2 OLAP Server installation program prompts you for the edition and options that you have purchased, and uses that information to enable DB2 OLAP Server and add-on components.

If you install a license of DB2 OLAP Server and later want to add more features or additional users, run the license program from the /bin directory of the DB2 OLAP Server installation directory. The license program will automatically update the license.

Using agent commands

DB2 OLAP Server has a server process called the Agent. When you start the main server program (ESSBASE), the Agent becomes the active terminal, unless you start the server in the background. The Agent acts as a console for high-level server operations. Pressing the Enter key in the Agent terminal provides a list of all available commands.

Note: The Agent commands are available only when the server is started in the foreground.

After you have successfully started the server, install the Application Manager and verify that you can connect to the server from there.

The following table shows the commands that are used to manage DB2 OLAP Server.

When you start DB2 OLAP Server, the Agent window opens. The Agent window is the console for high-level server operations. Press Enter to view the following list of all available commands.

Table 12. Agent Commands

Command	Description
START <i>appname</i>	Starts the specified application.

Table 12. Agent Commands (continued)

Command	Description
STOP <i>appname</i>	Stops the specified application.
USERS	Displays a list of all users that are connected to the server, the total number of connections, and the number of ports that are available.
LOGOUTUSER <i>username</i>	Disconnects a user from the server and frees a port. This command requires the DB2 OLAP Server password.
PASSWORD	Changes the system password that is required to start DB2 OLAP Server.
VERSION	Displays the server version number.
HELP	Lists all valid commands and their respective functions.
PORTS	Displays the number of ports that are installed on the OLAP server and how many are in use. Number of available ports = number of licensed ports + 1 system administrator port.
DUMP <i>filename</i>	Dumps information from the DB2 OLAP Server security system to a specified file in ASCII format. If you do not supply a path with the file name, the file is saved to the \ARBORPATH\bin directory (/ARBORPATH/bin on UNIX systems). This command requires the DB2 OLAP Server password.
QUIT/EXIT	Shuts down all open applications and stops DB2 OLAP Server.

Chapter 6. Installing and configuring Java

This chapter describes how to enable DB2 OLAP Server and DB2 OLAP Integration Server to work with the Java Runtime Environment (JRE) and the Java API for XML Parsing (JAXP).

DB2 OLAP Server 8.1 requires Java for custom-defined macros and custom-defined functions. Both the JRE and the JAXP must be installed on machines running the server component. This includes the Java Virtual Machine (JVM), which is a component of the JRE. If you have already installed a supported version of the JRE, you do not have to reinstall the JRE. However, you must verify that JAXP component is installed, set the `JVMMODULELOCATION` parameter of the `essbase.cfg` file, and set the environment variables.

After the Java components are installed on your system, you can use a Java Development Kit (JDK) of your choice to create your own custom-defined macros and custom-defined functions. A JDK is not supplied with DB2 OLAP Server or DB2 OLAP Integration Server.

Setting up Java on UNIX

In Version 8.1, DB2 OLAP Integration Server also requires Java, but not at the same level used by DB2 OLAP Server. The following table lists the levels of Java you will need:

Table 13. Levels of Java required for various UNIX operating systems.

Product	Operating system	Java Runtime Environment level
DB2 OLAP Server	AIX	1.2.2
	Solaris	1.3.0
	HP-UX	1.2.2
DB2 OLAP Integration Server	AIX	1.3.0
	Solaris	1.3.0_02
	HP-UX	1.3.0.01

To enable Java to work with both DB2 OLAP Server and DB2 OLAP Integration Server on UNIX, perform the following steps:

1. Install the Java Runtime Environment (JRE) and the Java API for XML Parsing (JAXP).

2. Set the Java environment variables for DB2 OLAP Server.
3. Set the JVMMODULELOCATION parameter to give DB2 OLAP Server access to JAXP. This parameter is set in the `essbase.cfg` file, so you can designate a specific installation of the Java Runtime Environment for use with DB2 OLAP Server. This setting is particularly useful if you have multiple versions of Java installed on the machine that is running the OLAP Server component.
4. Set advanced configuration options for Java, if necessary.
5. If you use DB2 OLAP Integration Server, set the Java environment variables for it.

The JRE installation packages and installation procedures for each of the supported platforms are substantially different. README files and installation instructions from the JRE providers are included in the `.txt` or `.htm` files that come with each package. Review these instructions carefully. You may need to refer to the web site of the JRE provider for additional information about installing the JRE.

After following the instructions for setting up Java on UNIX, if you are still having difficulties getting DB2 OLAP Server to work with Java, it may be necessary to set startup options for the Java Virtual Machine (JVM). You can make DB2 OLAP Server use specific JVM startup options when it loads the JVM by setting the environment variables `ESS_JVM_OPTION1` through `ESS_JVM_OPTION9`.

Installing Java on UNIX

To install the Java Runtime Environment code follow the steps below, regardless of which UNIX operating system you are using:

1. Log in using an account that has root privileges.
2. Copy the `ARBORPATH/bin/jrte_<platform>_ess.tar` file into the `/tmp` directory.
3. Extract the Java package using the `tar -xvf jrte_<platform>_ess.tar`. On AIX, a `.gz` file is extracted from the `jrte` tar file, which can then be opened using `gunzip` on UNIX, or `Winzip` or `PKZip` on Windows, to extract the `.rte` package to be installed.
4. If you are using DB2 OLAP Integration Server:
 - a. Copy the `jre130-20010330.tar_` file into the `.tmp` directory.
 - b. Extract the Java package using the `tar -xvf jre130-20010330.tar` file. This creates a directory called `jre`.
5. Once extracted, install the files into their sections.

Installing Java on AIX

AIX requires that the following packages are installed prior to installing this package. You must have root privileges to install these packages:

- X11.adt.lib 4.3.3.0
- X11.adt.motif 4.3.3.1
- bos.adt.include 4.3.3.0
- bos.adt.prof 4.3.3.0

To install the JRE for DB2 OLAP Server, run the following command:

```
installp java_devX.rte
```

The next step is to set the Java environment variables for your operating system.

Installing Java on Solaris

To install Java on Solaris, update the permissions on the .bin file:

```
chmod +x jXrel_XXXXX-solsparc.bin
```

where XXXXX represents the specific version number. Then run the self-extracting binary:

```
jXrel_XXXXX-solsparc.bin
```

The next step is to set the Java environment variables for your operating system.

Installing Java on HP-UX

To install the Java Runtime Environment (JRE), run the following command:

```
swinstall rte_XXXXX.depot
```

where XXXXX represents the specific version number.

The next step is to set the Java environment variables for your operating system.

Updating Java variables for DB2 OLAP Server on UNIX

On UNIX operating systems, you must update environment variables to enable DB2 OLAP Server to use the Java Runtime Environment. *JREHOME* represents the installation directory for the JRE.

After setting the following environment variables, log out of the system and then back on to activate the new settings. Run the following command to verify that the correct version of Java is set before starting DB2 OLAP Server:

```
>java -fullversion
```

For DB2 OLAP Server, you will see:

- On HP-UX: Java full version 1.2.2.05-))/08/14-PA_RISC2.0
- On AIX: Java full version J2RE 1.2.2 IBM build ca122-19991217

- On Solaris: Java full version 1.3.0

Scripts are provided to set up the environment variables to match the correct level of JRE. Table 2 shows which script to use with each product.

Table 14. Environment variables to match levels of JRE for DB2 OLAP Server

Product	Shell	Script
DB2 OLAP Server	Korn or Bourne	essjava.sh
DB2 OLAP Server	CShell	essjava.csh

Updating Java environment variables on AIX

On AIX, you must update the LIBPATH setting to include the *JREHOME/bin/* and the *JREHOME/bin/classic* directories. These LIBPATH directory entries must be added to the beginning of the LIBPATH environment variable.

If you use the C shell, add the following text to the *.cshrc* file:

```
setenv JREHOME "/usr/java_dev2/jre"
setenv LIBPATH "${JREHOME}/bin:${JREHOME}/bin/classic:${LIBPATH}"
setenv LD_LIBRARY_PATH
"${JREHOME}/bin:${JREHOME}/bin/classic:${LD_LIBRARY_PATH}"
set path=( ${JREHOME}/sh /usr/java_dev2/sh/ $path )
setenv CLASSPATH
"${ARBORPATH}/java/jaxp.jar:${ARBORPATH}/java/parser.jar:${CLASSPATH}"
```

If you use the Bourne or Korn shell, add the following text to the *.profile* file:

```
export JREHOME=/usr/java_dev2/jre
export LIBPATH=$JREHOME/bin:$JREHOME/bin/classic:$LIBPATH
export
LD_LIBRARY_PATH=$JREHOME/bin:$JREHOME/bin/classic:$LD_LIBRARY_PATH
export PATH=$JREHOME/sh:/usr/java_dev2/sh:$PATH
export
CLASSPATH=$ARBORPATH/java/jaxp.jar:$ARBORPATH/java/parser.jar:$CLASSPATH
```

For both the C and the Bourne or Korn shell, note that *\$JREHOME/bin* must be added to the beginning of LIBPATH.

Updating Java environment variables on Solaris Operating System

On Solaris, you must set the ESS_JVM_OPTION option. Note that JREHOME must be at the beginning of the PATH statement, before /usr/bin.

If you use the C shell, add the following text to the *.cshrc* file:

```
setenv ESS_JVM_OPTION1 "-XX: +AllowUserSignalHandlers"
setenv JREHOME "${ARBORPATH}/bin/j2re1_3_0"
set path=( ${JREHOME}/bin $path)
```

```
setenv CLASSPATH
"${ARBORPATH}/java/jaxp.jar:${ARBORPATH}/java/parser.jar:${CLASSPATH}"
setenv LD_LIBRARY_PATH "${JREHOME}/bin:${LD_LIBRARY_PATH}"
```

If you use the Bourne or Korn shell, add the following text to the `.profile` file:

```
ESS_JVM_OPTION1=-XX:+AllowUserSignalHandlers
export ESS_JVM_OPTION1
JREHOME=$ARBORPATH/bin/j2rel_3_0
export JREHOME
PATH=$JREHOME/bin:$PATH
export PATH
export
CLASSPATH=$ARBORPATH/java/jaxp.jar:$ARBORPATH/java/parser.jar:$CLASSPATH

export LD_LIBRARY_PATH=$JREHOME/bin:$LD_LIBRARY_PATH
```

Updating Java environment variables on HP-UX

On HP-UX, you must update the `SHLIB_PATH` setting to include the `JREHOME/lib/PA_RISC/` and the `JREHOME/lib/PA_RISC/classic` directories.

Attention: The directory, `PA_RISC`, may be `PA_RISC2.0`, depending on the processor used by the computer running HP-UX. Additionally, due to a problem with the Just In Time compiler in Release 1.2.2 of the JRE for HP-UX, you must also set an `ESS_JVM_OPTION` option.

Some of these environment variable settings are coded assuming the variables are already set on your HP-UX system, and they add new values to your existing settings. HP-UX does not automatically create these variables if they do not exist. If you receive an error message about environment variables not being set, you must manually create the variables. See your HP-UX documentation for more information about setting environment variables.

If you use the C shell, add the following text to the `.cshrc` file:

```
setenv JREHOME "/opt/java1.2/jre"
setenv SHLIB_PATH
"${JREHOME}/lib/PA_RISC:${JREHOME}/lib/PA_RISC/classic:${SHLIB_PATH}"
set path=( $path ${JREHOME}/bin )
setenv ESS_JVM_OPTION1 "-Djava.compiler=NONE"
setenv CLASSPATH
"${ARBORPATH}/java/jaxp.jar:${ARBORPATH}/java/parser.jar:${CLASSPATH}"
```

If you use the Bourne or Korn shell, add the following text to the `.profile` file:

```
JREHOME=/opt/java1.2/jre
export JREHOME
SHLIB_PATH=$JREHOME/lib/PA_RISC:$JREHOME/lib/PA_RISC/classic:$SHLIB_PATH;
export SHLIB_PATH
PATH=$PATH:$JREHOME/bin;
```

```

export PATH
ESS_JVM_OPTION1=-Djava.compiler=NONE;
export ES_JVM_OPTION1
CLASSPATH=$CLASSPATH:$ARBORPATH/java/jaxp.jar:$ARBORPATH/java/parser.jar;

export CLASSPATH

```

Giving DB2 OLAP Server access to JAXP on UNIX

The Java API JAXP for XML parsing is required by DB2 OLAP Server to enable Java-based features such as custom-defined macros and custom-defined functions. JAXP is installed on your system when you install the JRE. In most situations, installation and configuration of the JRE for DB2 OLAP Server is complete when you install the JRE. However, if you have installed the JRE but your server cannot detect JAXP, make sure the following file has been copied to JREHOME/lib/ext:

```
/ARBORPATH/java/*.jar
```

Then set the JVMMODULELOCATION to enable Hyperion Essbase to find and load Java. JVMMODULELOCATION is a setting in the `essbase.cfg` file that enables you to designate a specific installation of the Java Runtime Environment for use with DB2 OLAP Server. This setting is particularly useful if you have multiple versions of Java installed on the computer running the OLAP server component.

To set the JVMMODULELOCATION parameters, you must specify the full path and filename of the Java Virtual Machine (JVM) library. The name and location of this library varies, depending on the JRE version and operating system you are using.

To set JVMMODULELOCATION:

1. On the computer that is running DB2 OLAP Server, change directories to the `ARBORPATH/bin` directory.
2. In the `essbase.cfg` file, add a new line to the end of the file and type the following text:

```
JVMMODULELOCATION FullpathAndFilenameOfJvmLibraryFile
```

The following examples show how to set JVMMODULELOCATION on the UNIX systems supported by DB2 OLAP Server. Do not include spaces in the path and filename. Do not enclose the parameter in quotation marks. A semicolon is not required at the end of the line.

On AIX (JRE 1.2.2): make sure you enter this setting in mixed case, as follows:

```
JvmModuleLocation /usr/java_dev2/jre/bin/classic/libjvm.a
```

On Solaris (JRE 1.3): The path of the JVM library file is:

```
/usr/j2se/jre/lib/sparc/client/libjvm.so
```

On Solaris, do not set the `JVMMODULELOCATION` to point directly to the `libjvm.so` file. If you do, the JVM will crash when DB2 OLAP Server attempts to load it. Instead, set `JVMMODULELOCATION` to the symbolic link for this file. The default location for the symbolic link is:

```
/olaptest/essbase7/bin/j2re1_3_0/lib/sparc/libjvm.so
```

where `/olaptest/essbase7/bin` is the `ARBORPATH` in this example.

On HP-UX (JRE 1.2): Make sure you enter this setting in mixed case, as follows:

```
JvmModuleLocation /opt/java1.2/jre/lib/PA_RISC/classic/libjvm.sl
```

The directory, `PA_RISC`, might be `PA_RISC2.0`, depending on the processor used by the computer running HP-UX.

Updating Java variables for DB2 OLAP Integration Server on UNIX

On UNIX operating systems, you must update environment variables to enable DB2 OLAP Integration Server to use the Java Runtime Environment.

After setting the environment variables below, log out of the system and back in to activate the new settings. Run the following command to verify that the correct version of Java is set before starting Essbase:

```
>java_fullversion
```

For DB2 OLAP Integration server, you will see:

- **On HP-UX:** java full version "JavaVM-1.3.0.01"
- **On AIX:** java full version "1.3.0"
- **On Solaris:** java full version "1.3.0_02"

Scripts are provided to set up the environment variables to match the correct level of JRE. The following table shows which script to use with each product.

Table 15. Environment variables to match levels of JRE for DB2 OLAP Integration Server

Product	Shell	Script
DB2 OLAP Integration Server	Korn or Bourne	hisjava.sh
DB2 OLAP Integration Server	CShell	hisjava.csh

Updating environment variables for DB2 OLAP Integration Server on AIX

On AIX, you must update the `LIBPATH` setting to include the `JREHOME/bin/` and the `JREHOME/bin/classic` directories. These directory entries must be added to the beginning of the `LIBPATH` environment variable.

