

IBM<sup>®</sup> DB2<sup>®</sup> OLAP Server<sup>™</sup>



# AS/400 Administrator's Guide

*Version 1.1*



IBM<sup>®</sup> DB2<sup>®</sup> OLAP Server<sup>™</sup>



# AS/400 Administrator's Guide

*Version 1.1*

**Note**

Before using this information and the product it supports, be sure to read the information in "Notices" on page 101.

**First Edition (June 2000)**

This edition applies to Version 1.1 of DB2 OLAP Server for AS/400 and to any subsequent releases unless otherwise indicated in new editions or technical newsletters. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

Your feedback is important in helping to provide the most accurate and high-quality information. Address your comments to:

IBM Corporation  
BWG/G2  
555 Bailey Avenue  
P.O. Box 49023  
San Jose, CA 95161-9023  
U. S. A.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1998, 2000. All rights reserved.

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

© 1997-1999 ShowCase Corporation. All Rights Reserved.

---

# Contents

<b>Introduction</b> . . . . .	<b>v</b>	Disaster Recovery Backups . . . . .	17
What's in This Book . . . . .	v	After Restoring Server Information . . . . .	19
What's Not in This Book . . . . .	vi	Changing Physical Files That Have Dependent Data Views . . . . .	20
<b>Chapter 1. Starting and Stopping the Servers.</b> . . . . .	<b>1</b>	Changing Data Views That Have Dependent Data Views . . . . .	20
Warehouse Manager. . . . .	1	<b>Chapter 4. Warehouse Manager Maintenance</b> . . . . .	<b>21</b>
Starting Warehouse Manager Server . . . . .	1	Using TCP/IP with DB2 OLAP Server for AS/400 . . . . .	21
Ending Warehouse Manager Server . . . . .	1	Configuring Warehouse Manager TCP/IP Support . . . . .	21
OLAP Server Component . . . . .	1	Finding Available Port Numbers . . . . .	22
Starting the OLAP Server Component. . . . .	2	Migrating Users to a New Warehouse Manager Library . . . . .	22
Ending the OLAP Server Component . . . . .	2	Checking Current Usage of a Warehouse Manager Library . . . . .	22
<b>Chapter 2. Controlling and Managing Performance</b> . . . . .	<b>3</b>	Ending Warehouse Manager TCP/IP or Prestarted Jobs . . . . .	23
Controlling Level of Parallelism . . . . .	3	Ending ShowCase TCP/IP . . . . .	23
Assigning the Level of Parallelism . . . . .	3	Ending ShowCase Prestarted Jobs. . . . .	23
Enabling Expert Cache . . . . .	5	Removing Warehouse Manager TCP/IP or Prestarted Jobs . . . . .	24
Enabling Expert Cache . . . . .	5	Removing ShowCase TCP/IP . . . . .	24
Tuning System Performance . . . . .	5	Removing ShowCase Prestarted Jobs. . . . .	24
DB2 Symmetric Multiprocessing for OS/400 . . . . .	6	Removing a Warehouse Manager Library . . . . .	24
Optimizing Performance for Warehouse Builder . . . . .	6	Removing Warehouse Manager Server Libraries . . . . .	24
Choosing the Most Efficient Type of Distribution . . . . .	6	Backing Up a Warehouse Manager Library. . . . .	25
Using Pull Distributions . . . . .	7	Save the User-Defined Stored Procedure Declarations . . . . .	26
Creating Libraries . . . . .	7	Save Server Information . . . . .	26
CLRPFM Instead of Mass Deletes . . . . .	7	Restoring a Warehouse Manager Library . . . . .	28
Sending Only Changed Records. . . . .	7	Restore the Stored Procedure Declarations . . . . .	28
Other Performance Tuning Issues/Solutions . . . . .	9	Restore Server Information . . . . .	28
Optimizing OLAP Performance . . . . .	9	Detecting and Removing Obsolete Information . . . . .	30
OLAP Performance Tuning . . . . .	9	Restricting Access to a Warehouse Manager Library. . . . .	31
Designing an Optimized SQL Data Load Plan . . . . .	9	Applying the Restricted State . . . . .	31
Optimizing the OLAP Calculation Process . . . . .	10	Removing the Restricted State . . . . .	32
<b>Chapter 3. Managing Information on the AS/400</b> . . . . .	<b>13</b>	Working with Journals . . . . .	33
Moving Data from a Test to Production Environment . . . . .	13	Warehouse Manager and AS/400 Work Management. . . . .	33
Save Data and Then Restore It . . . . .	13		
Re-Direct Data as Needed . . . . .	14		
Create Custom Library Lists for Users . . . . .	15		
Backing Up Data . . . . .	15		
Temporary Backups . . . . .	15		

Work Management Controls . . . . .	33	Loading DB2/400 Data Into OLAP . . . . .	69
Security Issues . . . . .	40	Setting Up Your AS/400 to Allow SQL	
User Exit Programs . . . . .	40	Loads . . . . .	69
Distributed Relational Database		Loading Data Using the SQL Interface	
Architecture (DRDA) . . . . .	42	Support . . . . .	69
<b>Chapter 5. OLAP Maintenance . . . . .</b>	<b>45</b>	Substitution Variables in Rule File SQL	
Understanding OLAP Server Configuration	45	Statements . . . . .	70
Using OLAP Configuration File Settings . . . . .	45	Flat File Support . . . . .	70
Using OLAP Environment Settings . . . . .	46	Using Warehouse Builder with OLAP . . . . .	71
Understanding the Configuration and		Dimension Build and Data Load Rejected	
Environment Relationship . . . . .	48	Records . . . . .	71
Using OLAP National Language Support		Reloading Rejected Records . . . . .	72
(NLS) . . . . .	48	Naming the OLAP Server . . . . .	72
Migrating to the Euro CCSID . . . . .	49	Security . . . . .	72
Displaying the Current EBCDIC CCSID . . . . .	49	ALLOBJ Authority . . . . .	72
Displaying the Current ASCII CCSID . . . . .	50	Integrated Security . . . . .	73
Changing the ASCII and EBCDIC CCSIDs . . . . .	50	Setting up OLAP Integrated Security . . . . .	73
Understanding OLAP File System Usage . . . . .	51	Setting Up Group-Level Control . . . . .	77
OLAP Installation Library and Path . . . . .	52	OLAP Jobs . . . . .	81
OLAP QSYS.LIB Objects Created During		SQL Drill-Through Using Essbase Add-ins . . . . .	81
Installation . . . . .	52	Message Logging . . . . .	81
Specifying IFS Syntax for QSYS.LIB Objects	53	Backing Up OLAP Files . . . . .	81
Specifying IFS Syntax for Root File System		Restoring OLAP Files . . . . .	82
Objects . . . . .	53	<b>Appendix A. Commands . . . . .</b>	<b>83</b>
Finding Files Within the Integrated File		AS/400 Commands . . . . .	83
System (IFS) . . . . .	53	Warehouse Manager Commands . . . . .	84
Using QSYS.LIB Files from a 5250 Display		OLAP Server Commands . . . . .	85
Session . . . . .	54	<b>Appendix B. Sample Database . . . . .</b>	<b>89</b>
Using ESSCMD/400 . . . . .	55	RSTSMPLEB (Restore Sample Library) . . . . .	89
Using ESSCMD/400 in Interactive Mode	56	The ORDERS Table . . . . .	89
Using ESSCMD/400 in Batch Mode . . . . .	57	The CUSTOMERS Table . . . . .	91
Using an ESSCMD/400 Input Script File	58	The PRODUCTS Table . . . . .	93
ESSCMD/400 Script File Differences . . . . .	58	The SALESORG Table . . . . .	94
Using an ESSCMD/400 Output or Error		The SALESREPS Table . . . . .	95
File Specified from Within ESSCMD . . . . .	61	The DATETYPES Table . . . . .	96
Getting Help ESSCMD Commands . . . . .	62	The TIMEDIM Table . . . . .	97
Using ESSCMD API Toolkit . . . . .	62	The SALESREPS Table . . . . .	97
Introduction . . . . .	62	The EMPLOYEES Table . . . . .	98
Overview . . . . .	62	<b>Notices . . . . .</b>	<b>101</b>
Command Structure . . . . .	62	Trademarks . . . . .	102
Auto-logoff Considerations . . . . .	63	<b>Glossary . . . . .</b>	<b>103</b>
Supplemental Commands . . . . .	64	<b>Index. . . . .</b>	<b>119</b>
Output Processing . . . . .	66		
Controlling Error Handling When			
Programming . . . . .	66		
Debugging . . . . .	67		
Maintaining OLAP Log Files . . . . .	68		
Index Cache Size . . . . .	68		

---

## Introduction

Welcome to the DB2 OLAP Server for AS/400 *AS/400 Administrator's Guide*. To help you get started using this book, this introduction provides the following information:

- What you'll find in this book and what it intends to accomplish.
- What you won't find in this book and alternative places to locate this information.