If you use a Bourne or Korn C shell, add the following additional variables from the `hisjava.sh` file:

```
export TEMP=/tmp
export JAVA_HOME=$ISHOME/jre
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=$JAVA_HOME/lib:$CLASSPATH
export LIBPATH=$JAVA_HOME/lib:$LIBPATH
```

If you use the C shell, add the following additional variables from the `hisjava.sh` file:

```
setenv TEMP "/tmp"
setenv JAVA_HOME "${ARBORPATH}/is"
setenv PATH "${JAVA_HOME}/bin:${PATH}"
setenv CLASSPATH "${JAVA_HOME}/lib:${CLASSPATH}"
setenv LIBPATH "${JAVA_HOME}/lib:${LIBPATH}"
```

Updating environment variables for DB2 OLAP Integration Server on Solaris

On Solaris, you must set the `ESS_JVM_OPTION` option. Note that `JREHOME` must be at the beginning of the `PATH` statement, before `/usr/bin`.

If you use Bourne or Korn C shell, add the following additional variables from the `hisjava.sh` file:

```
export TEMP=/tmp
export JAVA_HOME $ARBORPATH/is/bin/j2re1_3_0_02
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=$JAVA_HOME/lib:$CLASSPATH
export LD_LIBRARY_PATH=$JAVA_HOME/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=$JAVA_HOME/lib:$LD_LIBRARY_PATH
```

If you use C shell, add the following additional variables from the `hisjava.sh` file:

```
setenv TEMP "/tmp"
setenv JAVA_HOME "${ARBORPATH}/is"
setenv PATH "${JAVA_HOME}/bin:${PATH}"
setenv CLASSPATH "${JAVA_HOME}/lib:${CLASSPATH}"
setenv LD_LIBRARY_PATH "${JAVA_HOME}/lib:${LD_LIBRARY_PATH}"
```

Updating environment variables for DB2 OLAP Integration Server on HP-UX

On HP-UX, you must update the `SHLIB_PATH` setting to include the `JREHOME/lib/PA_RISC/` and the `JREHOME/lib/PA_RISC/classic` directories.

Attention: The directory, `PA_RISC`, may be `PA_RISC2.0`, depending on the processor used by the computer running HP-UX. In addition, due to a problem with the Just In Time compiler in release 1.2.2 of the JRE for HP-UX, you must also set an `ESS_JVM_OPTION` option.

Some of these environment variable settings are coded assuming the variables are already set on your HP-UX system, and they add new values to your existing settings. HP-UX does not automatically create these variables if they do not exist. If you receive an error message about environment variables not being set, you must manually create the variables. See your HP-UX documentation for more information about setting environment variables.

If you use a Bourne or Korn C shell, add the following additional variables from the `hisjava.sh` file:

```
export TEMP =/tmp
export JAVA_HOME=/opt/java1.3/jre
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=$JAVA_HOME/lib:$CLASSPATH
export SHLIB_PATH=$JAVA_HOME/lib$SHLIB_PATH
```

If you use a C shell, add the following additional variables from the `hisjava.sh` file:

```
setenv TEMP " /tmp"
setenv JAVA_HOME "${ARBORPATH}/is"
setenv PATH "${JAVA_HOME}/bin:${PATH}"
setenv CLASSPATH "${JAVA_HOME}/lib:${CLASSPATH}"
setenv SHLIB_PATH
      "${JAVA_HOME}/lib:${ISHOME}/esslib:${ISHOME}/islib:${SHLIB_PATH}"
```

Finding which shell you are using

To find out which shell is currently running on your UNIX system, run this command:

```
> echo $SHELL
```

Following is a list of various responses you might get, depending on the kind of shell you are using:

- `/usr/bin/sh` Bourne shell
- `/usr/bin/rsh` Restricted Bourne shell
- `/usr/bin/jsh` Bourne shell with job control
- `/usr/bin/ksh` Korn shell
- `/usr/bin/rksh` Restricted Korn shell
- `/usr/bin/csh` C shell

Another way to find out which shell you are currently using is to examine the `etc/passwd` file. You can see this by running the `cat` command, which is as follows:

cat /etc/passwd

Setting up Java on Windows

On Windows, both DB2 OLAP Server and DB2 OLAP Integration Server require JRE 1.3. To enable Java to work with both DB2 OLAP Server and DB2 OLAP Integration Server, you must:

1. Set up the environment variables for the Java Runtime Environment (JRE).
2. Set the JVMMODULELOCATION parameter in the `essbase.cfg` file to that you can designate a specific installation of the JRE for use with DB2 OLAP Server. This setting is particularly useful if you have multiple versions of Java installed on the computer running the DB2 OLAP Server component.

After the Java components are installed on your system, you can use a Java Development Kit (JDK) of your choosing to create your own custom-defined macros and custom-defined functions. A JDK is not supplied with DB2 OLAP Server or DB2 OLAP Integration Server.

Setting Java environment variables on Windows

The Java Runtime Environment (JRE) is required by DB2 OLAP Server to enable Java-based features, such as custom-defined macros and custom-defined functions. The JRE must be installed on the computer running the DB2 OLAP Server component.

The DB2 OLAP Server installation program copies the files to your workstation. To complete the installation of Java, you need to update your PATH statement:

On Windows 2000 and Windows XP, add this variable to your PATH statement:

```
%JREHOME%\bin
```

For example, you can set your PATH to the following value:

```
PATH=%JREHOME%\bin;%PATH%
```

On Windows NT, add this variable to your PATH statement:

```
C:\ibm\db2olap\java\jre\bin
```

For example, you can set your PATH to one of the the following values:

```
PATH= C:\ibm\db2olap\java\jre\bin;%PATH%
```

```
PATH= %ARBORPATH%\java\jre\bin;%PATH%
```

If you have already installed a supported version of the Java Runtime Environment, you do not have to reinstall it. However, you must verify that

the JAXP component is installed. You must also set the JVMMODULELOCATION parameter to enable DB2 OLAP Server to find and load Java.

Setting JVMMODULELOCATION on Windows

To set the JVMMODULELOCATION parameters, you must specify the full path and filename of the Java Virtual Machine (JVM) library. On Windows systems (JRE 1.3), the default location and name of the JVM Library File is:

```
c:\ibm\db2olap\java\jre\bin\hotspot\jvm.dll
```

where

```
c:\ibm\db2olap\
```

is %ARBORPATH% in this example.

To set the JVMMODULELOCATION:

1. On the computer that runs the DB2 OLAP Server component, change to the ARBORPATH\bin directory.
2. Locate the essbase.cfg file in this directory and open the file in a text editor. If the file does not exist, create and open a new text file named essbase.cfg.
3. In the essbase.cfg file, add a new line to the end of the file and type JVMMODULELOCATION, followed by the full path and file name of the JVM library file. In the following example, the statement should be all on one line:

```
JVMMODULELOCATION c:\ibm\db2olap\  
java\jre\bin\hotspot\jvm.dll
```

4. Save the changed text file as essbase.cfg.
5. Close the file and exit the text editor.

Chapter 7. Configuring the SQL Interface

This chapter explains how to configure the SQL Interface.

As of DB2 OLAP Server Version 8.1, the SQL Interface is no longer installed separately. Instead, it is installed with the OLAP server component. You need to purchase the Tools Bundle to use the SQL Interface, and indicate that you purchased it in the installation program.

After you install the DB2 OLAP Server and the SQL Interface and set your environment variables, you must configure your data source. You may need to configure your ODBC driver. For this information, see the *SQL Interface Guide*.

For installations on UNIX, this chapter tells you what to do after you install and before you link your drivers to SQL Interface.

Configuring SQL Interface on Windows

You can configure The SQL Interface on Windows from the Essbase CD, either at the same time you install the Essbase OLAP Server or after you have installed the Essbase OLAP Server.

To configure SQL Interface on Windows:

1. Make sure the relational database management system (RDBMS) environment is set up correctly. You must be able to connect to the RDBMS from the Essbase server machine before SQL Interface can work. Confirm that:
 - a. The RDBMS server is installed and running. You must set the environment variables required to run the RDBMS. For more information, see your RDBMS documentation.
 - b. The RDBMS database client software is installed on the machine where Essbase is (or will be) installed; for example, the correct version of Client Application Enabler (CAE) for DB2 is installed, or the correct version of Open Client for Sybase is installed.
 - c. You can connect to the RDBMS.
 - d. You can issue a query to the RDBMS using an RDBMS client utility.
2. Configure the ODBC data source. Before you can access relational data using SQL Interface, you must use the ODBC Administrator (odbcad32.exe) to tell DB2 OLAP Server what data sources are available and which drivers it should use. For detailed instructions, see the *SQL Interface Guide*.

Additionally, On Windows NT and Windows 2000, each MERANT ODBC driver has its own help file. You can access it using the ODBC Administrator. The MERANT ODBC drivers come with a document called MERANT DataDirect Connect ODBC Reference. This reference is in PDF format for online viewing and printing in Adobe Acrobat Reader (Version 3.0.1 or later).

If you accept the default during installation, the installation program places all SQL Interface program files in the C:\ibm\db2olap\bin directory on your Windows computer. This includes `esssql.dll`, the ODBC driver files and their corresponding help (`.hlp`) files, and `.bnd` and `.lic` files. If you choose to install SQL Interface in a different directory from the default, the installation program stores the files in the bin directory in the appropriate path.

The SQL Interface installation program also installs ODBC driver files and their corresponding help and `.pdf` files in the `ARBORPATH/ODBC/Merant40/Docs` directory.

Configuring SQL Interface on UNIX

Install Essbase SQL Interface on UNIX (AIX, HP-UX, and Solaris) from the Essbase CD-ROM after you have installed the Essbase OLAP Server. Before installing SQL Interface, ensure that your RDBMS environment is set correctly. After installing SQL Interface, set up the environment and the ODBC drivers.

The following workflow is an overview of tasks required to effectively install Essbase SQL Interface on UNIX.

Workflow for configuring SQL Interface on UNIX

To install SQL Interface on UNIX:

1. Make sure the RDBMS environment is set up correctly before installing SQL Interface. See “Setting up RDBMS for the SQL Interface on UNIX” on page 87.
2. Set up the ODBC driver you plan to use. Steps vary depending on platform/driver configurations. For examples, see “Setting up ODBC drivers before linking them to SQL Interface” on page 88.
3. Run `inst-sql.sht` link SQL Interface to your ODBC driver. See “Linking SQL Interface to the ODBC driver on UNIX” on page 90
4. Complete post-installation tasks, including setting the library path and configuring the `.odbc.ini` file. See “After you link SQL Interface to your ODBC Driver on UNIX” on page 91.
5. To learn more about SQL Interface and how to configure your ODBC data source, see “Configuring the data source and ODBC driver” on page 94.

Setting up RDBMS for the SQL Interface on UNIX

You must be able to connect to the RDBMS from the Essbase server machine before SQL Interface can work.

To set the RDBMS environment variables, make sure:

1. The RDBMS server is installed and running. You must set the environment variables required to connect to the RDBMS.

Examples of Setting RDBMS Environment Variables (Solaris):

For DB2 Universal Database:

- Set DB2INSTANCE and INSTHOME.
- Make sure that LD_LIBRARY_PATH contains the DB2 library path (\$INSTHOME/sql1lib/lib).
- Make sure that the path contains the DB2 bin (\$INSTHOME/sql1lib/bin).

For Informix:

- Set INFORMIXDIR and INFORMIXSERVER.
- Make sure that LD_LIBRARY_PATH contains the Informix library path (\$INFORMIXDIR/lib).
- Make sure that the path contains the Informix bin (\$INFORMIX/bin).

For Oracle:

- Set ORACLE_HOME and ORACLE_SID.
- Make sure that LD_LIBRARY_PATH contains the Oracle library path (\$ORACLE_HOME/lib).
- Make sure that the path contains the Oracle bin (\$ORACLE_HOME/bin).

For Sybase:

- Set SYBASE and DSQUERY.
- Make sure that LD_LIBRARY_PATH contains the Sybase library path (\$SYBASE/lib).
- Make sure that the path contains the Sybase bin (\$SYBASE/bin).

Library path variable names vary across UNIX platforms. For more information, see your RDBMS documentation.

2. The RDBMS client software is installed on the machine where the Essbase OLAP Server is (or will be) running.
3. You can connect to the RDBMS.
4. You can issue a query to the RDBMS using an RDBMS client utility.

See the “Supported ODBC drivers and relational data sources” on page 39 for information on tested and supported SQL Interface platforms and ODBC drivers.

You must also set the environment variables required to run your RDBMS.

Directories and files created

The SQL Interface installation program installs files into the `$ARBORPATH/bin` and `$ARBORPATH/d11s` directories.

The SQL Interface installation program:

- Copies the `odbc.ini` file into the `$ARBORPATH/bin` directory on your UNIX server machine. This tells Essbase which drivers are installed.

If you accepted the default directory during installation, `$ARBORPATH` is `/home/hyperion/essbase`.

- Installs MERANT ODBC drivers files to an `$ARBORPATH/d11s` directory on your UNIX server machine.
- Installs the SQL Interface files to the `$ARBORPATH/bin` directory. These include:
 - On AIX: `$ARBORPATH/bin`
 - On HP-UX: `libesssql.insolv.sl`
 - On Solaris: `libesssql.so.1`

When the OLAP server component is installed on UNIX, it creates:

- The SQL Interface library file:
 - On AIX: `libesssql.a`
 - On HP-UX: `libesssql.sl`
 - On Solaris: `libesssql.so`
- A stub ODBC drivers file:
 - On AIX: `libesssql.arbor.a`
 - On HP-UX: `libesssql.arbor.sl`
 - On Solaris: `libesssql.arbor.so.1`

After you install SQL Interface, you need to run the shell script `inst-sql.sh`, which creates a symbolic link between the SQL Interface library file and the MERANT ODBC drivers file. See “Linking SQL Interface to the ODBC driver on UNIX” on page 90.

Setting up ODBC drivers before linking them to SQL Interface

After you install SQL Interface on AIX, HP-UX, or Solaris, you must set up your ODBC driver before you link SQL Interface to your ODBC driver by running `inst-sql.sh`.

To set your environment so that you can link SQL Interface to the driver you chose, complete the steps after you run `setup.sh`, but before you run `inst-sql.sh`. The steps you take depend on the driver and platform you use. This section includes examples on how to set up the drivers.

For information about setting up MERANT Oracle or Informix drivers, see the MERANT DataDirect Connect ODBC Reference. Also see Oracle or Informix documentation.

Setting up the MERANT DB2 Driver on AIX or Solaris

To set the environment so that you can link SQL Interface to the MERANT DB2 driver, complete the following steps after you run `setup.sh`, but before you run `inst-sql.sh`:

1. Log on as the instance owner, and make sure you have Read or Read/Write access to `$ARBORPATH`.
2. Set the DB2 environment variable, for example, `INSTHOME`, to point to the home directory of your DB2 instance. An instance corresponds to one or more databases. For every database you want to connect to, you must ensure that an instance corresponding to that database exists.
3. Set the environment variable `DB2INSTANCE` to point to the instance name, and set `LIBPATH`.

Here is a Bourne shell example:

```
DB2INSTANCE=inst1
LIBPATH=/lib:/usr/lib:/home/essbase/bin:$INSTHOME/sql/lib/lib:
export INSTHOME DB2INSTANCE LIBPATH
```

where `INSTHOME` is the environment variable that you defined in step 2, and `inst1` is the instance name.

For more information about setting environment variables for the MERANT DB2 driver, see the *MERANT DataDirect Connect ODBC Reference*.

Setting up the IBM DB2 Driver on AIX

To set the environment so that you can link SQL Interface to the IBM DB2 driver, complete the following steps after you run `setup.sh`, but before you run `inst-sql.sh`:

1. Make sure the relational client software for DB2 is installed on the same AIX machine that you are using for SQL Interface.
2. Create a configuration file, `esssql.cfg`. An example is provided below:

```
[
Description "IBM DB2 Call Level Interface"
DriverName db2.o
Database 1
Password 1
UserId 1
SingleConnection 0
UpperCaseConnection 0
IsQEDriver 0
]
```

3. Set the environment variable `INSTHOME` to point to the home directory of your DB2 instance.

4. Define DB2 instance names for SQL Interface. An instance corresponds to one or more databases. For every database you want to connect to, make sure that an instance corresponding to that database exists.
5. Log on as the DB2 OLAP Server system administrator account. This is the account used to install and run DB2 OLAP Server. You should have Read or Read/Write access to the driver manager library, `libodbc.a`, in `INSTHOME/sql1lib/odbc/lib/lib`.
6. Set the environment variables `DB2INSTANCE` and `LIBPATH` as in the following Bourne shell example:

```
DB2INSTANCE=inst1
LIBPATH=/lib:/usr/lib:/home/essbase/bin:$INSTHOME/sql1lib/lib:
export INSTHOME DB2INSTANCE LIBPATH
```

where `INSTHOME` is the environment variable that you defined in 3 on page 89, and `inst1` is the instance name.