---

## What's in This Book

This book provides a map to the information and assistance available for . It includes pointers to information shipped with the product, such as books and online help, and information on contacting ShowCase Corporation via telephone, fax, and the Internet.

This book provides information about the installation and maintenance of the AS/400 components of DB2 OLAP Server for AS/400 (Warehouse Manager server and OLAP server components).

After installation is complete, read “Chapter 1. Starting and Stopping the Servers” on page 1. The chapter explains starting and stopping the servers and other additional steps required to prepare the servers for client connections.

“Chapter 2. Controlling and Managing Performance” on page 3, contains detailed information about managing performance to ensure a successful deployment.

“Chapter 3. Managing Information on the AS/400” on page 13, describes procedures necessary to maintain your Warehouse Manager library, including information on backing up and restoring, checking for usage, and configuring TCP/IP.

“Chapter 4. Warehouse Manager Maintenance” on page 21, describes typical maintenance procedures that system administrators must perform on Warehouse Manager.

“Chapter 5. OLAP Maintenance” on page 45, explains how to use and maintain your OLAP library. If you are familiar with Essbase on other platforms, we recommend reading this chapter to understand the difference in using OLAP on an AS/400 platform.

You will find a list of commands commonly used to AS/400 libraries for DB2 OLAP Server for AS/400 in “Appendix A. Commands” on page 83.

We’ve also included, in “Appendix B. Sample Database” on page 89, a detailed description of the sample database installed with Warehouse Manager server. The sample database supports the product tutorials shipped with DB2 OLAP Server for AS/400.

---

## What’s Not in This Book

This guide does not describe the following:

- Getting started and deployment information. For this information, see the DB2 OLAP Server *Using DB2 OLAP Server*.
- How to install PC client software for DB2 OLAP Server for AS/400. For this information, see the *Using DB2 OLAP Server* manual.
- Procedural tasks and steps for product features not specifically related to the maintenance of DB2 OLAP Server for AS/400 software (i.e., Warehouse Manager server and OLAP server component). For more information, see the online help in the PC version of that particular product.
- Selecting a communications protocol, setting up a PC router, or configuring TCP/IP. For this information, see the *Using DB2 OLAP Server*.



---

# Chapter 1. Starting and Stopping the Servers

This chapter outlines the procedures for starting and stopping Warehouse Manager server, Analyzer OLAP server (AOS), and OLAP server.

---

## Warehouse Manager

If you are using the APPC communications protocol, no commands are required to start or end the server. However, before you begin any maintenance you must verify that no users are connected to the library.

### Starting Warehouse Manager Server

If you are using the TCP/IP communications protocol, the Warehouse Manager TCP/IP server job should start automatically when the installation finishes, provided TCP/IP was configured correctly.

However, if you opted to set up Warehouse Manager TCP/IP after installation or if you ended Warehouse Manager TCP/IP, you must use the following procedure to start the Warehouse Manager TCP/IP server:

1. Log on as QSECOFR or with equivalent authority (\*ALLOBJ, \*IOSYSCFG, \*SAVSYS, \*JOBCTL, and \*SECADM special authorities are needed).
2. Add the Warehouse Manager library to your library list.
3. Run the DB2 OLAP Server for AS/400 command STRSCTCP.

### Ending Warehouse Manager Server

To end the Warehouse Manager TCP/IP server:

1. Log on as QSECOFR or with equivalent authority (\*ALLOBJ, \*IOSYSCFG, \*SAVSYS, \*JOBCTL, and \*SECADM special authorities are needed).
2. Add the Warehouse Manager library to your library list.
3. Run the DB2 OLAP Server for AS/400 command ENDSCTCP.

The ENDSCTCP command will terminate all current TCP/IP connections to the Warehouse Manager library. For more information on this and related topics, see “Ending Warehouse Manager TCP/IP or Prestarted Jobs” on page 23, or “Prestart Jobs” on page 36.

---

## OLAP Server Component

The following sections give instructions on how to start and end the DB2 OLAP Server for AS/400.

## Starting the OLAP Server Component

To start the OLAP server component, type: `library/STRESSSVR`

Where *library* is the library name where the OLAP server component is installed (the default is ESSBASE). This command has one parameter: the job description under which the OLAP server component will run.

If this is the first time that the OLAP server component is being run and these parameters were not set up during the install, you will get a screen requesting the following information:

1. At “*Enter your company name*”, type the name of the company to which the OLAP server component is licensed.
2. At “*Enter your initial system supervisor user ID*”, enter an OLAP server component system user ID. This user ID is independent of your OS/400 user profiles. It does not need to match an existing OS/400 user profile name, although you may choose a name that does correspond to an AS/400 user ID.
3. At “*Enter the system password*”, type a password to correspond with the user ID specified in step 2.
4. At “*Confirm the system password:*” retype the password and press **Enter**.

Your OLAP server component should now be running.

## Ending the OLAP Server Component

To stop the OLAP server, use the ENDESSSVR command.

```
ENDESSSVR SVRUSER(username) SVRPW(password)
```

Where: *username* is the OLAP server component system user ID, and *password* is the OLAP system user ID password

---

## Chapter 2. Controlling and Managing Performance

Managing the many performance tuning aspects of your relational and multidimensional databases can be a challenge for even the most experienced database administrator. This section is intended to provide insight into performance planning and tuning of both your relational and multidimensional databases.

This section is divided into sections as follows:

- Controlling Level of Parallelism
- Enabling Expert Cache
- Tuning System Performance
- DB2 Symmetric Multiprocessing for OS/400
- Optimizing Performance for Warehouse Builder
- Optimizing OLAP Performance

The first eight sections focus primarily on performance tuning various aspects of your relational database, while the last section, Optimizing OLAP Performance, focuses exclusively on performance tuning and planning your multidimensional database.

It is important to remember that performance tuning of your relational databases is significantly different from the performance tuning of your multidimensional databases. In fact, some of the tuning you do for your relational databases can have a negative impact on multidimensional databases that use that relational data as a data source.

---

### Controlling Level of Parallelism

This section only applies if you have the DB2 Symmetric Multiprocessing for OS/400 feature installed on your AS/400. Controlling the level of parallelism will determine whether the system will allow a query to be logically split into multiple independent tasks that can be carried out in parallel.

The AS/400 system value QQRYDEGREE can be used to tune the level of parallelism that is allowed when running queries on your system.

#### Assigning the Level of Parallelism

To determine the amount of parallelism that you will allow on your system, use the following procedure:

1. Consider all work that is performed on your system and decide if queries can be allowed to consume additional CPU and I/O resources in order to finish quickly.
2. Decide the amount of parallelism that you are going to allow. Currently, the system value gives you the following control:
  - **\*NONE** - Do not allow any parallelism. Queries are not the highest priority on the system and you cannot afford to allow the system to devote the resources necessary to an individual query.
  - **\*IO** - Allow IO parallelism. You are willing to let the system use additional IO and mainstore resources to complete a query faster. In some cases, allowing the system this freedom to use more resources while running a query will improve the cumulative total resources used.
  - **\*OPTIMIZE** - Allow the system to choose whether IO and CPU parallelism should be used to improve query performance and overall system performance. In some cases, allowing the system this freedom to use more resources while running a query will improve the cumulative total resources used.
  - **\*MAX** - Allow IO and CPU parallelism. You want parallelism to be used whenever possible if it will help improve the individual query. You only use this system for running queries, so there is little impact on other types of applications.
3. Once you have set the level of parallelism, review system impact and adjust as necessary until you have the setting that works best for your environment.

### **An Example**

You have three systems in your company. You run queries on all systems so you bought DB2 Symmetric Multiprocessing for OS/400 to get the additional data access methods and now want to use the appropriate level of parallelism as well. The following shows how each of your systems is used.

- **SYS-A:** You run an order entry application on this system and run occasional queries.
- **SYS-B:** This is a development and test system where you enhance your application and test your queries.
- **SYS-C:** This machine is a dedicated data warehouse system and is where most of your queries are run.

Based on analysis you choose to set the QQRYDEGREE system value as follows for each of your systems.

- **SYS-A=\*NONE:** Your order entry application is your first priority. You cannot afford to have a single query use a majority of the system resources even for a short time.

- **SYS-B=\*OPTIMIZE:** You will let the system decide. You have no mission critical applications running on this system and you do want to test queries that sometimes do use parallelism.
- **SYS-C=\*MAX:** This system is used only for decision support. You want the queries to be fast and are willing to allow the query to use any resources necessary to complete.

---

## Enabling Expert Cache

Expert cache is a set of improved paging algorithms used for database files and related objects. By enabling expert cache, you are improving query completion time and reducing system paging and faulting rates.

Enabling expert cache should be the first step in tuning your system for running queries.

### Enabling Expert Cache

1. To enable expert cache, determine the storage pool that the server job is using. Normally, this will be \*BASE which is storage pool 2.
2. On the AS/400, use the WRKSYSSTS command with the assistance level set to 2=Intermediate.
3. Press **F11** until you see the heading 'Paging Option'. Type \*CALC over the top of the existing value to enable the expert cache algorithms. Press **Enter**.
4. Alternatively you can use the CHGSHRPOOL command.

---

## Tuning System Performance

System performance tuning refers to adjusting pool sizes, time slice, and subsystem allocations to improve response time, throughput, paging rates, and contention.

System performance tuning can improve overall usage of system resources and individual query response times. However, in most cases, you must tune the *entire* system to get the real benefits. For this reason DB2 OLAP Server for AS/400 makes the following recommendations:

- Always consider the entire system and all system work when adjusting system-wide resources.
- Never adjust system resources for the server without considering the rest of the system load.
- System tuning starts with system performance analysis; do not tune the system without first doing this analysis.
- Do system performance tuning *after* following the other performance suggestions in this section.

- In most cases, it is best to hire a consultant who specializes in system tuning. System tuning can be an exercise in trial and error. A good consultant already knows what works and what doesn't in many situations.

If you decide system level performance tuning is right for your system, here are some sources of information to help you get started.

- *OS/400 Work Management Guide* (SC41-4306-00)
- *Performance Tools/400 - Getting Started* (SC41-4343-00)
- An AS/400 consultant specializing in system performance

## DB2 Symmetric Multiprocessing for OS/400

The DB2 Symmetric Multiprocessing feature improves query performance by adding the following to DB2/400:

- Additional data access methods and optimizer improvements
- CPU parallelism for query processing
- Support for the QQRYTIMLMT and QQRYDEGREE system values

Warehouse Manager server takes advantage of the functionality provided by this feature. And while its name indicates that it is for multi-processor systems, it also provides improvements on single processor machines.

## Optimizing Performance for Warehouse Builder

There are several ways to improve the performance of your Warehouse Builder distributions. The following provide a brief description of ways to improve the performance of your relational database distributions.

### Choosing the Most Efficient Type of Distribution

There are several types of distributions that Warehouse Builder provides. The type of distribution you choose will have a significant impact on the performance of your distribution and its effect on your AS/400 system. The performance characteristics of the various types of distributions is described in Table 1.

*Table 1. Distribution Types and Performance Levels*

Type of Distribution	Performance Level
Local (within the same machine)	Very good performance
Pull (from a remote AS/400 data source to a target table on the control server)	Very good performance
Push (from the control server to a remote AS/400)	Not very good performance
Pass Through (from a remote AS/400 to another remote AS/400)	Not very good performance

## Using Pull Distributions

By far, the fastest type of distribution is a pull distribution. Pulling data from a target system is between 10 to 15 times faster than pushing data to another system. This is due to the lack of blocking support in DRDA when doing inserts into a remote file. Therefore, we recommend that all performance critical distributions pull data from the source system. This means that the control server and the destination server should be the same system.

## Creating Libraries

To store distributed data on the target system, Warehouse Builder automatically creates libraries on the target system, improving distribution performance. However, to store distributed data on a server other than an AS/400, Warehouse Builder can create collections instead of libraries. Whether Warehouse Builder creates collections or libraries depends on the **Create Target SQL Collection** option in the **Defaults** tab of the **Options** dialog box. If this option is selected, Warehouse Builder will create collections, and if this option is not selected, Warehouse Builder will automatically create libraries.

## CLRPFM Instead of Mass Deletes

When the target table is defined in Warehouse Builder, you can specify that the target table is deleted on each distribution. When you do this, a SQL DELETE statement is issued to ensure the client code would work against any SQL data source. The SQL DELETE statement can be issued by selecting the Warehouse Builder **SQL DELETE** option, which is located in the **Table Overwrite** section of the **Defaults** tab in the **Options** dialog box.

On the AS/400, the other method to remove all records from a file is to issue a CLRPFM (Clear Physical File Member) command. This command can only be used with an AS/400. Using the CLRPFM command before the data distribution is started enhances performance. This method has two distinct advantages:

- The removal of old records is very fast and consumes very little system resources
- Disk space used by the deleted records is recovered

The **CLRPFM** command can be issued by selecting the **Clear Physical File Member Command** option, which is located in the **Table Overwrite** section of the **Defaults** tab in the **Options** dialog box.

## Sending Only Changed Records

Another way to improve distribution performance is to send only changed records between the source and target servers. While there is currently no automatic method to send only changed records between a source and target system (for example, as you can with products that have journal-based

replication services), the following paragraphs describe several methods to facilitate sending only changed records if you have a file that only gets inserts and does not get updates or deletes.

### **Using CURDATE()**

If there is a date in the file(s) referenced on a target system, you can use a variety of built-in SQL functions to determine what data is new since the last distribution.

For example, if you move data nightly from an orders file and that file has a date (either a date data type or a date that can be manufactured out of several fields and converted to a real date), then you can compare the date in the record with the current date (for example, `ORDERDATE = CURDATE()`).

### **Using a Sequence Number**

Another method is to use a sequence number that always grows in value to determine which records were new since the last distribution. To use this method you might:

1. Run a full file refresh to get the target primed
2. The post-transfer SQL would find the maximum sequence number and store it in a file
3. The subsequent distributions would ensure that all sequence numbers taken from the target file were greater than the sequence number stored as part of the post-transfer SQL
4. Repeat step 2 again as part of the post-transfer SQL



## Other Performance Tuning Issues/Solutions

The efficiency of the SELECT statement that runs on the source system will have a direct influence on both the distribution time and the amount of system resources used. Therefore, when SELECT statements run against large files, it is recommended you use the Performance Analyzer feature of Query to first determine if the statement is optimized and/or to get a rough idea how long the query will run, or turn on auditing and run it as a local distribution or in Query to get the auditing information logged.

Based on that information, you may either adjust the SQL statement, build indexes to speed up the query, or perform the query in multiple passes (such as break the query down into several smaller ones to reduce the complexity of the statement or perform more of the work on the target system instead of the source).

---

## Optimizing OLAP Performance

In this section we highlight important information for your OLAP implementation. There are two important areas of optimization. The first is the optimization of the OLAP server. The second is the AS/400 system on which the server is running.

### OLAP Performance Tuning

It is strongly recommended that you understand the concepts described in the *Database Administrator's Guide* and *IBM's AS/400 Work Management book* (SC41-5306-02).

Issues that are critical to the success of your OLAP implementation are load, calculation, and retrieval performance. Performance varies dramatically based on the relationships between the source data, the amount of data, the configuration of the OLAP outline, the size and complexity of your retrievals, and the tuning applied to the database settings.