Linking SQL Interface to the ODBC driver on UNIX

After you have completed the following actions, you are ready to link SQL Interface to your ODBC driver by running `inst-sql.sh`:

- Installed and configured your RDBMS.
- Installed SQL Interface.
- Set up the ODBC driver you selected during installation

Running `inst-sql.sh` ensures that server processes can access the software necessary for SQL Interface. The installation program, which is launched by `setup.sh`, places `inst-sql.sh` in the `$ARBORPATH` directory. The default is `/home/hyperion/essbase`.

The `inst-sql.sh` file uses the `odbc.ini`, `odbcinst.ini`, and `template` (extension `.tmp1`) files in the `$ARBORPATH/setup` directory to create a set of files that point DB2 OLAP Server to the correct driver product.

The script handles the following tasks for you:

- Ensures root privileges are not accidentally granted to a user-level script.
- Checks for the availability of the OLAP server.
- Checks for the availability to the OLAP server of supported database systems.
- Provides default environment extensions as shell files in the `$ARBORPATH/bin` directory.
- Provides initialization files in the `$ARBORPATH/bin` directory for access by the driver software.
- Creates and dynamically compiles an extra library if needed.

- Generates the shell scripts `odbc.csh` for the C shell, and `odbc.sh` for the Bourne or Korn shell. These files contain commands that update the library path to specify the correct environment variables for Essbase. After you run `inst-sql.sh` to link SQL Interface to the ODBC driver, you can optionally use these scripts to update your library path, or you can do it manually. See “Setting the library path on UNIX”.

To run `inst-sql.sh`:

1. Using the system administrator account (this is the account used to install and run DB2 OLAP Server), enter the following command:

```
sh inst-sql.sh
```
2. Supply information when the script prompts you for it.
 When the script finishes, the command prompt is displayed.
3. Type the following command:

```
exit
```

After you link SQL Interface to your ODBC Driver on UNIX

Now that you have linked SQL Interface to your ODBC driver, you are ready to complete the following post-installation tasks:

- Set the library path, either manually or by using the `odbc.sh` or `odbc.csh` shell scripts. See “Setting the library path on UNIX”.
- If you are using the MERANT DB2 driver, bind files in the `/ESSBASE/bin` directory. See “Binding files for the MERANT DB2 driver on AIX or Solaris” on page 93.
- Complete the configuration of your ODBC drivers by editing the `.odbc.ini` file. Some examples are provided in “Editing `.odbc.ini` driver files” on page 93.
- Configure the ODBC data source and (if applicable) configure your non-MERANT ODBC driver.

Setting the library path on UNIX

On UNIX, you must add the `$ARBORPATH/dlls` directory to the library path before you can use SQL Interface. If your installation of SQL Interface is an upgrade to a preexisting installation, you may not have to reset the library path.

Alternatively, you can use shell scripts created by `inst-sql.sh` to update your library path. See “Using shell scripts to set the library path (optional)” on page 92 for more information.

- On Solaris, add the `$ARBORPATH/dlls` directory to the `LD_LIBRARY_PATH` variable:

For example, if you are using the C shell, the `.cshrc` file might look like this:

```
setenv ARBORPATH /ibm/db2olap
setenv LD_LIBRARY_PATH /usr/lib:$ARBORPATH/dlls
```

- On AIX, add the \$ARBORPATH/dlls directory to the LIBPATH variable:
For example, if you are using the C shell, the .cshrc file might look like this:

```
setenv ARBORPATH /ibm/db2olap
setenv LIBPATH /usr/lib:$ARBORPATH/dlls
```

- On HP-UX, add the \$ARBORPATH/dlls directory to the SHLIB_PATH variable:

For example, if you are using the C shell, the .cshrc file might look like this:

```
setenv ARBORPATH /ibm/db2olap
setenv SHLIB_PATH /usr/lib:$ARBORPATH/dlls
```

The .profile file, if you are using the Bourne or Korn shell, might look like this:

```
ARBORPATH=/home/ibm/db2olap
export ARBORPATH
SHLIB_PATH=$SHLIB_PATH:$ARBORPATH/dlls
export SHLIB_PATH
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$ARBORPATH/dlls:
/usr/bin/X11:/sbin:.export PATH
```

HP-UX uses SHLIB__PATH to search for shared library files at runtime and LPATH to link to shared library files while compiling. For more information, see your UNIX documentation.

Using shell scripts to set the library path (optional)

The script that links SQL Interface to your ODBC driver (inst-sql.sh) also creates shell scripts which you can use to automatically update the library path for Essbase. The inst-sql.sh script generates the following files and places them in the \$ARBORPATH/setup directory:

- odbc.csh for the C shell
- odbc.sh for the Bourne or Korn shell.

These files contain commands that specify the correct environment variables for Essbase. \$ARBORPATH/dlls is added to the library path, and ODBCINI is set to \$ARBORPATH/bin/.odbc.ini.

To use these scripts so that Essbase environment variables are available to your shell environment every time you log on, complete one of the following tasks after you run inst-sql.sh:

- In the C shell, use the **source** command against odbc.csh in the .login or .cshrc file. For example, edit the .login or .cshrc file to include this line:
% source \$ARBORPATH/setup/odbc.csh
- In the Bourne or Korn shell, use the **.** (dot) command against odbc.sh in the .profile file. For example, edit the .profile file to include this line:
% . \$ARBORPATH/setup/odbc.sh

Binding files for the MERANT DB2 driver on AIX or Solaris

To bind files so that the DB2 driver can understand the MERANT commands on AIX or Solaris, complete the following steps after you run `inst-sql.sh`:

1. Change to the `$ARBORPATH/bin` directory.

For example:

```
cd /home/ibm/db2olap/bin
```

2. Start the DB2 command-line interface.

For example:

```
db2
```

3. Connect to the DB2 database.

For example:

```
CONNECT TO DATABASENAME
```

4. Bind all files, or at least the `.bnd` files, in your `$ARBORPATH/bin` directory.

For example:

```
BIND iscsso.bnd
```

File names may change for new driver versions. For the files you need to bind, list all files with the `.bnd` extension in the `$ARBORPATH/bin` directory.

It is recommended that you bind all the files in the `bin` directory.

5. Grant user privileges for accessing data.

For more information about binding and granting privileges, see the *MERANT DataDirect Connect ODBC Reference*.

Editing `.odbc.ini` driver files

To complete your ODBC driver setup after installing Essbase SQL Interface, you need to modify `.odbc.ini` to include the correct driver and data source names. See the following examples for the IBM DB2 driver on AIX, and for the MERANT DB2 driver on Solaris.

Editing `.odbc.ini` for the IBM DB2 driver on AIX: To set up your `.odbc.ini` file for DB2 Universal Database on AIX, complete the following tasks after you run `inst-sql.sh`.

On AIX, `inst-sql.sh` prompts you to choose between the MERANT DB2 driver and the IBM Call-Level Interface (CLI) for DB2 driver. `inst-sql.sh` links SQL Interface to different libraries, depending on which DB2 driver you choose. To switch DB2 drivers sets after you run `inst-sql.sh`, reinstall SQL Interface.

In \$ARBORPATH (the directory where DB2 OLAP Server is installed), in the /bin directory, modify .odbc.ini to include the correct driver and data source name.

For example:

```
[ODBC Data Sources]
SAMPLE=IBM DB2 ODBC DRIVER

[SAMPLE]
Driver=/home/db2inst/sql1lib/lib/db2.o
Database=dbname

[ODBC]
Trace=0
TraceFile=odbctrace.out
InstallDir=/home/db2inst/sql1lib/odbc1ib
```

In these examples, /home/db2inst is the \$INSTHOME directory. You cannot use environment variables in .ini files. See the documentation for DB2 Universal Database for more information.

Editing .odbc.ini for the MERANT Oracle driver on Solaris: To set up your .odbc.ini file for Oracle 8 on Solaris, complete the following tasks after you run inst-sql.sh:

In \$ARBORPATH (the directory where Essbase is installed), modify .odbc.ini to include the correct driver and data source name.

For example:

```
[ODBC Data Sources]
Oracle8=MERANT 4.0 Oracle 8 Driver
[Oracle8]
Driver=/home/ibm/db2olap/dlls/ARor815.so
Description=Oracle8
ServerName=name
```

where *name* is the appropriate Oracle server name.

Configuring the data source and ODBC driver

You have already completed the following tasks:

- Installed SQL Interface.
- Set up the ODBC driver for your platform/driver configuration.
- Linked SQL Interface to the ODBC driver using inst-sql.sh.
- Set the library path for Essbase.
- Completed ODBC driver setup by editing the odbc.ini file.
- Bound files in ESSBASE/bin, if you are using the DB2 driver.

You are now ready to configure the data source. You may also need to configure your ODBC driver.

Chapter 8. Configuring data sources for DB2 OLAP Integration Server

To create OLAP models or build metaoutlines, you must connect DB2 OLAP Integration Server to both an OLAP Metadata Catalog and a relational data source. To make these connections, you must first configure each data source by mapping 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, use the `odbcconfig` utility or edit the `odbc.ini` file. All configuration of ODBC data sources is done only on the computer that runs Essbase Integration Server.

If a supported ODBC driver is already mapped to the relational data source, you do not need to map it again. In this case, you only need to map a supported ODBC driver to the OLAP Metadata Catalog.

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.

DB2 OLAP Integration Server, ODBC, and the database client software must be installed on the same computer.

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

- DB2 Universal Database
- Informix CLI32
- MS SQL Server Client, optional, or MDAC
- Oracle Net8 or SQL*Net
- Sybase Open Client, optional on Windows if you use the ASE driver

Ensure that the versions of the database client and the RDBMS are compatible. This chapter provides information on how to configure the ODBC drivers. For information on how to install and configure the database client, see the RDBMS and database client documentation.

Configuring the database client software

The client software for the RDBMS and the ODBC driver must be installed on the same computer as Essbase Integration Server. The database client software works with the ODBC driver to make ODBC connections to RDBMS databases.

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.

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 Essbase 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 the RDBMS that you are using.

Supported ODBC drivers

DB2 OLAP Integration Server supports specific ODBC drivers for connecting to RDBMSs. For a list of supported ODBC drivers by platform, see “Supported ODBC drivers and relational data sources” on page 39.

For detailed information on the MERANT drivers provided with Essbase Integration Services, see the MERANT DataDirect Connect ODBC Reference, which is available on Windows from the **Start > Programs > IBM DB2 OLAP Server 8.1 > Documentation** information map.

Configuring data sources on Windows systems

To configure a relational data source on Windows NT, Windows 2000, or Windows XP systems, you must have an Open Database Connectivity (ODBC) connection from the DB2 OLAP Integration Server to the relational data source and to the OLAP Metadata Catalog. DB2 OLAP Integration Server V8.1 requires an `odbc.ini` file for operation of ODBC, and this file is in the Registry under `\HKEY_LOCAL_MACHINE\SOFTWARE\ODBC`. Use ODBC Data Source Administrator to store information about how to connect to a relational data source. The `ODBC.ini` file is available in ODBC software packages and is included with Microsoft Office software. For more information about applications that install ODBC drivers or the ODBC Administrator, visit the Microsoft support Web site.

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.

The following example creates a DB2 data source. The dialog boxes for other RDBMSs will differ.

To configure a relational data source with ODBC Administrator:

1. On the Windows desktop, open the Control Panel window and 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.
2. In the ODBC Data Source Administrator dialog box, click the **System DSN** tab.
3. Click **Add** to open the Create New Data Source dialog box.
4. In the driver list box of the Create New Data Source dialog box of ODBC Administrator, select an appropriate driver, such as IBM DB2 ODBC Driver, and click **Finish** to open the ODBC IBM DB2 Driver – Add dialog box.
5. In the ODBC IBM DB2 Driver – Add dialog box Database alias drop-down list, select the name of the database for your relational data source (for example, TBC in the sample application). The name of the selected database is automatically displayed in the **Data Source Name** text box.
6. 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.
7. 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 Essbase Integration Services Console.

8. 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.

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

1. On the Windows desktop, open the Control Panel window and 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.
2. In the ODBC Data Source Administrator dialog box, click the **System DSN** tab.
3. Click **Add** to open the Create New Data Source dialog box.
4. 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.
5. In the ODBC IBM DB2 Driver – Add dialog box 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.
6. 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. 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

7. 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 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 Essbase Integration Services Console.

8. 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 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, add a new driver or data source, or change the driver or data source. You can use the **odbcconfig** utility or your favorite editor to edit the `odbc.ini` file. You can also use **odbcconfig** to check environment settings. The **odbcconfig** utility is located in the `ishome/bin` directory.

To use the **odbcconfig** utility with DB2 Universal Database, the environment variable `INSTHOME` must be set to the DB2 Universal Database installation directory name, and exported. Use the **odbcconfig** utility to verify, add, or delete ODBC connections. See the *Integration Server Administration Guide* for more information.

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 DB2 OLAP 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.

The procedure in this section shows you how to configure a data source by manually editing the `odbc.ini` file. You can also configure a data source using the **`odbcconfig`** utility. See the *Integration Server Administration Guide* for more information.

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 Essbase 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.

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.

4 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 `$ISHOME/odbc/merant40/docs` 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 Universal Database

Here is an example of how you might edit `odbc.ini` to connect to a relational data source, such as `db2data` on DB2 Universal Database (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 Informix

The following example illustrates how you might edit `odbc.ini` to connect to a relational data source, such as `ifxdata`, on Informix Version 9 (on HP-UX), using a MERANT Version 4.0 ODBC driver:

```
[ODBC Data Sources]
ifxdata=Informix 9 Source Data on HP-UX
...
[ifxdata]
Driver=
/home/hyperion/essintegration/odbc/merant40/lib/ARinf17.s1
Description=Informix9 ODBC Driver
Database=ifxdata
HostName=informixhost
```

Example of ODBC settings for Oracle

The following example illustrates how you might edit `odbc.ini` to connect to a relational data source, `oradata`, on Oracle Version 8 (on Solaris), using a MERANT Version 4.0 ODBC driver:

```
[ODBC Data Sources]
oradata=Oracle8 Source Data on Solaris
...
[oradata]
Driver=
/home/hyperion/essintegration/odbc/merant40/lib/ARor817.so
Description=my oracle source
ServerName=OracleSID
```

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

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 101. No other changes are required.

You can also use the `odbcconfig` utility to define an OLAP Metadata Catalog configuration. See the *Integration Server Administration Guide* for information on `odbcconfig`. You must create an OLAP Metadata Catalog database in a supported RDBMS before configuring it as an ODBC data source.

The following example illustrates how you might edit `odbc.ini` to connect to the OLAP Metadata Catalog, `TBC_MD`, on DB2 Universal Database (on Solaris Operating System), using a native ODBC driver:

```
[ODBC Data Sources]
ocd6a5a=db2 v6
...
[ocd6a5a]
Driver=/home/db2inst1/sqllib/lib/db2.0
Description=db2
```

The following example illustrates how you might edit `odbc.ini` to connect to an OLAP metadata catalog, such as `TBC_MD`, on the Solaris Operating System using a MERANT ODBC driver:

```
4
[ODBC Data Sources]
sybdata=Sybase11 on Solaris Operating System
...
[sybdata]
Driver=/home/hyperion/essintegration/
odbc/merant40/lib/ARase17.so
Description=Data in sybase database on solaris
Database=sybdata
LogonID=uid
Password=pwd
NetworkAddress=serverhost,4100
EnableDescribeParam=1
EnableQuotedIdentifiers=0
OptimizePrepare=1
RaiseErrorPositionBehavior=0
SelectMethod=0
ApplicationUsingThreads=1
```

Chapter 9. Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server

Before you run DB2 OLAP 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.