### Designing an Optimized SQL Data Load Plan

To optimize an OLAP data load, consider the following:

- Sort your input records so that same sparse member combination are grouped consecutively. In addition, sort your input records so that the sparse dimensions appear in the same order as the sparse dimensions in the outline. By sorting your data on the sparse dimension members, you minimize the number of times an OLAP data block must be read from disk, decompressed, enlarged, re-compressed, and written back to disk. To sort your input data, specify the appropriate ORDER BY clause on the SQL statement in your rule file. For information on using SQL loads, refer to the *Database Administrator's Guide*.

- Load the data from a data source on the server. If the data is not located on the source, moving it to the server will improve load times since the data doesn't have to be transported over the network at load time.
- For additional information and other optimization tips, refer to Debugging and Optimizing Data Loads in Volume II of the *Database Administrator's Guide*.

## Optimizing the OLAP Calculation Process

Calculating an OLAP model is CPU intensive and decisions you make about the calculation process can affect performance. Refer to Optimizing Your Calculations in the *Database Administrator's Guide* for more information about optimizing your OLAP calculations.

### AS/400 Performance Tuning

Performance can change depending on the system resources available to the OLAP server component (CPU, number and type of disk units, and main memory), the number of users accessing the data, and the other concurrent activities on the system.

For more information on tuning your AS/400, see the IBM Work Management Guide (SC41-5306-02)

Once you understand your analyst's requirements, examine your implementation environment to determine the limits that will be placed on your solution. The following sections provide some insight into issues you should consider and raise questions you should be able to answer before continuing to the next phase of your implementation.

### Balancing Priority between Warehouse Builder and the OLAP Server

If you attempt to run Warehouse Builder distributions while running your OLAP server, be aware that these two activities compete for system resources. This is especially true if you are running on a single processor system.

Depending on how you installed Warehouse Manager, Warehouse Builder distributions run a default priority of 20. If you start the OLAP server as a typical batch job, it runs at a default batch priority of 50. Both of these tasks are AS/400 resource intensive. Since warehouse builder jobs run at a higher priority, your OLAP operations (load/calc/retrievals) will run very slowly.

To prevent slow OLAP operations, set identical tuning priorities for your OLAP server and data distributions. For example, if you run the OLAP server at priority 50, set the data distribution to priority 50 as well. In addition, set your time slice under 25 milliseconds. This will prevent starvation of one of the jobs.

## Memory

OLAP performance is greatly impacted by the amount of memory resources it is allowed to use. Just like any other OS/400 application, you can manipulate the standard OS/400 work management techniques to control memory allocations for the OLAP server component. On the AS/400, memory is allocated to various memory pools by OS/400. For example, memory can be allocated to the \*BASE pool, \*INTERACTIVE pool, or other user defined pools. For more information on memory pools, consult the IBM Work Management Guide (SC41-5306-02).

You can run under the \*BASE pool or under a dedicated memory pool. The important thing is the memory pool is sufficiently large and that it does not have excessive competition from other applications running on your AS/400. We suggest starting with a memory pool of 256 megabytes for small databases being accessed by a limited number of users. As you increase the number of databases, the size of the databases, or the number of users, consider increasing the amount of memory available for OLAP.

## Grouping of Databases into Applications

Typical OLAP installations run many OLAP databases on a single AS/400. While you can group several databases into a single application, a single application and database for each business problem is almost always the best choice for implementation to simplify administrative tasks.

## OLAP and SMP

Symmetric Multi-processing (SMP) usually refers to a system with more than one main processor. For example, some AS/400 models can be equipped with 2, 4, 8, or more processors. In this case, SMP is a feature of the hardware. Multiple processors let the system simultaneously process more than one task.

On the AS/400, SMP can also refer to a DB2/400 software offering that can improve DB2 query performance regardless of whether the system has more than one main processor. This software offering gives OS/400 the enhanced ability to split the work to process a single query into several steps that can be run simultaneously in separate jobs.

The OLAP server component is implemented as a multi-threaded program. Usually this allows OLAP to take advantage of multiple processors.

Currently, the OLAP calculation process of a given database is single threaded and makes use of only a single processor. To use multiple processors during the calculation of a database, subdivide the database by creating *transparent partitions*. You can then take advantage of multiple processors by calculating each partition at the same time. For more information on transparent partitions, refer to *Database Administrator's Guide*.

Since each database is calculated under a separate thread, you can take advantage of multiple processors when calculating more than one database at the same time. Performance will vary depending on other AS/400 resources such as the amount of available memory and the performance of your disk units. Performance also depends on other activity on the system.

If you have a single processor system, limited memory, or limited I/O capacity (few disk arms), you will achieve better results by calculating only one database at a time.

OLAP takes advantage of multiple processors while processing end user retrievals. This applies to retrievals submitted using the spreadsheet add ins, Analyzer, or user-written report scripts. For each retrieval, OLAP will use 2 threads. As multiple users run simultaneous retrievals, OLAP uses multiple threads to process these retrievals. Since OLAP is using multiple threads, it takes advantage of SMP and the work is spread across multiple processors.

---

## Chapter 3. Managing Information on the AS/400

Warehouse Manager server enables support for the Warehouse Builder. Data produced by these products is stored on the AS/400. You will need to protect your DB2 OLAP Server for AS/400 information by performing tasks involving the Warehouse Manager server. The following tasks are described in this section:

- Moving Data From a Test to Production Environment
- Backing Up Data
- Changing Physical Files That Have Dependent Data Views
- Changing Data Views That Have Dependent Data Views

Also, see “Chapter 4. Warehouse Manager Maintenance” on page 21 for more information.

---

### Moving Data from a Test to Production Environment

Query, Report Writer, and Warehouse Builder give you the powerful advantage of developing queries, reports, data models, data views, and data distributions directly against the AS/400 database. The environment in which you develop these files is critical to the success of database operations on a daily basis. If you develop these files in a production environment, you could adversely affect the performance of other operations. To prevent this problem, develop against test data, either on a separate AS/400 machine or in a separate library. Once testing is complete, you can put your queries, reports, data models, data views, and/or data distributions into production.

There are various ways to “move” data from a test environment to a production environment, as described below:

#### Save Data and Then Restore It

This is a global approach, allowing you to do a complete backup of all information that resides on the AS/400. This approach is the most efficient as it allows you to move all the information at once, but it limits you from moving individual queries, reports, data views, and so forth. Use the Save/Restore procedure to save and back up server information from one library and restore it to another. The target library can be on a different system or on the same system. This procedure is outlined in “Chapter 1. Starting and Stopping the Servers” on page 1. See “Backing Up Data” on page 15 for backup recommendations.

## Re-Direct Data as Needed

With this approach, you re-direct a query, report, data view, and so forth from a test environment to run in a production environment. You do this on a file-by-file basis, which is helpful if you do not want to move all data at once by doing a save and restore. However, this method requires more planning and structure to be successful. The key here is to keep the structure of your libraries, tables, columns, and column data types consistent in both your test and production environments.

Sometimes, it is necessary to modify the database structure by renaming libraries or tables, or by removing or renaming columns. However, it is highly recommended to keep structural modifications to a minimum. If you find you're doing a lot of it, it's probably time to do a thorough database design review. Existing queries, for example, may cease to run after these kinds of changes. Change Link error messages are invoked when PC client applications determine that libraries, tables, or columns are missing, or if column data types have changed.

Consider the following two scenarios to help you decide how to structure your AS/400 database:

### Scenario 1: Test and Production Are Done on Separate AS/400s

If test and production are done on separate AS/400s, maintain identical database structure on both systems. That is, the library, table, and column names should match, and the column data types should be equivalent (e.g., numbers to numbers, characters to characters) on both systems. For example:

From			
SYS1	FY98	CUSTOMERS	CUSTNAME
SYS1	FY98	ORDERS	SHIPDATE
To			
SYS2	FY98	CUSTOMERS	CUSTNAME
SYS2	FY98	ORDERS	SHIPDATE

After testing is complete, use the Change Data Source feature to switch a query, report, data view, data model, or data distribution to run on the production system. This feature is available in PC client applications for DB2 OLAP Server for AS/400.