If you intend to use the sample application to learn DB2 OLAP Integration Server, skip to Chapter 10, “Setting up the sample applications” on page 111. 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, either automatically, through the use of Essbase Integration Services Console, or manually. For manual creation of an OLAP Metadata Catalog, this chapter lists the SQL scripts you must run to create tables for the catalog. The procedures in this chapter assume that you know how to create a database using a relational database management system (RDBMS) and how to create tables by running SQL scripts.

About OLAP metadata catalogs

An OLAP Metadata Catalog is a relational database containing 33 tables in which DB2 OLAP Integration Server stores OLAP models and metaoutlines. DB2 OLAP Integration Server supports the following RDBMSs:

- IBM DB2 Universal Database
- Informix CLI32
- MS SQL Server Client
- Oracle Net8 or SQL *Net
- Sybase Open Client

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 DB2 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.

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 and, by using the XML Import/Export utility, you can move OLAP models and metaoutlines from one OLAP Metadata Catalog to another.

Creating an OLAP Metadata Catalog

To create a database for an OLAP Metadata Catalog, you must have permission to create tables in the database, or similar access privileges required by the RDBMS you are using.

On MS SQL Server, the tables in a catalog will not be accessible if they were created by a user that does not have db_owner privileges.

To create an OLAP metadata catalog:

1. 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
2. Create a database for OLAP Metadata Catalog tables using an RDBMS:
 - a. Create a database with 30 MB for storage.
 - b. Create user names and passwords for the database.
 - c. Grant user privileges or permissions for the database.
3. Take one of the following actions:
 - Create an OLAP Metadata Catalog automatically by using the Integration Services Console (see *Creating an OLAP Metadata Catalog Automatically*).
 - Create tables for the OLAP Metadata Catalog database by running SQL scripts (see *Creating an OLAP Metadata Catalog Manually*).

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 an OLAP Metadata catalog automatically

When you complete installation of DB2 OLAP Integration Server and start Essbase Integration Services Console, the program automatically displays the OLAP Metadata Catalog Setup dialog box shown in Figure 14. This dialog box enables you to create an OLAP Metadata Catalog automatically, using Essbase

Integration Services Console. If you have an older version of the OLAP Metadata Catalog, you also use this dialog box to upgrade it to the current version.

To create the OLAP metadata catalog automatically:

1. From the **Essbase Integration Server** drop-down list in the **OLAP Metadata Catalog Setup** dialog box, select the server computer on which you have installed DB2 OLAP Integration Server.
2. From the ODBC Data Source Name (DSN) drop-down list, select the Data Source Name for the OLAP Metadata Catalog that you are creating.

Note: You must have created the database in which you will store the OLAP Metadata Catalog, assigned the appropriate user permissions, and configured the ODBC connection before you can create the catalog.

3. In the **User Name** text box, type the user name to which you have assigned permission to access the database in which the OLAP Metadata Catalog will be stored.
4. In the **Password** text box, type the password for the user name to which you have assigned permission to access the database in which the OLAP Metadata Catalog will be stored.
5. Leave the **Show this dialog at Startup** check box selected to have the **OLAP Metadata Catalog Setup** dialog box display automatically each time you start up Essbase Integration Services Console.
6. If you do not select the **Show this dialog at Startup** check box, you can access the OLAP Metadata Catalog Setup dialog box at any time by selecting **Tools > OLAP Metadata Catalog Setup** from either the OLAP Model or OLAP Metaoutline main window.
6. Click **Create**.

Essbase Integration Services Console creates the OLAP Metadata Catalog automatically. Proceed to the section titled Setting up the Sample Application.

Note: If you are migrating from a previous release of DB2 OLAP Integration Server, the OLAP Metadata Catalog auto-creation process upgrades your existing OLAP Metadata Catalog to the current version.

Deleting an OLAP Metadata Catalog

You can delete an existing OLAP Metadata Catalog at any time. Remember that if you delete an OLAP Metadata Catalog, you also delete the OLAP models and metaoutlines that it contains.

To delete an OLAP Metadata Catalog:

1. From either the OLAP Model or OLAP Metaoutline main window, select **Tools > OLAP Metadata Catalog Setup** to display the OLAP Metadata Catalog Setup dialog box.
2. From the **Server Name** drop-down list, select the appropriate DB2 OLAP Integration Server computer.
3. From the **ODBC Data Source Name (DSN)** drop-down list, select the Data Source Name for the OLAP Metadata Catalog that you want to delete.
4. Enter your user name and password and click **Delete**.
5. Click **Yes** in the confirmation prompt to delete the catalog.

If the OLAP Metadata Catalog that you are attempting to delete is currently in use, you will be prompted to disconnect from the catalog before you can delete it.

Creating an OLAP Metadata catalog manually

If you choose to create the OLAP Metadata Catalog manually, you must create a set of tables that the catalog uses to store OLAP model and metaoutline metadata. Create the tables for the OLAP Metadata Catalog by running SQL scripts using the same utility program you normally use to create tables. The SQL scripts to create tables for the OLAP Metadata Catalog are located in the ocscript directory where you installed DB2 OLAP Integration Server.

The utilities listed in Table 16 have been tested to work with the SQL scripts.

Table 16. Tested utilities for creating TBC tables

Database	SQL script	Utility program
DB2	<ul style="list-style-type: none"> • oc_create_db2.sql • oc_drop_db2.sql • oc_upgrade20_db2.sql • oc_upgrade61_db2.sql • oc_upgrade65_db2.sql 	<ul style="list-style-type: none"> • DB2 Command Window, or • >DB2 -tvf
Informix	<ul style="list-style-type: none"> • oc_create_informix.sql • oc_drop_informix.sql • oc_upgrade20_informix.sql • oc_upgrade61_informix.sql • oc_upgrade65_informix.sql 	DBAccess
MS SQL Server	<ul style="list-style-type: none"> • oc_create_sqlsrv.sql • oc_drop_sqlsrv.sql • oc_upgrade20_sqlsrv.sql • oc_upgrade61_sqlsrv.sql • oc_upgrade65_sqlsrv.sql 	Query Analyzer (MS SQL Server 7.0 and 2000)

Table 16. Tested utilities for creating TBC tables (continued)

Database	SQL script	Utility program
Oracle	<ul style="list-style-type: none"> • oc_create_oracle.sql • oc_drop_oracle.sql • oc_upgrade20_oracle.sql • oc_upgrade61_oracle.sql • oc_upgrade65_oracle.sql 	SQL*Plus
Sybase	<ul style="list-style-type: none"> • oc_create_sybase.sql • oc_drop_sybase.sql • oc_upgrade20_sybase.sql • oc_upgrade61_sybase.sql • oc_upgrade65_sybase.sql 	ISQL

DB2 OLAP Server provides five SQL scripts for each RDBMS:

- oc_create_database_name.sql to build tables
- oc_drop_database_name.sql to drop tables
- oc_upgrade20_database_name.sql to upgrade tables from Version 7.1 GA to Version 7.1 FixPak 3
- oc_upgrade61_database_name.sql to upgrade tables from Version 7.1 FixPak 4 to Version 7.1 FixPak 7
- oc_upgrade65_database_name.sql to upgrade tables from Version 7.1 FixPak 7 to Version 8.1

To create tables for the OLAP Metadata Catalog database:

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 ocscrip 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 manually, you must map the catalog to a supported ODBC driver.

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.

To use DB2 OLAP Integration Server on Sybase, you must have execute permission for `sp-fkeys` in `Sybsystemprocs`.

Upgrading the OLAP Metadata Catalog

If you have existing OLAP Metadata Catalogs from a release earlier than DB2 OLAP Integration Server Version 8.1, and you choose to create the OLAP Metadata Catalog manually, you must upgrade the existing OLAP Metadata Catalogs before you can use them with this release.

Note: If you used Essbase Integration Services Console to create an OLAP Metadata Catalog automatically after completing installation, the system upgrades your existing OLAP Metadata Catalog and you do not need to upgrade it.

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 DB2 OLAP Integration Server. The upgrade scripts are named `oc_upgrade6x_database_name.sql` and are listed in Table 21 on page 125 and Table 16 on page 108, along with the utility programs with which they have been tested.

To upgrade tables for the OLAP Metadata Catalog database:

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.
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 10. Setting up the sample applications

This chapter describes how to set up the sample applications for DB2 OLAP Server and DB2 OLAP Integration Server.

DB2 OLAP Server offers four sample applications. These sample applications are based on The Beverage Company (TBC), a fictional company that manufactures and sells soft drink products.

- The main application used in the documentation is called Sample. It contains three databases: Basic, Interntl, and Xchgrate. The latter two contain specific features corresponding to the currency conversion option.
- The second application, called Demo, contains one database, Basic, which is referred to in the *Technical Reference*.
- The third application, called Samppart, is the first of two applications designed to show the features of Partitioning. Samppart contains a Company database that is the target of a partitioned database.
- The fourth application, called Sampeast, is the second of two applications designed for Partitioning. Sampeast contains an East database that is the source of data for a partitioned database.

DB2 OLAP Integration Server offer a sample application that contains these features:

- A database for a fictitious company called The Beverage Company (TBC)
- A sample OLAP model and metaoutline created from the TBC database
- A Hybrid Analysis-enabled OLAP model and metaoutline (for DB2 Universal Database and Oracle users only)
- A special OLAP Metadata Catalog to store the sample OLAP models and metaoutlines and any additional OLAP models and metaoutlines that you create using the sample data

Samples for DB2 OLAP Server

This section describes how to set up the sample applications for DB2 OLAP Server.

Loading the sample databases for DB2 OLAP Server

To prepare the sample databases for use, you must load data into them. Each of the sample databases has a corresponding data load file.

Table 17 on page 112 shows the sample applications, the databases that are contained within them, and their corresponding data load files:

Table 17. Sample databases and data load files

Application	Database	Data load file
Sample	Basic	Calcdat
	Interntl	Currcalc
	Xchgrate	Rates
Demo	Basic	Data
Samppart	Company	Calccomp
Sampeast	East	Calceast

To load data into the sample databases, choose an application and database and the corresponding data load file from Table 17 and complete these steps.

1. Open Application Manager.
2. Connect to the server on which the application is installed.
3. From the Applications list box, select one of the applications.
4. From the Databases list box, select one of the databases.
5. From the Database drop-down menu, select **Database > Load Data**. The Data Load dialog box is displayed. The Data Load dialog box is described in detail in the Application Manager online help, and in the *Database Administrator's Guide*.
6. Click the **Find** button above the **Data files** list box to locate a data file to load. The Open Server Data File Objects dialog box is displayed.
7. Ensure that **List Objects of Type** is **Text files**.
8. From the **Objects** list box, select a data load file.
9. Click **OK** to close the dialog box. DB2 OLAP Server updates the Data Load dialog box to display the file you selected.
10. Click **OK** to load the selected file. The data file is a fully calculated data set. It may take several minutes to load, and then the Data Load Completed dialog box is displayed.
11. Click **Close**.

Providing user access to the sample applications

DB2 OLAP Server provides a comprehensive security system for a secure multiple user environment. By default, the sample applications are created with a security access level of None, which means that no user can connect to the sample databases unless the user is defined as a supervisor.

The system supervisor, defined when the server component was installed, automatically has supervisor privilege. Therefore, the system supervisor can make the sample applications available to other users.

To provide all users with Write access to the sample databases, repeat the following procedure for each database.

1. Log on using the system supervisor account.
2. Open the Application Manager.
3. Connect to the server on which the application for which you want to provide access is installed.
4. From the Applications list box, select the application.
5. Select **Application > Settings**. The Application Settings dialog box is displayed.
6. From the **Minimum Database Access** group, select the **Write** option and click **OK** to save the setting. DB2 OLAP Server closes the dialog box and returns you to the Application Desktop window.

You can also use Administration Services to perform this task, using the Application Properties window. See the Administration Services Online Help for more information.

The applications are now ready for use. If you want to use the Partitioning applications, follow the instructions in "Setting the environment for the partitioning sample applications". If you do not want to use the Partitioning applications, your next step might be to create the appropriate user names to be able to log on to DB2 OLAP Server. See the *Database Administrator's Guide* for information on creating users.

Setting the environment for the partitioning sample applications

In some licensing instances, installation of the sample applications may be restricted to the main application (Sample) and its three databases (Basic, Interntl, and Xchgrate). Refer to your license agreement for details.

DB2 OLAP Server includes two sample applications and databases that show the features of Partitioning:

- Samppart, Company
- Sampeast, East

These applications and databases include partition definitions stored in .ddb files. The .ddb files define the map between member combinations in the target database, Company, and the source database, East. The *Database Administrator's Guide* describes Partitioning, member combinations, and the differences between target and source databases.

For these applications to work in your environment, you need to match or change the information contained in the .ddb files.

To use the applications:

1. Create a user named `partitionuser`. This is necessary because the sample partitions use this user name to transfer data between the Company database and the East database.
2. Change the server name that is embedded in the applications.
Do not edit the `.ddb` files to change this information. Instead, complete the steps in the following sections: “Creating the partition user” describes how to create the required user name, so that your users can use these applications. “Changing embedded server names in the sample partition definitions” on page 115 describes how to change the embedded server name to your server name.

Creating the partition user

You must create a user named `Partitionuser` before you work with the `Samppart` and `Sampeast` applications. `Partitionuser` must have Application Designer access to both applications.

To create the `Partitionuser`:

1. Open the Application Manager.
2. Using a user name that is a supervisor or that has Create/Delete Users, Groups privilege, connect to the server on which the `Samppart` and `Sampeast` applications are installed.
3. Select **Security > Users/Groups**. The User/Group Security dialog box is displayed.
4. Click **New User**. DB2 OLAP Server displays the New User dialog box.
5. In the **Username** text box, type `partitionuser`. Be sure to use all lowercase letters.
6. In the **Password** text box, type a password. DB2 OLAP Server displays asterisks in the place of the characters you type. Type the password again in the **Confirm Password** text box.
7. Click **App Access**. DB2 OLAP Server displays the User/Group Application Access dialog box.
8. From the **Applications** list box, select `Sampeast` and click App Designer in the Access group.
9. Select `Samppart` from the **Applications** list box, then click App Designer in the Access group.
10. Click **OK** to return to the New User dialog box.
11. Click **OK** to return to the User/Group Security dialog box.
12. Click **Close**.

You can also use Administration Services to perform this task, using the User Properties window. See the online help for Administration Services for more information.

Changing embedded server names in the sample partition definitions

The following steps change the embedded server name in the partition definition files (.ddb) to your server name. The .ddb files shipped with Samppart and Sampeast are based on the server name localhost.

To change the server name in the Samppart Company .ddb file:

1. In the Application Desktop window, select the **Samppart** application and **Company** database.
2. From the **Database** drop down menu, select **Database > Partition Manager**. After a brief pause, DB2 OLAP Server displays the Partition Manager dialog box. If the Partition Manager dialog box does not appear or if it contains no definition entries, you might not have purchased and installed Partitioning. Contact your DB2 OLAP Server system administrator to ascertain this.
3. Under the **Source Cube** heading, select the first definition entry, Localhost: SampEast: East.
4. Click **Edit**. The Dangling Remote Reference dialog box is displayed. Because you are not connected to a server named Localhost, DB2 OLAP Server warns you before it enables you to edit the definition.
5. Click **Advanced**. The Fix Dangling Partition Reference dialog box is displayed.
6. In the **Action** group, select **Correct the invalid partition reference**.
7. In the **Server** text box in the **Reconnect** group, replace Localhost with your server name.
8. Click **Apply**. DB2 OLAP Server displays the Partition Manager dialog box.
9. In the Source Cube list, select the second definition entry, Localhost: Sampeast: East.
10. Repeat 4 through 8.
11. In the Target Cube list, select Localhost: Sampeast: East.
12. Repeat 4 through 8.
13. Select **Partition > Close** to return to the Application Desktop window.

To change the server name in the Sampeast East .ddb file, take the following actions:

1. In the Application Desktop window, select the Sampeast application and East database.
2. From the **Database** drop down menu, select **Partition Manager**. After a brief pause, DB2 OLAP Server displays the Partition Manager dialog box.
3. Under the **Source Cube** heading, select Localhost: samppart: Company.

4. Click **Edit**. The Dangling Remote Reference dialog box is displayed. Because you are not connected to a server named Localhost, DB2 OLAP Server warns you before it enables you to edit the definition.
5. Click **Advanced**. The Fix Dangling Partition Reference dialog box is displayed.
6. In the **Action** group, select **Correct the invalid partition reference**.
7. In the **Server** text box in the **Reconnect** group, replace Localhost with your server name.
8. Click **Apply**. DB2 OLAP Server displays the Partition Manager dialog box.
9. In the Source Cube list, select the first definition entry, Localhost: samppart: Company.
10. Repeat steps 4 through 8.
11. In the Target Cube list, select Localhost: Sampeast: East.
12. Repeat steps 4 through 8.
13. Select **Partition > Close** to return to the Application Desktop window.

Loading Data into Samppart and Sampeast

After setting up the environment for the Samppart and Sampeast databases, you need to load data into each database. This topic shows you how to load the Samppart Company database.

1. Open the Application Manager.
2. Connect to the server on which Samppart is installed.
3. From the **Applications** list box, select Samppart.
4. From the **Databases** list box, select Company.
5. From the **Database** drop-down list, select **Load Data**. The Data Load dialog box is displayed.
6. From above the Data files list box, click the **Find** button. The Open Server Data File Objects dialog box is displayed.
7. Ensure that **List Objects of Type** is **Text files**.
8. From the Objects list box, select CALCCOMP.
9. Click **OK** to close the dialog box.
10. Click **OK** to load the selected file.
11. Click **Close**.

Repeat for the Sampeast application, East database, and calceast data file.

Samples for DB2 OLAP Integration Server

DB2 Universal database and Oracle users can create the sample application (TBC, TBC Model, and TBC Metaoutline) either manually or by using the automated functionality in the Essbase Integration Services Console. However, to create the sample application with Hybrid Analysis enabled, you must also complete the tasks in “Setting up the sample application manually” on page 120. Upon completion of these tasks, the sample TBC relational data source will contain Hybrid Analysis data. The OLAP Metadata Catalog for DB2 and Oracle users will contain a Hybrid Analysis-enabled OLAP model (HA TBC Model) and metaoutline (HA TBC Metaoutline).

Setting up the sample application involves two main tasks:

1. You create two relational databases: one for the data in the TBC sample application and one for the OLAP Metadata Catalog database (TBC_MD). This is a manual task that you must perform before setting up the sample application, regardless of whether you choose the automatic or manual method.
2. You then create tables in each of these databases and load data into them, using scripts provided with DB2 OLAP Integration Server. This can be performed either automatically, through the use of Essbase Integration Services Console, or manually.

When you finish setting up the sample application, the OLAP metadata catalog database (TBC_MD) will contain an OLAP model (TBC Model) and a metaoutline (TBC Metaoutline) based on data in the sample TBC relational data source.

If you have installed the sample application from a previous release of DB2 OLAP Integration Server, you should back up and preserve your existing sample database, OLAP Metadata Catalog, and the OLAP models and metaoutlines that OLAP Metadata Catalog 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 110). You cannot, however, store newly-created OLAP models and metaoutlines in your previous catalog.

This section tells you what to consider before and after you set up the sample application, whether automatically or manually. For manual setup, this section describes the scripts and batch files you must run to create and load the TBC database and the TBC_MD OLAP Metadata Catalog.

The procedures in this section assume that you know how to create a database using a relational database management system (RDBMS) and have installed the required database client software. For manual setup, the procedures assume that you know 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.

About the DB2 OLAP Integration Server samples

DB2 OLAP Integration Server provides a sample database based on a fictitious company named The Beverage Company (TBC) and includes the following components:

- Sample OLAP Metadata Catalog (TBC_MD)
- Sample data source (TBC)
- Sample OLAP model (TBC Model) and sample OLAP metaoutline (TBC Metaoutline)
- DB2 Universal Database and Oracle: Additional sample data that demonstrates Hybrid Analysis functionality.
- DB2 Universal Database and Oracle: Additional sample Hybrid Analysis-enabled OLAP model (HA TBC Model) and sample Hybrid Analysis-enabled OLAP metaoutline (HA TBC Metaoutline)

The sample OLAP models and metaoutlines are provided as XML files, supported by Java. The sample application demonstrates how to create a Measures dimension recursively, using a sort on an alternative column in the relational data source. It also shows how to load metadata from database columns. Additionally, the sample application includes UDAs and Date Time Series data, additional years, drill-through paths, and a Time table to demonstrate data load incremental updates.

For DB2 and Oracle users, the sample application also provides sample data, a sample OLAP model, and a sample OLAP metaoutline that demonstrate Hybrid Analysis functionality.

Before you set up the sample application

Before you set up the sample application, you must install the database client software for a supported RDBMS. When setting up the sample application, you must use a computer on which the server component of DB2 OLAP Integration Server and the necessary database client software are both installed. For more information, see “About configuring relational data sources” on page 97.

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

It is recommended that you use the automatic installation process to install the sample applications (see “Setting up the sample application manually” on page 120).

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. For manual sample application setup, refer to Table 18 and also verify that the appropriate client utility program is installed.

Table 18. Required utilities for setting up the sample application

Database	Utility Programs
DB2 Universal Database	DB2 Command Window, or > DB2 -tvf
Informix	DBAccess
MS SQL Server	Query Analyzer (MS SQL Server 7.0 and 2000)
Oracle	SQL * Plus
Sybase	<ul style="list-style-type: none"> • ISQL • BCP command line utility

Setting up the sample application automatically

When you bring up DB2 OLAP Integration Server, it displays the OLAP Metadata Catalog Setup dialog box to enable automatic setup of the catalog (see “Creating an OLAP Metadata catalog automatically” on page 106). Whether or not you chose to create the OLAP Metadata Catalog Setup automatically, you can choose to create the sample application automatically (after you have finished creating the OLAP Metadata Catalog). The automatic sample application creation feature enables you to complete the process quickly and easily. You are encouraged to use this convenient feature, but if you choose manual setup, see “Setting up the sample application manually” on page 120 for detailed procedures.

DB2 and Oracle: If you plan to set up the sample application for Hybrid Analysis, you must follow the procedures in “Setting up the sample application manually” on page 120.

ODBC connections must be configured before you begin to set up the sample application. If you have not yet configured ODBC connections, see Chapter 8, “Configuring data sources for DB2 OLAP Integration Server” on page 97 for information on setting up ODBC data sources, including an OLAP Metadata Catalog and relational data sources.

To set up the sample application automatically:

1. Select **Tools > Create Sample** to access the Sample Application Setup dialog box.
2. In the **ODBC Name (DSN)** text box, select or type the ODBC name for creating or dropping (deleting) tables in the source database.

The **User Name** text box displays the name, TBC (lowercase tbc for Informix). This name must be used to create the TBC sample application database.

3. In the **Password** text box, type the password for the user to which you have assigned privileges to access the sample application TBC database.
4. Click **Create** to create the sample application.

DB2 OLAP Integration Server creates the necessary tables, loads the TBC sample data into the tables, and loads the sample OLAP model and metaoutline data into the OLAP Metadata Catalog.

If DB2 OLAP Integration Server detects that you are not currently connected to the OLAP Metadata Catalog in which to load the sample OLAP model and metaoutline data, the Login dialog box is displayed so that you can enter the information necessary to connect. After you close the Login dialog box and are returned to the Sample Application Setup dialog box, click **Create** again to initiate setup of the sample application.

If, after clicking **Create**, DB2 OLAP Integration Server detects either an existing sample data source or an existing sample OLAP model and metaoutline in the OLAP Metadata Catalog, you are prompted with the message that tables already exist in the database. Click **OK** to return to the Sample Application Setup dialog box to reenter information, or click **Delete** to delete the existing sample application data. If you click **Delete**, your existing OLAP Metadata Catalog sample OLAP models and metaoutlines will be deleted and replaced with the sample OLAP model and metaoutline for the current release.

After you have created the sample application, you are ready to begin working with Essbase Integration Services Console to create your own sample OLAP models and metaoutlines.

Setting up the sample application manually

To set up the sample application manually, you begin by creating two databases in a supported 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 Essbase Integration Services. The names and directory locations of these scripts are detailed in the topics that follow.

It is recommended that you use the automatic installation process to install the sample applications “Setting up the sample application automatically” on page 119; however, if you plan to set up the sample application for Hybrid Analysis, you must also complete the procedures in this section.

Be aware that the sample application setup process for Hybrid Analysis may take some time.

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

- A table creation script for the TBC sample database, located in the `samples\tbcdbase` directory.
DB2 Universal Database and Oracle: To create the sample application with Hybrid Analysis enabled, you must run two scripts, `tbc_create_*.sql` and `tbc_create_ha_*.sql`.
- A data load script for the TBC sample database, located in the `samples\tbcdbase` directory.
DB2 Universal Database and Oracle: To create the sample application with Hybrid Analysis enabled, you must run two scripts, `sampledata.sql` and `sampledata_ha.sql`.
- A table creation script for the TBC_MD OLAP Metadata Catalog database, located in the `ocscript` directory
- XML files to import a sample OLAP model and metaoutline in the TBC_MD OLAP Metadata Catalog database, located in the `samples\tbcmodel` directory
DB2 Universal Database and Oracle: To create the sample application with Hybrid Analysis enabled, you must import a second OLAP model (`sample_ha_model.xml`) and metaoutline (`sample_ha_metaoutline.xml`).

The sample application scripts and catalog creation scripts are installed with the server software during the installation process. Different versions of the scripts are provided for each of the supported RDBMSs.

DB2 Universal Database and Oracle: Scripts are provided to create the sample application with Hybrid Analysis enabled. These scripts contain the letters `ha` in their names.

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.

Setting up the TBC relational data source

The relational data source for the sample application is TBC. 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 relational data source:

1. Create the TBC database using an RDBMS. For more information, see “Creating the TBC database” on page 122.
2. Create tables for the TBC database by running SQL scripts. For instructions, see “Creating tables for the TBC database” on page 122.

3. Load data into the TBC database by running SQL scripts. For instructions, see “Loading data into the TBC tables” on page 123.

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

1. Create a database device or tablespace named TBC.
2. Allot 20 MB for storage.
3. Create a user TBC who can drop and create tables. The user TBC must create the tables for the TBC database, or portions of the sample application will not work.
4. Grant user privileges or permissions to create and drop tables.

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 DB2 OLAP Integration Server.

The utilities listed in Table 19 have been tested to work with the SQL scripts:

Table 19. Tested utilities for creating TBC tables

Database	SQL script	Utility program
DB2	<ul style="list-style-type: none"> • tbc_create_db2.sql • tbc_create_ha_db2.sql • tbc_drop_db2.sql • tbc_drop_ha_db2.sql 	<ul style="list-style-type: none"> • DB2 Command Window, or • >DB2 -tvf
Informix	<ul style="list-style-type: none"> • tbc_create_informix.sql • tbc_drop_informix.sql 	DBAccess
MS SQL Server	<ul style="list-style-type: none"> • tbc_create_sqlsrv.sql • tbc_drop_sqlsrv.sql 	Query Analyzer (MS SQL Server 7.0 and 2000)
Oracle	<ul style="list-style-type: none"> • tbc_create_oracle.sql • tbc_create_ha_oracle.sql • tbc_drop_oracle.sql • tbc_drop_ha_oracle.sql 	<ul style="list-style-type: none"> • SQL*Plus • SQL*Loader command line utility
Sybase	<ul style="list-style-type: none"> • tbc_create_sybase.sql • tbc_drop_sybase.sql 	ISQL

Each RDBMS has two scripts—one to build tables and another to drop tables.

DB2 and Oracle: To create the sample application with Hybrid Analysis enabled, you must run two scripts, `tbc_create_*.sql` and `tbc_create_ha_*.sql`.

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:

1. Start the utility program.
2. Connect to the TBC database as user TBC. You must create the tables for the TBC database as user TBC or portions of the sample application will not work.

Use lowercase `tbc` for Informix; use uppercase `TBC` for Sybase and SQL Server.

3. In the `samples\tbcdbase` directory, open the appropriate SQL script file.
4. Run the SQL script.

MS-SQL Server: You should receive a message that you did not create data or rows. This message is normal because you created only tables and columns.

Informix: If you run the Informix SQL script (`tbc_create_informix.sql`) more than once, you must first run `tbc_drop_informix.sql` to drop tables before you build them again.

DB2 and Oracle: To create the sample application with Hybrid Analysis enabled, you must run two scripts, `tbc_create_*.sql` and `tbc_create_ha_*.sql`.

5. Verify that you have created the TBC tables; for example, type:

```
SELECT * FROM PRODUCTDIM
```

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: Load data into the TBC tables by running an SQL script using the same utility program you normally use to load tables by running SQL scripts.

DB2 and Oracle: To create the sample application with Hybrid Analysis enabled, you must run two scripts, `samp1edata.sql` and `samp1edata_ha.sql`.

Be aware that the sample application setup process for Hybrid Analysis may take some time.

The utilities listed in Table 20 on page 124 have been tested to work with SQL scripts.

Table 20. Tested utilities for loading data into TBC tables

Database	SQL script	Utility program
DB2	<ul style="list-style-type: none"> sampledata.sql sampledata_ha.sql 	<ul style="list-style-type: none"> DB2 Command Window, or >DB2 -tvf
Informix	<ul style="list-style-type: none"> sampledata.sql 	DBAccess
MS SQL Server	<ul style="list-style-type: none"> sampledata.sql 	Query Analyzer (MS SQL Server 7.0 and 2000)
Oracle	<ul style="list-style-type: none"> sampledata.sql sampledata_ha.sql 	SQL*Plus
Sybase	<ul style="list-style-type: none"> sampledata.sql 	ISQL

To load data into the TBC tables:

1. From the command line, move to the samples\tbcdbase directory where you installed DB2 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 sampledata.sql script file.
5. Run the SQL script using your RDBMS tool.
 DB2 and Oracle: To create the sample application with Hybrid Analysis enabled, you must run a second script, sampledata_ha.sql.
6. Verify that you have loaded data into the TBC tables; for example, type:

```
SELECT * FROM TBC.PRODUCT
```

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

Setting up TBC_MD

The OLAP Metadata Catalog for the sample application is TBC_MD. For more information on OLAP Metadata Catalogs, see Chapter 9, “Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server” on page 105.

To set up the TBC_MD OLAP Metadata Catalog:

1. Create a TBC_MD database using an RDBMS. For more information, see “Creating the TBC_MD Database” on page 125
2. Create tables for the TBC_MD database by running SQL scripts. For instructions, see “Creating tables for the TBC_MD database” on page 125.
3. Load data into the TBC_MD database using the XML import utility. For instructions, see “Loading data into the TBC_MD tables using the XML Import utility” on page 127.

If you have a previous release of DB2 OLAP 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 DB2 OLAP Integration Server.

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

1. Create a database device or tablespace named TBC_MD.
2. Allot 20 MB for storage.
3. Create a user TBC who can drop and create tables. The user TBC must create the tables for the TBC_MD database, or portions of the sample application will not work.
4. Grant user privileges or permissions to create and drop tables.

Creating tables for the TBC_MD database: Create tables for the TBC_MD database with the same utility program you normally use.