Use the Performance Analyzer feature to evaluate the performance of a query on your production AS/400. This feature is available in Query and Report Writer.

## Scenario 2: Test and Production Are Done on the Same AS/400

If test and production are done on the same AS/400, maintain a set of test tables in a test library, parallel to the production library. The same concerns regarding matching names and data types apply here, except for the library names, which will be different. For example:

SYSTEM	LIBRARY	TABLE	COLUMN
From			
SYS1	SAMPLE	CUSTOMERS	CUSTNAME
SYS1	SAMPLE	ORDERS	SHIPDATE
To			
SYS1	FY98	CUSTOMERS	CUSTNAME
SYS1	FY98	ORDERS	SHIPDATE

After testing is complete, use the Change Link feature to convert a query, report, data view, data model, or data distribution to run against the production library. This feature is available in the PC client applications for DB2 OLAP Server for AS/400.

### Create Custom Library Lists for Users

Another approach for moving data from a test to a production environment is to create custom library lists for users. In this case, developers/testers run against a library list that points to test libraries; users rerun against a library list that points to production libraries. To give users access to only those AS/400 libraries in their library lists, choose the **Server Options** command in Warehouse Manager client. In the **Server Options** dialog box, choose the **Library Access** tab and specify "Return only libraries in user portion of the library list." This simplifies the database for end-users because all the data they need to work with is contained in only a few libraries. However, it's difficult to troubleshoot database errors because tables are implicit in the libraries, thus making the exact location of data less obvious.

---

## Backing Up Data

It is strongly recommended that you back up the Warehouse Manager server periodically. How you should back up this server depends entirely on the purpose of the backup. Will you be doing a temporary backup so that you can, for example, move server information between systems? Or will you be doing a backup as part of your disaster recovery?

### Temporary Backups

The following situations require you to back up Warehouse Manager server information temporarily:

- You need to install a new release of DB2 OLAP Server for AS/400 into the Warehouse Manager server library.
- You need to delete the Warehouse Manager server library, but want to save the library's contents, such as security settings, queries, and data views.
- You need to move the Warehouse Manager server library from a test system to a production system.
- You need to move Warehouse Manager server contents from a test library to a production library on the same system.
- You need to distribute the contents when you buy a license for an additional AS/400.

For any of these situations, use the Save/Restore procedure to save and back up server information from one library and restore it to another. (See “After Restoring Server Information” on page 19 for configuration issues you will need to consider once Warehouse Manager server information has been restored.) The target library can be on a different system or on the same system.

Using this procedure, you have the choice of saving server information to a save file or device, such as a tape. In these situations, a save file is recommended. If you are moving data between systems, it is easiest to ship save files electronically. Especially for remote sites, saving to a file is more cost-effective than mailing tapes. If you cannot send save files electronically between systems, you can always back up to a device.



## Disaster Recovery Backups

To back up Warehouse Manager server information for disaster-recovery purposes, you should follow the same procedure you do for other backups. Whether you use the AS/400 GO SAVE command to back up the entire system or the SAVLIB command to back up individual libraries, you should always back up to a tape or other media device and then store the media offsite.

If you typically back up the entire system, Warehouse Manager server information will be backed up when you do the system save. For complete details about an entire system backup, refer to the IBM document *OS/400 Backup and Recovery*. If you have Web access, go to <http://as400bks.rochester.ibm.com>. Once at this site, choose the version of your OS/400 and then find the book *OS/400 Backup and Recovery*, which contains backup information specific to your OS/400 version.

If you typically just back up individual libraries, simply include the Warehouse Manager library (e.g., SCSEVER) in the list of libraries that you already back up. To save your libraries, follow the steps below:

1. On an AS/400 command line, type SAVLIB and press **F4**.
2. Enter the names of the libraries that you usually back up, plus the Warehouse Manager library name.
3. In the Device prompt, enter the name of the device to save to (e.g., TAP01 if you're saving to a tape).

If you need to restore information from a Warehouse Manager library that was saved in this way, recommends that you restore this information from the device to a different library. That is, the name of the library that you restore to should not be the same name as the original library. If you restore to the original library, that library could be corrupted and you will lose data.

To restore the Warehouse Manager library, follow the steps below:

1. On an AS/400 command line, type RSTLIB and press **F4**.
2. In the Saved library prompt, enter the name of the Warehouse Manager library to be restored.
3. In the Device prompt, enter the name of the device from which to restore this library.
4. Press **F9** to display more prompts and then page down to the next screen.
5. In the Restore to library prompt, enter the name of a different library (e.g., SCLIB) where you want to restore Warehouse Manager server information. This library should not be the original Warehouse Manager library.

After restoring the original Warehouse Manager library (e.g., SCSEVER) to another library (e.g., SCLIB), you should install a fresh copy of DB2 OLAP

Server for AS/400 into the original library. The installation procedure is described in the *Using DB2 OLAP Server* manual.

To recap, to this point you have restored the SCSEVER library from a tape back to a library called SCLIB, and then re-installed DB2 OLAP Server for AS/400 into a brand new SCSEVER library. (From this point forward, the original Warehouse Manager library is referred to as SCSEVER, the other library as SCLIB.)

Next, we recommend saving the server information in the SCLIB library and restoring it using the Save/Restore procedure. That is, you will want to run the SAVSRVINF command from the SCSEVER library against the SCLIB library, and then run the RSTSRVINF command from the SCSEVER library to restore the information.

See “Save Server Information” on page 26 and “Restore Server Information” on page 28 for detailed steps of the Save/Restore procedure. An abbreviated version of this procedure, as it relates to disaster-recovery backups, is described below:

1. Add the SCSEVER library to your library list. On an AS/400 command line, type ADDLIB SCSEVER and press **Enter**.
2. Save the server information from the SCLIB library. On an AS/400 command line, type SAVSRVINF and press **F4**. This command will run from the new SCSEVER library you installed. Enter the following prompts:
  - ShowCase server library = Type SCLIB.
  - Device = Type \*SAVF to save the data to a save file. A save file is easiest since you will be keeping the server information “live” instead of archiving it to tape.
  - Save file = Type the save file name and the name of the library where you want the save file to be created.  
The save file will be created if it doesn’t already exist.
3. Press **Enter** to save the information.
4. Restore the server information to the SCSEVER library. On an AS/400 command line, type RSTSRVINF and press **F4**. This command will run from the new SCSEVER library you installed. Enter the following prompts:
  - Device = Type \*SAVF to restore the data from a save file.
  - Save file = Type the name of the save file to restore from and the name of the library where you created that file.
5. Press **Enter** to restore the information.

## After Restoring Server Information

After running RSTSRVRINF to restore Warehouse Manager server information, you may need to do additional configuration so your server will work correctly.

If you restored Warehouse Builder information, consider the following:

- If you restored the information to an AS/400 different from the one from which it was saved, you may want to change several things about your distribution sets and definitions. For example, if you were performing “pulls” of data from one AS/400 to a test system, and you moved the distribution information to your production system, change the relational database (RDB) name of the target system so that the “pull” will now pull data to your production system. In Warehouse Builder, choose the **Definition Properties** command on the **Tools** menu to make these changes.
- If you restored to a different AS/400, you should also ensure that APPC and/or TCP/IP communications are configured and active between the new system and the systems defined as source and target servers for the distributions.
- If you restored to a new AS/400 or to a new library on the same AS/400, any schedule information defined for your distribution sets and any scheduled queries/reports are not restored. In Warehouse Builder, choose the **Work with Warehouse Builder Sets** command on the **Sets** menu to edit your sets and set up schedules.
- If you restored Warehouse Builder information into a server library already containing distribution definitions, notice that the object IDs of the distribution sets have changed. If you are running the sets from your own CL programs and you have coded the object IDs in either the STRDD or SBMDD command, the hard-coded references may need to be changed.

If you restored Warehouse Manager administration or Query information, consider the following:

- You may display the physical file RSTSRVPF to see the actions performed on the server catalog files.
- If you restored Warehouse Manager administration or Query information to a different AS/400, or if your environment has changed substantially since setting up this information, do a catalog cleanup. In the Warehouse Manager PC application, choose the **Clean Up ShowCase Catalogs** command on the **Manage** menu.

---

## Changing Physical Files That Have Dependent Data Views

Query enables you to create and manage simplified views of any AS/400 database. This simplified view, or data view, is saved on the AS/400 as an SQL view. Since SQL views are saved on the AS/400 and reference physical files (also known as tables), you may not be able to change physical files which have dependent views. Query includes several pre-created queries and reports you can run to find out which data views are built over a given AS/400 table. You can run the queries and reports from the **Tools** menu in Query.

If you find that there are data views dependent on a physical file you need to change, you can use the CHGPF command, which allows you to change a physical file without deleting dependent views. As a precaution and for recovery purposes, back up your views before using CHGFP.

To do this, use the Save/Restore procedure. For more information see “Save Data and Then Restore It” on page 13. That is, you will want to run the SAVSRVRINF command on the AS/400 to save data views. In this case, leave the Delete Views After Saving prompt at \*NO when running the SAVSRVRINF command.

---

## Changing Data Views That Have Dependent Data Views

Just as data views can reference physical files, data views can also reference other data views. If you built a data view over another view, you cannot delete, modify, or replace the base view because it has a dependent view. For example, if VIEW2 is built over VIEW1, you cannot delete, modify, or replace VIEW1. You have three options, all of which can be done within Query:

- Delete the dependent view, VIEW2, if that view is no longer needed.
- Edit the dependent view to remove the dependency.
- Create a copy of the base view, VIEW1, and give the copied view a unique name (e.g., VIEW1TEMP). Change the dependent view, VIEW2, to use the new view, VIEW1TEMP. This removes the dependency on VIEW1. You are now free to make your changes to VIEW1. After completing the changes to VIEW1, change link VIEW2 back to using VIEW1. You can then delete the temporary view, VIEW1TEMP.

For complete details about data view tasks, see the Query Online Help.

---

## Chapter 4. Warehouse Manager Maintenance

This chapter discusses typical maintenance procedures that system administrators must perform on the Warehouse Manager server.

---

### Using TCP/IP with DB2 OLAP Server for AS/400

DB2 OLAP Server for AS/400 software can be used in a Transmission Control Protocol/Internet Protocol (TCP/IP) network environment without the need for a SNA APPC connection to the AS/400. Similar to APPC, DB2 OLAP Server for AS/400 client and server products can communicate with each other using connections through a TCP port.

On the PC, you must have either Windows NT or Windows 95 or later to use TCP/IP.

Using TCP/IP support provides the following benefits:

- If your primary network is TCP/IP, the administration of DB2 OLAP Server for AS/400 is easier because you only have to be concerned with a single network architecture.
- The TCP/IP stack comes packaged with OS/400, Windows NT, and Windows 95; therefore, you do not need to purchase other APPC router software.
- If you want to change the TCP port number after the install, you can use the RMVSCTCP command to disable TCP, and then use the ADDSCTCP command to enable it with the correct port number. This makes it easier to migrate from a test environment to a production environment. See “Migrating Users to a New Warehouse Manager Library” on page 22.

---

### Configuring Warehouse Manager TCP/IP Support

This section explains how to configure Warehouse Manager TCP/IP if it was not configured during installation. Before configuring TCP/IP, verify that the following prerequisites are met:

- Warehouse Manager only supports TCP/IP on Windows 95 or later and Windows NT.
- OS/400 TCP/IP must be configured and active.

To enable Warehouse Manager TCP/IP support, run the command ADDSCTCP. The command adds information to the subsystem, creates objects, defines the port, and runs STRSCTCP.

The following additional commands are available in the server library to help users maintain Warehouse Manager TCP/IP support:

- RMVSCTCP, which runs ENDSCTCP, removes information from the subsystem and deletes TCP/IP-specific objects.
- STRSCTCP, which starts the “manager job.”
- ENDSCTCP, which ends the “manager” and “client” jobs (who the client is talking to).

### **Finding Available Port Numbers**

A TCP/IP port number is a numeric alias for an application. You assign a specific port to a Warehouse Manager installation library so the client software knows which TCP/IP server application to connect to. To enable Warehouse Manager for TCP/IP, you must assign a unique port number for each library.

If you chose to configure TCP/IP during the installation, the next available port is automatically selected for you.

To view a list of the ports that are in use, run the AS/400 command WRKSRVTBLE.

The command ADDSCTCP assigns the port number 43419 by default. If this number is already in the display after you run the command WRKSRVTBLE, you must choose another port number. The next available number should be fine.

### **Migrating Users to a New Warehouse Manager Library**

At times, it may be necessary to have your DB2 OLAP Server for AS/400 users change libraries. For example, you may install a new release of Warehouse Manager and first verify that it works in your test environment. You then want users to quit using the old release and begin using the new one. In an APPC network, the set-up is difficult. Users’ ODBC data source administrator for APPC uses the Warehouse Manager library name. Since Warehouse Manager does not allow you to just rename the library, each user would have to change his or her ODBC configuration.

However, in a TCP/IP network, users’ client configuration uses the TCP/IP port to determine which Warehouse Manager installation to connect to. This allows you to easily remove the port number from the previous library and then change the port number of the new library to the port number of the previous library. This way, users can automatically connect to the new library without having to make any changes to their client configuration.

---

### **Checking Current Usage of a Warehouse Manager Library**

The following procedures will enable you to check whether ShowCase TCP/IP or prestarted jobs are active.

For Warehouse Manager libraries, run the following command: `WRKOBJLCK OBJ(server-library/SCCONNECT) OBJTYPE(*USRSPC)` Where *server-library* is the name of the library you want to check

If TCP/IP is active, the screen shows the ShowCase TCP/IP Manager job for the library (job name `SCSERVER`, which is the same as the library name) and may show TCP/IP connections (job name `SCTCPSRV`). If prestart jobs are active, the screen shows the prestarted jobs (job name `RSCGWY0`).

If you need to see which users are connected (e.g., to ask them to disconnect before an install), use Work With Job Locks (option 8). You should see a lock on the user profile (type `*USRPRF`). (If you see `QUSER`, this indicates no user is connected.) From the user profile name, you should be able to determine the people using the DB2 OLAP Server for AS/400 product. Alternately, you can use the Warehouse Manager client software to view active connections.

---

## Ending Warehouse Manager TCP/IP or Prestarted Jobs

Before ending Warehouse Manager TCP/IP or prestarted jobs, verify that no one is using the library. (Refer to “Checking Current Usage of a Warehouse Manager Library” on page 22 for instructions. If you see jobs named `RSCGWY0`, then prestarted jobs are enabled. If you see the library name in the list of jobs with locks, or jobs named `SCTCPSRV` are listed, ShowCase TCP/IP is enabled.)

### Ending ShowCase TCP/IP

To end ShowCase TCP/IP, run the command `ENDSCTCP` from the library in which you want to end ShowCase TCP/IP support. The command will by default end ShowCase TCP/IP clients and stop the ShowCase TCP/IP manager job for the library.

### Ending ShowCase Prestarted Jobs

To end prestarted jobs:

1. Run the command `ENDPJ`.
2. In the Subsystem prompt, type the name of the subsystem used during the installation of ShowCase prestarted jobs.
3. In the Program prompt, enter `RSCGWY0`.
4. In the Library prompt, enter the name of the ShowCase library you are working with.
5. Press **Enter**.

It may take 30 seconds or more for the jobs to be completely ended.

---

## Removing Warehouse Manager TCP/IP or Prestarted Jobs

Before removing Warehouse Manager TCP/IP or prestarted jobs, verify that no one is using the library. (Refer to “Checking Current Usage of a Warehouse Manager Library” on page 22 for instructions. If you see jobs named RSCGWY0, then prestarted jobs are enabled. If you see the library name in the list of jobs with locks, or jobs named SCTCPSRV are listed, then ShowCase TCP/IP is enabled.)

### Removing ShowCase TCP/IP

To remove ShowCase TCP/IP, run the command RMVSTCP from the library from which you want to remove ShowCase TCP/IP support. The command will end ShowCase TCP/IP and clean up the system by removing all entries associated with subsystems for this library, all spool file entries for ShowCase TCP/IP jobs for the installation library, and all TCP/IP service/port information for the installation library.