The sample application SQL scripts used to create tables for the TBC_MD database are in the ocscript directory where you installed DB2 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 9, “Creating OLAP Metadata Catalogs for DB2 OLAP Integration Server” on page 105

The utilities listed in Table 21 have been tested to work with the SQL scripts:

Table 21. Tested utilities for creating TBC_MD tables

Database	SQL script	Utility program
DB2	<ul style="list-style-type: none"> • oc_create_db2.sql • oc_drop_db2.sql • oc_upgrade20_db2.sql • oc_upgrade61_db2.sql • oc_upgrade65_db2.sql 	<ul style="list-style-type: none"> • DB2 Command Window, or • >DB2 -tvf
Informix	<ul style="list-style-type: none"> • oc_create_informix.sql • oc_drop_informix.sql • oc_upgrade20_informix.sql • oc_upgrade61_informix.sql • oc_upgrade65_informix.sql 	DBAccess

Table 21. Tested utilities for creating TBC_MD tables (continued)

Database	SQL script	Utility program
MS SQL Server	<ul style="list-style-type: none"> • oc_create_sqlsrv.sql • oc_drop_sqlsrv.sql • oc_upgrade20_sqlsrv.sql • oc_upgrade61_sqlsrv.sql • oc_upgrade65_sqlsrv.sql 	Query Analyzer (MS SQL Server 7.0 and 2000)
Oracle	<ul style="list-style-type: none"> • oc_create_oracle.sql • oc_drop_oracle.sql • oc_upgrade20_oracle.sql • oc_upgrade61_oracle.sql • oc_upgrade65_oracle.sql 	SQL*Plus
Sybase	<ul style="list-style-type: none"> • oc_create_sybase.sql • oc_drop_sybase.sql • oc_upgrade20_sybase.sql • oc_upgrade61_sybase.sql • oc_upgrade65_sybase.sql 	ISQL

DB2 OLAP Integration Server provides five SQL scripts for each RDBMS:

- oc_create_database_name.sql to build tables
- oc_drop_database_name.sql to drop tables
- oc_upgrade20_database_name.sql to upgrade tables from DB2 OLAP Integration Server Version 7.1, General Availability (Essbase Integration Services Release 1.9) to Version 7.1, FixPak 3 (Essbase Integration Services Release 2.0)
- oc_upgrade61_database_name.sql to upgrade tables from DB2 OLAP Integration Server Version 7.1, FixPaks 3 through 7 (Essbase Integration Services Release 2.0) to Version 7.1, FixPak 8 (Essbase Integration Services Release 6.2)
- oc_upgrade65_database_name.sql to upgrade tables from DB2 OLAP Integration Server Version 7.1, FixPaks 7 and later (Essbase Integration Services Release 6.1 or 6.2) to Version 8.1 Essbase Integration Services Release 6.5)

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

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. You must create the tables for the TBC_MD database as user TBC or portions of the sample application will not work.
3. In the ocscript directory, open the appropriate SQL script file.
4. Run the SQL script.