### Removing ShowCase Prestarted Jobs

To remove prestarted jobs:

1. End the prestarted jobs (see “Ending Warehouse Manager TCP/IP or Prestarted Jobs” on page 23 for instructions).
2. To remove the prestarted jobs from the subsystem, run the command RMVPJE
3. In the Subsystem prompt, enter the name of the subsystem used during the installation of ShowCase prestarted jobs.
4. In the Program prompt, enter RSCGWY0.
5. In the Library prompt, enter the name of the ShowCase library you are working with.
6. Press **Enter**.

---

## Removing a Warehouse Manager Library

You may occasionally need to remove AS/400 libraries installed on your system, such as when you want to remove test libraries.

### Removing Warehouse Manager Server Libraries

1. Log on as QSECOFR or equivalent authority (\*ALLOBJ, \*IOSYSCFG, \*SAVSYS, \*JOBCTL, and \*SECADM special authorities are needed).
2. Back up any Warehouse Manager settings that you want to save. See “Backing Up a Warehouse Manager Library” on page 25 for instructions.
3. Verify that no users are connected to the library.
4. Restore the UNINSTSC utility from the UNINSTSC save file which exists in the Warehouse Manager server library and enter the command:  
RSTOBJ OBJ(\*ALL) SAVLIB(QTEMP) DEV(\*SAVF) SAVF(SERVER\_LIB/UNINSTSC) RSTLIB(QSYS)



Where *SERVER\_LIB* is the name of the DB2 OLAP Server for AS/400 library that will be removed

5. Enter the ShowCase command UNINSTSC.
6. Type the name of the Warehouse Manager library you want to remove.
7. To remove the library, press **Enter**.

The UNINSTSC command must be run interactively. It cannot be submitted to batch.

---

## Backing Up a Warehouse Manager Library

Because of the way Warehouse Manager is installed, the installation library on your AS/400 should not be backed up through conventional means. Following is the series of procedures you should use to back up a Warehouse Manager server library.

## Save the User-Defined Stored Procedure Declarations

This task is required only if you've added stored procedures to the library.

This procedure only saves the stored procedure definitions. You will have to save the stored procedure programs and the library containing the programs separately.

To back up the user-defined stored procedure declarations:

1. Log in with a user ID that has \*ALLOBJ authority. Warehouse Manager is installed with \*PUBLIC authority set to EXCLUDE, so you will need \*ALLOBJ authority to work with Warehouse Manager.
2. Add the DB2 OLAP Server for AS/400 library to your AS/400 library list; ensure that you only have one DB2 OLAP Server for AS/400 library in your library list. Run the command `ADDLIBLE server-library`, where *server-library* is the name of the DB2 OLAP Server for AS/400 library with which you want to work.
3. Save the stored procedure information by typing the command `SAVSP` and pressing **F4**.
4. Specify which stored procedures to save, where to save them from, and where to save them to.
  - Type the names of the stored procedures to back up, or type \*ALL to back up all stored procedures.
  - Type the name of the server library that you are saving from.
  - Type the name of the device to save to, or type \*SAVF to save to a save file. If you enter \*SAVF, you will be prompted for the save file name and the AS/400 library where you want to create the save file.
5. Type the lowest target release of the OS/400 on which you will be restoring and using the stored procedures.
6. When finished with all the prompts, press **Enter** to save the stored procedures.

## Save Server Information

This task is required only if you are using any of the following applications:

- Warehouse Manager (server options such as AS/400 library access and default public authority to AS/400 libraries and tables; security and resource settings, aliases and column attributes; and application security)
- Query (data views)
- Warehouse Builder (such as data distribution sets and definitions)
- Analyzer applications (such as forms, models, pinboards, and views)

The Warehouse Manager, Query, Warehouse Builder, and Report Writer applications store information on the Warehouse Manager server on the AS/400. Analyzer applications store information, such as views and

pinboards, on the Analyzer OLAP server on the AS/400. The information on these servers should be saved and backed up using the SAVSRVRINF command located in the DB2 OLAP Server for AS/400 library and then restored using the RSTSRVRINF command.

The Save and Restore feature, which consists of these two commands, is also useful if you need to move data from one server library to another. See “Moving Data from a Test to Production Environment” on page 13 and “Backing Up Data” on page 15 for recommendations on moving data from a test to production environment and backing up ShowCase data.

To back up information which STRATEGY applications store on the Warehouse Manager server:

1. Log in with a user ID that has \*ALLOBJ authority.
2. Add the DB2 OLAP Server for AS/400 library to your AS/400 library list; ensure that you only have one DB2 OLAP Server for AS/400 library in your library list. Run the command ADDLIBLE server-library, where *server-library* is the name of the library with which you want to work.
3. Save the server information by typing the command SAVSRVRINF and pressing **F4**.
4. In the ShowCase Server Library prompt, type the name of the server library from which server information should be saved. This library is typically SCSEVER.
5. In the Device prompt, type the name of the device to save to, or type \*SAVF to save to a save file. Press **Enter**. If you entered \*SAVE, type the save file name and the AS/400 library where you want to create the save file.
6. In the Type of Information to Save prompt, type one of the following:
  - \*ALLto save all of the information described in the bullets below
  - \*ADMIN to save only Warehouse Manager administration information (such as security settings and aliases)
  - \*AOS to save only Analyzer OLAP Server information (such as forms, models, pinboards, reports, and views) and Information Delivery applications' user information (such as shared queries)
  - \*DISTto save only Warehouse Builder information (such as data distribution sets and definitions)
  - \*DATAVIEWto save only Query data viewsEven if you save all information, you can restore individual groups of settings.
7. You can press **F10** to see additional prompts for the SAVSRVRINF command. These prompts do not need to be changed under most circumstances. However, the prompts are described below in case they might be useful to you:

- In the Target Release prompt, type \*PRV if you intend to restore saved information on the previous release of the OS/400, or leave the default of \*CURRENT if you intend to restore saved information on the same or a later release of the OS/400.
  - In the Delete Views After Saving prompt, specify whether or not to delete data views after saving. It is recommended that you leave this option at \*NO. However, if you need to change physical files which have dependent views, type \*YES to delete the data views on your system. (The views can later be restored with the RSTSRVRINF command.) Another option is to use the CHGPF command, which allows you to change a physical file without deleting dependent views.
8. Press **Enter** to save the server information.

---

## Restoring a Warehouse Manager Library

Following is the series of procedures you should use to restore a Warehouse Manager server library.

### Restore the Stored Procedure Declarations

To restore the stored procedures that were saved with the SAVSP command:

1. Add the library to your AS/400 library list; ensure that you only have one library in your library list. Run the command ADDLIBLE *server-library*, where *server-library* is the name of the DB2 OLAP Server for AS/400 library with which you want to work.
2. Restore the stored procedure information by typing the command RSTSP and then pressing **F4**.
3. Specify which stored procedure definitions to restore, where to restore them to, and where to restore them from.
  - Type the names of the stored procedures to restore, or type \*ALL to restore all stored procedures.
  - Type the name of the server library that you are restoring to.
  - Type the name of the device to restore from, or type \*SAVF to restore from a save file. If you enter \*SAVF, you will be prompted for the save file name and the AS/400 library where you created the save file.
4. When finished with all the prompts, press **Enter** to restore the stored procedures.

### Restore Server Information

To restore the server information that was saved with the SAVSRVRINF command:

1. Distribute your save file or move your tape or other device to the target system, if it is different from the first system.
2. Add the DB2 OLAP Server for AS/400 library to your AS/400 library list; ensure that you only have one library in your library list. Run the

command `ADDLIBLE server-library`, where *server-library* is the name of the library with which you want to work.