MS-SQL Server: You should 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:

```
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 using the XML Import utility: Use the XML Import utility to load OLAP model and metaoutline data into the TBC_MD tables.

The XML files listed in Table 22 have been tested to work with their respective RDBMS. These files are located in the `samples\tbcmodel` directory.

Table 22. XML files for loading data into TBC_MD tables

Database	XML File in <code>sample\tbcmodel</code>
<ul style="list-style-type: none">• DB2 Universal Database• Oracle	<ul style="list-style-type: none">• <code>sample_model.xml</code>• <code>sample_ha_model.xml</code>• <code>sample_metaoutline.xml</code>• <code>sample_ha_metaoutline.xml</code>
<ul style="list-style-type: none">• MS SQL Server• Sybase	<ul style="list-style-type: none">• <code>sample_model.xml</code>• <code>sample_metaoutline.xml</code>
Informix	<ul style="list-style-type: none">• <code>sample_model_informix.xml</code>• <code>sample_metaoutline.xml</code>

To complete the procedure below, you must start Essbase Integration Services Console and connect to the server component of DB2 OLAP Integration Server.

To start Essbase Integration Services Console, and connect to to the server component of DB2 OLAP Integration Server:

1. On the Windows desktop, **Start > Programs > IBM DB2 OLAP Server 8.1 > IBM DB2 OLAP Integration Server > Integration Server.**
2. On the Windows desktop, click **Start > Programs > IBM DB2 OLAP Server 8.1 > IBM DB2 OLAP Integration Server > Desktop.**
3. If necessary, click **Close** to clear the OLAP Metadata Catalog Setup dialog box. Essbase Integration Services Console automatically displays the Login dialog box.
 - a. In the **Server** text box, under Essbase Integration Services, select or type the name of a computer on which the server component of DB2 OLAP Integration Server has been installed.
 - b. In the **OLAP Metadata Catalog** text box, select or type the name of the OLAP Metadata Catalog, TBC_MD, for the sample TBC (The Beverage Company) database.
 - c. In the **User Name and Password** text boxes, type the user name and password for the sample application user, TBC.

To load sample OLAP model data into the TBC_MD tables using XML Import:

1. In Essbase Integration Services Console, select **File > XML Import/Export.**
2. Select the **Import** tab.
3. Click the **Open XML File** button to navigate to the location of the sample XML files. For example, if you selected the default directory during installation, the files are in `ibm\db2olap\IS\Samples\tbcmodel`.
4. Select the XML file to import for the sample OLAP model data, `sample_model.xml`.

Informix: Select the sample OLAP model for Informix, `sample_model_informix.xml`.

5. Click **OK.**

Essbase Integration Services Console displays the XML Import/Export dialog box with the XML file path and name in the **XML File Path** text box.

The name of the sample OLAP model is displayed in the **OLAP Model Name** text box.

6. Click the **Import to Catalog** button to load the selected XML file into the sample OLAP Metadata Catalog.

DB2 and Oracle: To create a sample OLAP model with Hybrid Analysis enabled, repeat this procedure to import the second XML file, `sample_ha_model.sql`.

To load sample OLAP metaoutline data into the TBC_MD tables using XML Import:

- In Essbase Integration Services Console, select **File > XML Import/Export.**
- Select the **Import** tab.

- Click the **Open XML File** button to navigate to the location of the sample XML files. For example, if you selected the default directory during installation, the files are in `ibm\db2olap\IS\Samples\tbcmodel`.
- Select the XML file to import for the sample OLAP metaoutline data, `sample_metaoutline.xml`.
- Click **OK**.
Essbase Integration Services Console displays the XML Import/Export dialog box with the XML file path and name in the **XML File Path** text box, the name of the sample OLAP metaoutline in the **OLAP Metaoutline Name** text box, and the name of the OLAP model upon which this metaoutline is based in the **Based on OLAP Model** text box.
The name of the sample OLAP model is displayed in the **OLAP Model Name** text box.
- Click the **Import to Catalog** button to load the selected XML file into the sample OLAP Metadata Catalog.
DB2 and Oracle: To create a sample metaoutline with Hybrid Analysis enabled, repeat this procedure to import the second XML file, `sample_ha_metaoutline.sql`.

After you set up the sample application

You must connect to both TBC and TBC_MD from Essbase Integration Services Console to create, modify, and store TBC OLAP models and TBC metaoutlines. To make these connections, each database (TBC and TBC_MD) must be mapped to a supported ODBC driver, as described in Chapter 8, “Configuring data sources for DB2 OLAP Integration Server” on page 97

When you connect to Essbase Integration Services Console, you can view TBC columns, tables, OLAP models, and metaoutlines in Essbase Integration Services Console. For more information, see “Viewing TBC tables, columns, OLAP models, and metaoutlines”.

You must connect to TBC and TBC_MD as user TBC, unless you create user name aliases or synonyms in the RDBMS

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 Essbase Integration Services Console.

1. Start DB2 OLAP Integration Server and the Essbase Integration Services Console
2. In a blank Essbase Integration Services Console window, select **Connections > OLAP Metadata Catalog > Connect**, and connect to the OLAP Metadata Catalog, TBC_MD.

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

To see the OLAP model (TBC Model) in the right frame of Essbase Integration Services Console, follow these steps:

1. Start DB2 OLAP Integration Server and the Essbase Integration Services Console
2. If the Login dialog box is not already displayed, in a blank Essbase Integration Services Console window, select **Connections > OLAP Metadata Catalog > Connect**.
3. Connect to the OLAP Metadata Catalog TBC_MD.
4. Click the **Existing** tab, select TBC Model, and click **Open**.
DB2 and Oracle: To view the Hybrid Analysis-enabled OLAP model, select HA TBC Model.
5. In the Data Source dialog box, connect to the TBC sample database.

To see the metaoutline (TBC Metaoutline) in the right frame of Essbase Integration Services Console:

1. Start DB2 OLAP Integration Server and the Essbase Integration Services Console
2. If the Login dialog box is not already displayed, in a blank Essbase Integration Services Console window, select **Connections > OLAP Metadata Catalog > Connect**.
3. Connect to the OLAP Metadata Catalog TBC_MD.
4. Click the **Existing** tab, then click the plus symbol (+) to the left of TBC Model to expand the view, and select TBC Metaoutline.
DB2 and Oracle: To view the Hybrid Analysis-enabled metaoutline, click the plus symbol (+) to the left of HA TBC Model to expand the view, and select HA TBC Metaoutline.
5. Click **Open**. The Data Source dialog box is displayed.
6. In the Data Source dialog box, connect to the TBC sample database.

Chapter 11. Upgrading and migrating from earlier versions

This chapter describes how to upgrade from an earlier release of DB2 OLAP Server or DB2 OLAP Integration Server to Version 8.1, and how to migrate your OLAP applications and databases from one computer to another.

It is recommend that you do a complete, not incremental, backup of all of your DB2 OLAP Server and DB2 OLAP Integration Server files before upgrading. You can then revert to the earlier version if you have problems.

Migrating DB2 OLAP Server

This section describes how to upgrade from an earlier version of DB2 OLAP Server to DB2 OLAP Server Version 8.1 on the same computer and describes what occurs during the upgrade process.

Migrate from RSM to MSM before upgrading to Version 8.1

CAUTION:

Starting in Version 8.1, the Relational Storage Manager (RSM) is withdrawn from DB2 OLAP Server. Applications created using the RSM in previous versions must be migrated to the Multidimensional Storage Manager (MSM) before installing Version 8.1, or you will lose data.

If you are upgrading from an earlier version of DB2 OLAP Server and you have applications that use the Relational Storage Manager (RSM), you must first migrate them to the Multidimensional Storage Manager (MSM) in the earlier version of DB2 OLAP Server before installing Version 8.1, or you will lose data. After migrating your RSM applications to the MSM, you can install Version 8.1, and then migrate your applications to Version 8.1.

The following steps show you how to migrate data from the RSM to the MSM:

1. Start the DB2 OLAP Server.
2. Create an OLAP database in the appropriate application using the Application Manager or a command from the command prompt.
3. Using the Application Manager, click **Start > Open** to open the outline for the database you want to migrate.
4. Determine which dense dimension you want to be the anchor dimension, and remove the RELANCHOR user-defined attribute to that dimension.
You must specify an anchor dimension.

5. Click **File > Save As** to save the outline for the new database that you created. If you are prompted to replace or overwrite an existing file, select Yes.

The OLAP database is activated and restructured, and the modified outline becomes the outline for the new database. During this process, the cube is stored in a multidimensional database.

6. Use the Application Manager to copy any rules files, calculation scripts, or report scripts from the original database to the migrated one. The rules and scripts are not affected by the differences in the two storage managers. The files exist in the same locations in the file system for both products.
7. Change any settings of the new OLAP database to match the original database.
8. Load data into the newly migrated database, and recalculate it in the same manner as the original. As data is loaded and calculated, it is stored in the multidimensional database.

Understanding Input/Output defaults and upgrading

Read this section before upgrading to understand the two I/O access modes available with Version 8.1, and how OLAP databases are affected by an upgrade to Version 8.1, in terms of cache sizes and the I/O access modes.

Determining Which I/O Access Mode to Use

Buffered I/O uses the buffer cache of the file system. If you are upgrading from a release prior to Version 7.1, your databases are using buffered I/O.

Direct I/O bypasses the file system buffer cache and is able to provide faster response time and more potential to optimize cache sizes. If you are upgrading from Version 7.1 or later, your databases are using direct I/O.

Cache memory locking can only be used if direct I/O is used. You also must use direct I/O if you want to use the no-wait (asynchronous) I/O of the operating system. For platform-support information related to I/O, see Table 24 on page 135

Table 23 shows the default I/O access mode for each release, and the I/O access mode choices available (if applicable) for each release. Use this table and this section to determine whether you are currently using buffered or direct I/O, and to decide which you will use after you upgrade to Version 8.1.

Table 23. Default I/O access modes for each DB2 OLAP Server version

Version	Direct I/O	Buffered I/O
Version 1, Version 1.0.1, and Version 1.1	N/A	Default

Table 23. Default I/O access modes for each DB2 OLAP Server version (continued)

Version	Direct I/O	Buffered I/O
Version 7.1, up to FixPak 7	Default	N/A
Version 7.1, FixPak 8 and later	Available by using DIRECTIO TRUE in <code>essbase.cfg</code> .	Default
Version 8.1	Available per-database, as a database setting. Available once for all new or upgraded databases, by using DIRECTIO TRUE config-file setting in <code>essbase.cfg</code> .	Default

The following list can help you to determine which I/O access mode your databases currently use, and how those databases will upgrade if you do not make any changes.

- Databases in Version 7.1, FixPak 8 and up (Essbase Release 6.2) use buffered I/O, unless `essbase.cfg` contains the setting `DIRECTIO TRUE`. When upgraded to Version 8.1, those databases will use same the I/O access mode they used previously. The I/O access mode for any database can be changed once upgraded to Version 8.1, using the database settings.
- Databases in Version 7, up through FixPak 7 (Essbase, Release 6.0 and Release 6.1) use direct I/O. When upgraded to Version 8.1, these databases will use buffered I/O, unless `essbase.cfg` contains the setting `DIRECTIO TRUE`. The I/O access mode for any database can be changed once upgraded to Version 8.1 by using the database settings.
- Databases in releases prior to Version 7.1 (Essbase 6.0) use buffered I/O. When upgraded to Version 8.1 (Release 6.5), all pre-Version 7.1 databases will use buffered I/O, unless `essbase.cfg` contains the setting `DIRECTIO TRUE`. The I/O access mode for any database can be changed once upgraded to Version 8.1, using the database settings.

The `DIRECTIO` setting introduced in Version 7 FixPak 8 is server-wide, and affects all databases. With Version 8, the access mode specified by `DIRECTIO` is only read once for each database, upon upgrading or first creation of a database. Thereafter, the I/O access mode must be changed per database using the database settings.

Changing or Preserving the I/O Access Mode

With Version 8.1, the I/O access mode is a database setting stored in the security file, rather than a server-wide `essbase.cfg` file setting that affects all databases. The `essbase.cfg` configuration-file setting `DIRECTIO TRUE|FALSE` is maintained for backward compatibility with Version 7 FixPak 8. It is also

used to provide a default value for newly created databases and for databases that are upgraded from an earlier release.

If a `DIRECTIO` setting exists in the `essbase.cfg` file at upgrade time, only newly created or upgraded databases will be affected by the setting. DB2 OLAP Server reads the I/O access mode specification from `essbase.cfg` once for each database, and copies that information to the security file (`essbase.sec`). Thereafter, if you want to change the I/O access mode used by any database, you must change it at the database level using the database setting.

The I/O access mode can be set from Application Manager (**Database Settings > Storage** tab), MaxL (**alter database set io_access_mode**), or programmatically using the Application Programming Interface. For more information, see the *Database Administrator's Guide*, the MaxL documentation in the *Technical Reference*, or the *API Reference*.

If you want to use the no-wait I/O of an operating system, select direct I/O as the DB2 OLAP Server I/O access mode. DB2 OLAP Server attempts to use no-wait I/O, when available, as long as direct I/O is the I/O access mode. To determine whether DB2 OLAP Server is using no-wait I/O at a particular time, view the database information in Application Manager (**Database Information > Storage** tab), MaxL (**display database**), or programmatically using the Application Programming Interface. For a list of platforms on which DB2 OLAP Server supports no-wait I/O, see Table 24 on page 135.

If you set a database to use direct I/O, DB2 OLAP Server will attempt to use direct I/O the next time the database is started. If direct I/O is not available on the platform at the time the database is started, DB2 OLAP Server will use buffered I/O, which is the default. However, DB2 OLAP Server stores the I/O access mode you selected as a setting in the security file, and will attempt to use that I/O access mode each time the database is started.

Understanding How Cache Sizes Are Affected by an Upgrade

When you upgrade, your cache sizes for existing databases will not change. If you are currently running Version 7.1 up through FixPak 7 and using the default I/O access mode (direct), your cache sizes for existing databases are probably large, because direct I/O requires larger cache sizes. If, after you upgrade, you plan to use the default Version 8.1 I/O access mode (buffered), you should reduce the cache size settings before upgrading or before you start the upgraded database.

The following list explains default cache sizes for DB2 OLAP Server databases when upgrading to Release 6.5, for each I/O access mode:

- The data cache default size is the same for direct I/O and buffered I/O: 3 MB.

- The index cache default size is smaller for buffered I/O than for direct I/O: 1 MB for buffered I/O and 10 MB for direct I/O.
- The data file cache default size is 32 MB when direct I/O is used. The data file cache is not used when buffered I/O is in effect.
- With Version 8.1, the size of index pages is fixed at 8 K, regardless of the I/O access mode. The index page size is unchangeable.

To override the defaults after upgrading, change the database settings or properties before you start the upgraded database for the first time.

For more information on buffered I/O and direct I/O, see the *Database Administrator's Guide*.

Platforms On Which DB2 OLAP Server Supports No-Wait I/O

Table 24 shows the platforms on which DB2 OLAP Server supports no-wait (or asynchronous) I/O. Although no-wait I/O is not used by DB2 OLAP Server on Solaris Operating System and AIX, direct I/O is still available for those platforms.

Table 24. Platforms on which DB2 OLAP Server supports no-wait (asynchronous) I/O

Platform	Direct I/O	No-Wait I/O	Cache Memory Locking
Windows 98	Not Supported	Not Supported	Not Supported
Windows XP	Supported	Supported	Supported ¹
Windows 2000	Supported	Supported	Supported ¹
Windows NT	Supported	Supported	Supported ¹
AIX	Supported	Not Supported	Not Supported
Solaris Operating Environment	Supported	Not Supported	Supported ²
HP-UX	Supported ³	Supported	Not supported

Notes:

1. On Windows XP, Windows NT, and Windows 2000, the DB2 OLAP Server Agent must be started by an administrator or power user to enable Cache Memory Locking (optional). Specifically, a power user is one who can run processes with the SE_INC_BASE_PRIORITY_NAME privilege set.
2. After the server software is installed, the root user must run a script, `root.sh`, to enable Cache Memory Locking (optional).
3. Direct I/O not supported on HFS on HP-UX.

Considerations when upgrading from previous releases

The following additional migration considerations may apply to your upgrade situation:

- As of Version 7.1, you are no longer required to install and run DB2 OLAP Server from an account called arbor. Any user can install, but this must be the user who starts DB2 OLAP Server.
- Version 7.1 introduced Cache Memory Locking, which locks the memory used for the index cache, data file cache, and data cache into physical memory, giving the OLAP kernel priority use of system RAM. After installing DB2 OLAP Server, you can use the Cache Memory Locking feature available in Release 6.0 or later if the following prerequisites are met:
 - You are using direct I/O. Direct I/O is the default for Version 7.1, and direct I/O is the only option for that release.
 - On Solaris, this feature requires you to first run the Bourne shell script, `root.sh`, from the root account as part of the installation procedure.

Upgrading databases to Version 8.1

This section provides migration details and tells you how to upgrade databases from earlier version of DB2 OLAP Server to Version 8.1.. For the order of steps to follow, see “Steps for upgrading databases to Version 8.1”.

When does DB2 OLAP Server migrate the files?

DB2 OLAP Server migrates databases when the database is started. By default, a database is set to start when its application starts. The OLAP kernel checks for files resulting from previous unsuccessful migrations, restarting the migration if necessary.

DB2 OLAP Server migrates the `ESSxxxx.IND`, `dbname.ESM`, and `thedbname.TCT` files when the database is started. DB2 OLAP Server migrates the `ESSxxxx.PAG` file as data blocks are accessed; hence, the `.PAG` file migrates when you run the `VALIDATE` command after starting the database. After the kernel files are migrated, they are not backward-compatible with the earlier release.

Steps for upgrading databases to Version 8.1

The following steps explain how to upgrade to Version 8.1 from an earlier release on the same computer. To upgrade and migrate databases to another computer, see “Migrating applications and databases across servers” on page 138.

After migration, you can restore databases from earlier releases only from backups. Therefore, be sure to back up databases before starting to upgrade.

To upgrade to Version 8.1 on the same computer, proceed in the following order for each database:

1. Run the `VALIDATE` command in `ESSCMD` against the database. `VALIDATE` prompts you to specify a name for the error log file that it will use.

If VALIDATE returns errors, revert to a backup that is free of those errors.

2. Back up all application files, database files, and the security file.
3. If you are using LROs in a production environment, run the LISTLINKEDOBJECTS command in ESSCMD before upgrading. This command returns a list of LROs contained in the databases.
VALIDATE checks for LRO errors, which you may need information about after migrating.
4. Stop the OLAP server, if it is running.
5. Install DB2 OLAP Server Version 8.1 to the same directory as the earlier DB2 OLAP Server installation.
6. Start the DB2 OLAP Server Agent (ESSBASE.EXE).
If you want to change database settings, this is a convenient point at which to do so. If you change the settings now, you will not have to restart the database to make the settings effective. See “Determining Which I/O Access Mode to Use” on page 132 for information on the default settings.
7. Select a database or load an application.
8. Run the VALIDATE command in ESSCMD against the migrated database. VALIDATE prompts you to specify a name for the error log file that it will use.
9. If VALIDATE returns only LRO-related errors to the log file after upgrading, you must restore data from the earlier backup and re-create the LROs:
 - a. Either restore data from a backup of the database that does not contain LROs, or reload from a database export.
 - b. Restart the database in DB2 OLAP Server Version 8.1.
DB2 OLAP Server migrates the database to Version 8.1 format if the database was restored.
 - c. Run the VALIDATE command.
 - d. Re-create the LROs, using the LISTLINKEDOBJECTS output as a guide. You may need to review the output from LISTLINKEDOBJECTS manually to verify its completeness.
10. Upon successful completion, unload the database and then back up the Version 8.1 database files.

Opening Retrieval Wizard files in Query Designer

In Version 7.1 and later, Essbase Query Designer (EQD) replaces Retrieval Wizard for creating queries. If you have Retrieval Wizard (.WIZ) files, the EQD may not properly translate Retrieval Wizard subsets to EQD member filters, which could cause the following problems:

- The EQD does not interpret or acknowledge parentheses within the subsets.

- If the subset contains both AND and OR conditions, the EQD may not group the conditions as expected, which may affect the result of the retrieval.

After opening Retrieval Wizard files, make sure the navigation panel nodes define the member filters as you want them. If necessary, manually promote, demote, or move nodes around or add new nodes.

Windows system file updates

On Windows platforms, the DB2 OLAP Server installation program updates the client or server system environment to run Version 8.1 software. Microsoft system files are installed to the Windows system directory (for example, C:\Winnt\System32 on Windows NT 4.0) if the files do not already exist, or if the version shipped with DB2 OLAP Server is newer.

The following table lists Windows system files installed with each DB2 OLAP Server component on supported Windows platforms. A check mark in a column indicates that the system file is installed with the specified DB2 OLAP Server component. These files will be installed to your system directory only if the files do not already exist, or if older versions exist.

Table 25. System files installed with each DB2 OLAP Server component

File	OLAP Server	Application Manager	Spreadsheet Add-in	Runtime Client	API	File Version
ATT.DLL			Yes			2.00.7024
CTL3D32.DLL		Yes				2.31.000
MFC42.DLL			Yes	Yes	Yes	4.21.7022 ¹
MSVCRT.DLL	Yes	Yes	Yes	Yes	Yes	5.00.7022 ²
MSVCIRT.DLL	Yes	Yes	Yes	Yes	Yes	5.00.7022

Notes:

1. Do not use file versions 6.00 through 6.00.8336, as these file versions may cause memory allocation problems.
2. Do not use file versions 6.00 through 6.00.8266, as these file versions may cause memory allocation problems.

Migrating applications and databases across servers

To move databases to a different computer, or to upgrade to Version 8.1 on a different computer manually, proceed in the following order for each database:

1. Run the VALIDATE command in ESSCMD against the database you will migrate. VALIDATE prompts you to specify a name for the error log file that it will use.

If VALIDATE returns errors, revert to a backup that is free of those errors.

2. Back up all application files, database files, and the security file on the original server (that is, the server from which you are migrating).
3. Install DB2 OLAP Server on the new server computer. If you are upgrading to a newer release of DB2 OLAP Server, install that release to the new server.
4. Copy the ESSBASE.CFGfile from the ESSBASE\BINdirectory on the original OLAP server to the same directory on the new OLAP server using the file system.
5. On the new OLAP server, define disk volumes. To allocate a new volume, use SETDBSTATEITEM 23 in ESSCMD and follow the prompts. You can also use the Database Settings dialog box in Application Manager or the **alter database** statement in MaxL.

Consider carefully how you configure your disk volumes. Any changes you make to your disk volumes settings after you have loaded data on the new OLAP Server are reflected only in new data loads; changes are not retroactive.

6. Using Application Manager, MaxL, or ESSCMD, create new applications and databases on the new OLAP Server.

Note: You can use the Security Migration Tool or Essbase Administration Services to migrate applications and databases (without re-creating them) from one OLAP Server to another, regardless of platform. For example, you can develop an application on a Windows server and migrate it to a production server running UNIX. When you migrate applications and databases across servers using Administration Services, the following information is copied to the target server: all databases and outlines, OLAP objects, user/group security for the application or database (including filter associations), and security filters.

The names of the applications and database you create do not need to be the same as the ones on the original server. However, if you make changes to the names, make sure that these changes are reflected as necessary in script files, spreadsheet macros, and API-based applications. In addition, make sure that these changes are clearly communicated to the user base.

CAUTION:

Do not move the application directory to the new server through a file transfer via the operating system or via FTP.

7. On a client or server computer that has connectivity to both the original and new OLAP Servers, open Application Manager.

8. Select File > Open, click Client, and select the original server, application, and database.
9. Open the original outline (.OTL) file, and select File > Save As.
10. Save the .OTL file to the new application and database on the new server.
At this point, you should avoid making changes to the outline if you want to be able to import the data file that you will export from the original application.
11. Move all calc scripts, rules files, and report scripts to the new server using one of the following procedures:
 - Repeat steps 8 through 11 of this procedure.
 - Move the files to the new server via the file system or via FTP.

If you are migrating between different server platforms, be sure to use the procedure described in steps 8 through 11 of this procedure. If you are migrating between the same server platforms, calc script and report script files can be moved using the operating system. If you move these files via the operating system, check to make sure that they function properly after moving them.

CAUTION:

Data load rules files are binary files and should always be migrated as described in steps 8–11.

12. Move any ESSCMD or MaxL scripts to the new server using the file system or via FTP. You can use the MaxL ESSCMD-to-MaxL script conversion utility.
13. If you are migrating a Version 8.1 database to another computer and the database uses linked reporting objects, export the LROs so you can re-import them to the migrated database. Use the **export lro** and **import lro** MaxL statements.

Note: If you decide to clear data from the original database, note that any LROs you want to migrate must be exported before data is cleared. When data is cleared, LROs are also cleared.

14. Export data from the application on the original server in one of the following ways:
 - Choose All Data, Input Level Data, or 0 Level Data, depending on the size of your data set as well as how the database was initially loaded.
 - Use the initial data load files that were used to populate the database on the original server.
15. Import data to the new application on the new server.
16. Recalculate your database if:
 - You loaded input-level data from data load files, or

- You exported only partial data from the database (for example, if you chose "Level 0" or "Input Level" data only)
17. If you exported any linked-reporting objects (LROs) from the original database, re-import them to the migrated database using the **import lro** MaxL statement.
 18. Repeat these steps for all other databases on the original server that you want to migrate to the new server.
 19. Migrate security information by recreating user filters, groups, and permissions on the new server.

Note: If you are migrating Version 8.1 databases from one server to another, you can use the Security Migration tool to migrate security information.

CAUTION:

Moving the security file (ESSBASE.SEC and its backup ESSBASE.BAK) between computers is not recommended or supported.

Migrating DB2 OLAP Integration Server

To help you create and execute your migration plan, this chapter provides information about platform support and hardware requirements, as well as planning for migrating OLAP outlines, databases, models, metaoutlines, and open database connectivity (ODBC) definitions.

Before you install DB2 OLAP Integration Server, back up all files from earlier releases. This will enable you to revert to an earlier release level if you encounter problems.

ODBC drivers on Windows

If you are using the MERANT ODBC drivers supplied with a previous release of DB2 OLAP Integration Server on Windows NT, 2000, or XP, you must upgrade the drivers to version 4.0 when you install DB2 OLAP Integration Server Version 8.1. See "Supported ODBC drivers and relational data sources" on page 39 for a complete listing of the ODBC drivers supported per database and operating system.

Java Runtime Environment

AIX and HP-UX: DB2 OLAP Server Version 8.1 supports Java Runtime Environment (JRE) 1.2.2, whereas DB2 OLAP Integration Server Version 8.1 requires JRE 1.3. If both DB2 OLAP Server and DB2 OLAP Integration Server are installed on same computer and the same user ID starts both servers, you must set the Java environment variables associated with each server so that, depending on whether you are logged on as the DB2 OLAP Server user or the DB2 OLAP Integration Server user, the correct version of Java is referenced.

For examples of how you might set the JAVA_HOME and related variables to point to the version of Java required by OLAP Server and Integration Services, see Chapter 6, “Installing and configuring Java” on page 73.

Supported JDBC drivers for XML import/export

JDBC drivers are required to support the XML Import/Export functionality in DB2 OLAP Integration Server.

JDBC Type 1 is installed when you install the Java Runtime Environment With the exception of Oracle, JDBC Type 1 is compatible with all RDBMSs supported by DB2 OLAP Integration Server. For information on supported JDBC drivers, see “Supported JDBC Drivers for XML Import/Export in OIS” on page 42.

Migrating existing OLAP metadata catalogs

If you have an existing OLAP Metadata Catalog from a previous release of DB2 OLAP Integration Server, you must update it before using OLAP Metadata Catalog with the new version of the software.

Attention: If you have installed the sample application from a previous release of DB2 OLAP Integration Server, you should back up the existing sample database, OLAP Metadata Catalog, and the OLAP models and metaoutlines stored in the catalog. You can then upgrade the existing catalog to be compatible with the current release of the software, as described in this section. You cannot, however, store new sample OLAP models and metaoutlines in the previous catalog.

CAUTION:

After you have updated an OLAP Metadata Catalog, you cannot roll back to the previous release. The new release of OLAP Metadata Catalog is not compatible with previous releases of DB2 OLAP Integration Server. In addition, do not attempt to use the Version 8.1 catalog with previous releases of DB2 OLAP Integration Server. Doing so could result in corrupted OLAP Metadata Catalog data.

To update an existing OLAP Metadata Catalog, complete the upgrading steps as described in this guide. When connecting to the relational database, make sure that you use the same user name and password as you used when you created the original OLAP Metadata Catalog.

The OLAP Metadata Catalog Setup dialog box is displayed at the initial logon to Essbase Integration Services Console. You can use this dialog box to upgrade an existing OLAP Metadata Catalog automatically.

Security Migration Tool

The Security Migration Tool is a utility that migrates OLAP Security data. It allows the user to migrate data from one server onto another server, as well as across platforms. The tool also allows the user to migrate security information on applications, databases, filters, users, and groups. The tool updates the `essbase.sec` file on the destination server. It generates log files, which include information on the servers and data chosen for migration, as well as a data file that includes the migrated security data. The data file can be copied onto a destination server and used to migrate data, and used as a backup of security data. Additionally, The Security Migration tool migrates the data files associated with the applications and databases, including outlines and reports.

Main features of the Security Migration tool

The Security Migration tool in Version 8.1:

- Automatically migrates security data from a source server onto a destination server, where the servers can be of either the same version of DB2 OLAP Server, or of different versions.
- Migrates security data from multidimensional databases.
- Migrates applications, databases, filters, users, and groups without the need for user interaction.
- Extracts the security data onto a text file.
- Notifies the user of any migration failures in log files.
- Migrates data between platforms by using the data file as a transfer medium.
- Allows the user to select the applications to migrate.
- Migrates both the databases and outlines.
- Can be run on a client machine that does not have DB2 OLAP Server installed or running.

Supported platforms

The Security Migration Tool is supported on AIX and Windows.

The versions of the supported platforms depends on the operating system level requirements of the DB2 OLAP Server products to be accessed by the Security Migration Tool.

Running the Security Migration tool

To run the Security Migration tool:

1. Download the installation program from the DB2 OLAP Server Web site:
www.ibm.com/software/data/db2/db2olap

Store the download file into a temporary directory.

2. Unzip the download file. Run the setup.exe file to start the installation program.
3. Select the migration option. This will:
 - Migrate security data from a source server directly onto a destination server.

This will run a complete migration. Note that most servers must be up and running. One of the servers must be on the machine where the Security Migration tool is located, and both servers must be on the same operating system.
 - Copy the security data from a source server, and store it in a data file.

The data file can be used as a reference and for later use to migrate data to another server.
 - Migrate security data onto a destination server using a data file.

This process uses a data file to migrate data onto a server on the current machine. The data file must be in the same directory as the Security Migration Tool program. Note that the server must be up and running.
4. Enter the server login information and security data options to migrate, such as applications, databases, filters, users, and groups. This includes the following:
 - a. The OLAP directory of the server (this is the top level of the OLAP directory).
 - b. The server name.
 - c. Login information (used to log into the server).

You cannot use an IP address; you must use a server name. The server name(s) must be recognized by the server you are using. The following files should include the server names:

- On Windows: WINNT/system32/drivers/etc/hosts
 - On UNIX: /etc/hosts
5. If the migration option was selected, or if the migration will be using a data file, you must choose whether or not to select specific applications to migrate.
 6. A progress/summary panel is presented that lists the server(s) for the migration and a list of the security data to be migrated. This panel remains until the migration is complete.
 7. A results panel displays after completion. This panel shows the options that migrated successfully, and errors that occur.

Note: If there is a server login error, you can go back and re-enter the login information.

Migration output

The following table shows examples of the messages you might receive at the end of a migration process.

Table 26. Output from migration

Migration .log file	Definition
secmgr.log	This will list in general what was migrated successfully (for example, "Applications migrated successfully"), and note what errors occurred. It will also reference the results.log file for error details.
results.log	Contains details of the data that was migrated both successfully and unsuccessfully, including error numbers.
essbase.log	The messages output to the Security Migration Tool by the source and/or destination OLAP servers.
connectXX.log	The connection log for each API version accessed by the Security Migration Tool. This can be used in case there is an error in locating the messages database file, or for referencing the OLAP path entered for the server by the user. If one of the servers used is of an incompatible version, the error code will be noted in this file.
data.txt	Contains all of the data chosen to be migrated.

Other notes on migration

When currency databases are migrated, some of the database settings are not set (i.e., the number of dimensions, currency time, currency category, etc.). Once the outlines and databases are migrated, these settings will be updated.

Users and groups can be migrated without applications and/or databases, but if the users or groups are added before the applications and databases, their association with those applications/databases must be manually updated.

Existing applications, databases, filters, users, and groups on the destination server are not overwritten or migrated. If they already exist, they will not be re-created.

When migrating from one OLAP version to another (i.e., from OLAP 1.1 to OLAP 7.1), the Security Migration Tool will create a data file that contains the source server data, and use that data file to migrate the data onto the destination server. The user must have write permission on the directory where the Security Migration Tool is stored, and there must be sufficient space within it for creating the data file.

The user must have supervisor privileges on the OLAP destination server.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will

be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licenses of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J74/G4
555 Bailey Avenue,
P.O. Box 49023
San Jose, CA 95161-9023
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks

1-2-3	Language Environment
AIX	Net.Data
AS/400	Notes
DB2	OS/2
DB2 OLAP Server	OS/390

DB2 Universal Database	RS/6000
DRDA	S/390
IBM	SecureWay
OpenEdition	OS/400

The following terms are trademarks or registered trademarks of other companies:

Lotus and 1-2-3 are trademarks of Lotus Development Corporation in the United States, or other countries, or both.

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

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

Tivoli and NetView are trademarks of Tivoli Systems Inc. in the United States, other countries, or both.

UNIX is a registered trademark in the United States, other countries or both and is licensed exclusively through X/Open Company Limited.

Other company, product, or service names may be trademarks or service marks of others.

Glossary

This glossary defines terms that are used in this book and throughout the IBM DB2 OLAP Server library.

accounts dimension. A dimension type that makes accounting intelligence available. You can tag one dimension as an accounts dimension, but an accounts dimension is not required.

administrator. A person who is responsible for installing and maintaining DB2 OLAP Server and for setting up user accounts and security.

agent. A process that starts and stops applications and databases, manages connections from users, and handles user-access security.

alias ID table. A table created by DB2 OLAP Server in your relational database that contains a mapping of multidimensional alias table names to ID numbers allocated by DB2 OLAP Server.

alias ID view. A view created by DB2 OLAP Server in your relational database that contains one row for each multidimensional alias table used with a relational cube. There is one alias ID view for each relational cube.

alias name. An alternate name for a dimension or member.

anchor dimension. A dense dimension that is specified as the dimension that DB2 OLAP Server uses to help define the structure of the fact table it creates in the relational cube for a multidimensional database.

API. Application programming interface. The DB2 OLAP Server API is a library of functions that you can use in a custom C or Visual Basic program to access DB2 OLAP Server.

Application Manager. A tool that you can use to create and maintain OLAP applications.

block. A string of data elements recorded or transmitted as a unit.

cache. A component of memory. Each multidimensional database contains a data cache and an index cache.

calculation. An equation within a database outline, a calculation script, or a report script that calculates a value for a particular member or point in a report.

calculation script. A text file that contains instructions to perform calculations within a multidimensional database. Also called a calc script.

Commit Block parameter. A parameter on the Transaction page of the Database Settings notebook in the Application Manager that you use to set the number of blocks that can be changed before DB2 OLAP Server commits the blocks.

cube catalog table. A table that DB2 OLAP Server creates in your relational database that contains a list of all the multidimensional databases that are stored in your relational database. The cube catalog table also shows the application with which each cube is associated. Each time that you create a multidimensional database, DB2 OLAP Server creates a new row in this table.

cube catalog view. A view that DB2 OLAP Server creates in your relational database that allows an SQL user to access a list of OLAP applications and relational cubes.

cube table. A table that DB2 OLAP Server creates in your relational database that contains a list of dimensions in a relational cube and information about each dimension.

cube view. A view that DB2 OLAP Server creates in your relational database that allows an

SQL user to access the names of all dimensions in a relational cube and associated information for each dimension. There is one cube view for each relational cube in your relational database.

data load. The process of populating a multidimensional database with data. Loading data establishes actual values for the values of the cells defined in the database outline for the database.

data load rules. A set of operations that DB2 OLAP Server performs on data as it is loaded from an external source file.

database administrator. A person responsible for administering a relational database.

database log file. A set of primary and secondary log files consisting of log records that record all changes to a database. The database log file is used to roll back changes for units of work that are not committed and to recover a database to a consistent state.

database managed space (DMS). Space in a table that is managed by the database.

database name. The name of the relational database where you want DB2 OLAP Server to store your multidimensional data.

database outline. The structure that defines all elements of a database within DB2 OLAP Server. It contains definitions of dimensions and members, dense or sparse dimension tags and attributes, the anchor dimension attribute, calculations, shared members, and alternations to the basic roll-up structure of the database.

database password. The password for the user ID that you want DB2 OLAP Server to use to log on to your relational database.

database section. A section in the Relational Storage Manager configuration file that contains values for parameters that override the values for the same parameters for the current application.

database settings. Settings that you can modify to improve performance and space utilization for

your relational database. You can change database settings by using utilities or commands supplied with your relational database management system.

database user ID. The user ID that you want DB2 OLAP Server to use to log on to your relational database. The default is the supervisor ID that you specify when you start DB2 OLAP Server for the first time.

dense dimension. A dimension with a high probability for occupying one or more data points in every combination of dimensions that occurs.

dimension. A data category, such as time, accounts, products, or markets. In a multidimensional database outline, the dimensions represent the highest consolidation level.

dimension table. A table that DB2 OLAP Server creates in your relational database that contains detailed information about the members in a dimension. There is one dimension table for each dimension in an outline.

dimension view. A view that DB2 OLAP Server creates in your relational database that allows an SQL user to access information about members contained in a dimension.

ESSCMD. A command-line interface used to perform server operations interactively or through a batch file.

generation name. A unique name that describes a generation in a database outline.

generation table. A table that DB2 OLAP Server creates in your relational database that contains generation numbers and names for each named generation specified when you created the outline. There is one generation table for each dimension in an outline.

level name. A unique name that describes a level in a database outline.

member. A discrete component within a dimension. For example, January 1997 or 1Qtr97 are typical members of a Time dimension.

metaoutline. A template containing the structure and rules for creating a database outline from an OLAP model. Using the metaoutline, you can create a database outline and load data into it. The DB2 OLAP Integration Server desktop includes a tool called OLAP Metaoutline that you can use to create one or more metaoutlines from an OLAP model.

model. A logical model (star schema) that you create from tables and columns in a relational database. You can use the OLAP model to create a metaoutline that generates the structure of a multidimensional database. The DB2 OLAP Integration Server desktop includes a tool called OLAP Metaoutline that you can use to design and create an OLAP model based on a relational data source.

multidimensional data. The data in a multidimensional database. Data can include basic data values (loaded from an external source) that represent combinations of the lowest level of members in the dimensions of the database; data values that are calculated from the base data values; and rolled up data values that are created by combining values for members in dimension hierarchies.

multidimensional database. An OLAP database that you create using the Application Manager or ESSCMD commands (in DB2 OLAP Server) or with the DB2 OLAP Integration Server desktop, which is in the Starter Kit. An OLAP database includes a database outline, data, associated optional calculation scripts, optional report scripts, and data load rules. DB2 OLAP Server stores the actual data and a shadow of the database outline in tables in a relational database.

named pipes. An API used for special node-to-node applications and particularly for access to communications and database servers.

OLAP application. An application that you create using the Application Manager or the

ESSCMD commands (in DB2 OLAP Server) or with the DB2 OLAP Integration Server desktop, which is in the Starter Kit. An OLAP application can contain one or more multidimensional databases and any associated calculation scripts, report scripts, and data load rules.

OLAP metaoutline. See *metaoutline*.

OLAP model. See *model*.

online analytical processing (OLAP). A multidimensional, multi-user, client server computing environment for users who need to analyze consolidated enterprise data in real time. OLAP systems feature zooming, data pivoting, complex calculations, trend analyses, and modeling.

outline. See *database outline*.

RDBMS. Relational database management system. A database that can be perceived as a set of tables and manipulated in accordance with the relational model of data.

relational attribute. A characteristic of a dimension table, represented by a column. You can run SQL statements against the data in relational attribute columns.

relational cube. A set of data and metadata stored in a relational database that together define a multidimensional database. A relational cube is similar to a multidimensional database, but refers to the portion of a multidimensional database that is stored in a relational database.

relational database. A database that is organized and accessed according to relationships between data items. A relational database contains a collection of relational tables, views, and indexes.

report script. An ASCII file that contains Report Writer commands that generate one or more production reports. Report scripts can be run in batch mode, using the ESSCMD command-line interface, or through the Application Manager. The script is a text file that contains data retrieval, formatting, and output instructions.

restructure. An operation to regenerate or rebuild the tables and views that DB2 OLAP Server created in your relational database.

shadow. Information stored in relational tables that shadows the information stored in a multidimensional database outline.

shared member. A member that explicitly shares storage space with another member of the same name. This member has an attribute that designates it as shared. Shared members prevent making extra calculations on a member that appears in more than one location in a database outline.

sibling. A child member at the same branch level.

sparse dimension. A dimension with a low percentage of available data positions filled. For example, a product that is not sold in all of a company's available markets would be a good sparse candidate.

Spreadsheet Add-in. Software that merges with Microsoft Excel and Lotus 1-2-3 to enable analysis of multidimensional databases. The software library appears as a menu Add-In to the spreadsheet and provides such features as connect, zoom-in, and calculate.

SQL application. An application that uses SQL statements. You can use SQL applications to access data in a relational cube.

SQL. Structured Query Language. A standardized language for defining and manipulating data in a relational database.

star schema. The type of relational database schema used by DB2 OLAP Server. When you use the Application Manager to create a multidimensional database, DB2 OLAP Server creates a main fact table and a set of dimension tables. The fact table holds the actual data values for the database, and the dimension tables hold data about members and their relationships.

star view. A relation view that DB2 OLAP Server creates in your relational database that

allows an SQL user to access data from the star schema in a single view with the JOIN already done.

table space. An abstraction of a collection of containers into which database objects are stored. A table space provides a level of indirection between a database and the tables stored within the database. A table space:

- Has space on media storage devices assigned to it.
- Has tables created within it. These tables will consume space in the containers that belong to the table space. The data, index, long field, and LOB portions of a table can be stored in the same table space, or can be individually broken out into separate table spaces.

time dimension. A dimension type that defines how often you collect and update data. You can tag only one dimension as Time, although you do not need to have a Time dimension.

Index

A

- Agent window
 - using in AIX, Solaris Operating Environment, and HP-UX 70
 - using in Windows 59
- application
 - automating startup on Windows 57

B

- books vii

C

- client and server compatibility 28
- client installation from file server 50
- Company name, parameter
 - for Windows 56
- compatibility with earlier releases 28

D

- data sources 97
- data sources, supported 39
- DB2 OLAP Integration Server
 - creating an OLAP metadata catalog 105
 - data sources 97
 - installing 48, 64
 - new function 11
 - publications vii
 - sample applications 111
 - what's new 1
- DB2 OLAP Server
 - automating startup on Windows 57
 - installing on AIX, Solaris Operating Environment, and HP-UX 62
 - installing on Windows 45
 - License Enablement 58, 70
 - new function 1
 - publications vii
 - sample application 111, 117
 - sample applications 111
 - starting on Windows 56
 - stopping on Windows 57
 - what's new 1
- documentation vii

E

- environment variables
 - for database clients 64

H

- hardware requirements 31

I

- installing DB2 OLAP Server
 - before 25
 - getting started 31
 - information needed before starting 45
 - installation steps for DB2 OLAP Server on AIX, Solaris Operating Environment, and HP-UX 62
 - installation steps for Windows 46
 - installing clients from file server 50
 - OLAP Miner 49
 - on AIX, Solaris Operating Environment, and HP-UX 62
 - on Windows 45
 - requirements 31
- installing, license key built during 45

J

- JDBC drivers supported 42

L

- License Enablement, DB2 OLAP Server 58, 70
- license key, how created 45
- local client installation 45
- locally installed clients 50

M

- metadata catalog
 - about 105
 - creating 106

N

- network client installation 50

O

- ODBC driver
 - on Windows 85

- ODBC drivers supported 39
- OLAP metadata catalog
 - about 105
 - creating 106
- OLAP Miner
 - installing on Windows 49, 65
- operations, managing server
 - commands for AIX, Solaris Operating Environment, and HP-UX 70
 - commands for Windows 59

P

- publications vii

R

- relational data sources 97
- relational data sources
 - supported 39
- remote installation of clients 50

S

- sample application
 - for DB2 OLAP Integration Server 117
 - for DB2 OLAP Server 111
- sample applications 111
- server and client compatibility 28
- server operations, managing
 - commands for AIX, Solaris Operating Environment, and HP-UX 70
 - commands for Windows 59
- setting up ODBC drivers
 - IBM 85
- software requirements 31
- SQL Interface
 - on Windows 85
- startup for Windows
 - applications, automatic startup 57
 - DB2 OLAP server 56
 - server, automatic startup 57
- system administration
 - commands for AIX, Solaris Operating Environment, and HP-UX 70
 - commands for Windows 59
- System password, parameter
 - for Windows 56

Y

Your name, parameter
for Windows 56

Contacting IBM

If you have a technical problem, please review and carry out the actions suggested by the product documentation before contacting DB2 OLAP Server Customer Support. This guide suggests information that you can gather to help DB2 OLAP Server Customer Support to serve you better.

For information or to order any of the DB2 OLAP Server products, contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

If you live in the U.S.A., you can call one of the following numbers:

- 1-800-237-5511 for customer support
- 1-888-426-4343 to learn about available service options

Product Information

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

<http://www.ibm.com/software/data/db2/db2olap/>

The DB2 OLAP Server World Wide Web pages provide current DB2 OLAP Server information about news, product descriptions, education schedules, and more.

<http://www.ibm.com/software/data/db2/db2olap/library.html>

The DB2 OLAP Server Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information.

Note: This information may be in English only.

[http://www.elink.ibm.com/public/applications/](http://www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi)

[publications/cgibin/pbi.cgi](http://www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi) The International Publications ordering Web site provides information about how to order books.

<http://www.ibm.com/education/certify/>

The Professional Certification Program from the IBM Web site provides certification test information for a variety of IBM products.

Note: In some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.



Program Number: 5724-OLP



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC27-1228-00