3. Restore the server information by typing the command `RSTSRVRINF` and pressing **F4**.
4. In the Device prompt, type the name of the device to restore from, or type `*SAVF` to restore from a save file. Press **Enter**. If you entered `*SAVF`, type the save file name and the AS/400 library where you created the save file.
5. Choose whether or not to restore Warehouse Manager administration information. Press **Enter**. If `*YES`, the Run Catalog Check Program prompt displays. Choose one of the following values for this prompt:
  - `*NO` to not run the catalog check program on the restored information.
  - `*AUTODELETE` to delete any records that do not apply to objects found on the system.
  - `*LISTONLY` to just create a list of records that do not apply to objects found on the system. You can later delete these records by running the `CHKSRVRCAT` command and specifying `*YES` on the `DLTORPHAN` parameter.

To view the results of a catalog cleanup operation, use the Warehouse Manager client application. On the **Manage** menu, choose the **Clean Up ShowCase Catalogs** command, and then choose the **View Results** button to display the results.

6. Choose whether or not to restore Analyzer OLAP Server information. Press **Enter**. If `*YES`, the Duplicate User Option and Restore Configuration File prompts display. Choose one of the following values for the Duplicate User Option prompt:
  - `*KEEP` to keep the existing AOS data in the target user-permission file. That is, existing system manager privileges and user settings will not be replaced with any duplicate AOS data being restored.
  - `*REPLACE` to replace the existing AOS data in the target user-permission file with any duplicate AOS data being restored.

Choose one of the following values for the Restore Configuration File prompt:

- `*YES` to replace the existing AOS configuration file with the saved (`SAVSRVRINF`) version of this file.
  - `*NO` to not restore this file and thus not overwrite the existing AOS configuration file.
7. If you specified `*YES` for either the Restore Warehouse Manager Information or Restore AOS Information prompt, the Duplicate Record Option prompt displays. This prompt specifies the action restore takes if it finds any saved Warehouse Manager or AOS data that duplicates data in the target server library. Choose one of the following values:

- \*KEEP to keep the existing data in the target server library. That is, the existing data will not be replaced with any duplicate data being restored.
  - \*REPLACE to replace the existing data in the target server library with any duplicate data being restored.
8. Choose whether or not to restore Warehouse Builder information. Press **Enter**. If \*YES, the Restore Logs prompt displays. Choose whether or not to restore the logging and message information from previous runs of the data distribution sets and definitions you saved.
  9. Choosing to restore Warehouse Builder information also displays a duplicate record option. Choose whether to keep or replace duplicate records by typing \*KEEP or \*REPLACE.
  10. Choose whether or not to restore Data View information.
  11. You can press **F10** to see additional prompts for the RSTSRVRINF command. The following prompt does not need to be changed under most circumstances. However, it is described below in case it might be useful to you.
    - In the Catalog File to be Processed prompt, the default \*ALL restores all catalog files that contain administration information. These files contain application resource settings, security settings, ShowCase aliases, and column attributes. Alternately, you can specify a single catalog file to restore. Press **F1** on this prompt to display the help, which lists the DB2 OLAP Server for AS/400 catalog files and their contents.
  12. Press **Enter** to restore the server information. Several status messages will flash by and then a screen with additional configuration steps will appear. Read this information carefully, especially if you restored Warehouse Builder, Warehouse Manager, Query, or Data View information.

For example, if you are restoring Warehouse Builder information to a new server library on the same system or to a different system, it is recommended that you use Warehouse Builder to make any necessary changes, to create target tables, and to reschedule jobs. Alternately, you can use the Warehouse Builder post save/restore utility found on the product's **Tools** menu.

## Detecting and Removing Obsolete Information

When using Warehouse Manager's Alias Manager and Security Manager, you define alias names or security rules. Alias names reference existing objects on your system. If those objects change, the alias names may reference objects or fields that no longer exist. Similarly, Security Manager defines access rules for a particular object and user. In this case, not only can the referenced objects change but the users may no longer exist.

When you use Query to create data views, Warehouse Manager stores information about these data views. It is possible that these data views could be renamed or deleted by programs other than Query. The Warehouse Manager information could then reference objects that no longer exist.

Given these situations, Warehouse Manager gives you a way to re-synchronize the catalog information with your changed system. On the **Manage** menu, click **Clean Up ShowCase Catalogs**. The command allows you to determine and view information that is no longer valid, and remove it if you so decide. For additional information on the command, see the Warehouse Manager client Online Help.

---

## Restricting Access to a Warehouse Manager Library

You must put a Warehouse Manager server library in a restricted state if you plan to change the Enforce AS/400 Security option. This restricted state prevents users from connecting to the server library during the time it takes to process this change. Once a server library is in a restricted state, you can easily remove the restricted state so that users can re-connect.

### Applying the Restricted State

To put a server library in a restricted state:

1. Log on as QSECOFR or equivalent authority (\*ALLOBJ, \*IOSYSCFG, \*SAVSYS, \*JOBCTL, and \*SECADM special authorities are needed).
2. Add the DB2 OLAP Server for AS/400 library to your AS/400 library list; ensure that you only have one library in your library list. Run the command `ADDLIBLE server-library`, where *server-library* is the name of the library with which you want to work.
3. Change the server status by typing the command `CHGSRVRSTS` and pressing **F4**.
4. In the ShowCase Server Library prompt, type the name of the server library you want to put in a restricted state. This library is typically `SCSERVER`.
5. In the Restrict Server Library prompt, type `*YES` to prevent users from connecting to the server library. Only your current job (running the `CHGSRVRSTS` command) and the TCP/IP master job (if it is running) remain connected to the restricted server library.

If users are already connected to the server library, it will not be put in a restricted state unless you force the restriction.

6. Press **Enter**.
7. In the Force Server Restricted State prompt, choose whether or not to force the server library into a restricted state:
  - `*N0` to attempt to put the server library in a restricted state. However, if users are connected to the server library, the attempt will fail.

- \*YES to force all connections to the server library to end (except for the jobs listed in step 5). This will end the jobs of any users who are currently connected to the server library.
8. Press **Enter**. If you specified \*NO in step 7, the server library may or may not be put in a restricted state. If you specified \*YES, the Maximum Force Wait Time prompt displays. This prompt allows you to set the maximum time that the CHGSRVRSTS command should run while waiting for all connections to the server library to end. The default time interval is 30 seconds. After you specify a time limit, press **Enter**.

The countdown begins *after* all connections to the server library have started to close. The server will be put in a restricted state once either of the following occurs:

- No more connections are found during the specified time interval or
- The time interval expires and there are no more connections.

If the time expires and there are still connections to the server library, the server will not be put in a restricted state. In this case, you will need to retry until the restricted state is achieved so that you can change the Enforce AS/400 Security option.

Regardless of whether or not the server library is put in a restricted state, when the CHGSRVRSTS command is issued with the Force=\*YES option, all active connections will begin to close. This process cannot be reversed once it has started. If the server library fails to enter a restricted state in the time interval specified, users whose connections were ended in the process will be allowed to reconnect, as will any other users.

## Removing the Restricted State

To release a server library from a restricted state:

1. Log on as QSECOFR or equivalent authority (\*ALLOBJ, \*IOSYSCFG, \*SAVSYS, \*JOBCTL, and \*SECADM special authorities are needed).
2. Add the library to your AS/400 library list; ensure that you only have one library in your library list. Run the command ADDLIBLE server-library, where *server-library* is the name of the library with which you want to work.
3. Change the server status by typing the command CHGSRVRSTS and pressing **F4**.
4. In the ShowCase Server Library prompt, type the name of the server library that you want to release from a restricted state.
5. In the Restrict Server Library prompt, type \*NO to allow users to access the server library again.
6. Press **Enter** to remove the restricted state.



---

## Working with Journals

Warehouse Manager creates two journals when it is installed. These journals are used to maintain data integrity when ShowCase catalog files are updated. The two journals are RSCSPCTLJ, which is associated with stored procedure catalogs, and RSCSCCTLJ, which is associated with Warehouse Manager, Warehouse Builder, and Query client catalogs.

The main concern with journaled files is the space consumed by the journal receivers. To ensure the journal receivers do not consume too much space on your system, you can use one of the following approaches:

- Monitor the size of the receivers and issue a CHGJRN command when they get too large. Once the CHGJRN command is issued and new journal receivers are attached to the journal you can (optionally) save the receivers and then delete them.
- You can let the system maintain the journal receivers. Enter the commands in the following steps to do so:
  1. CRTJRNRCV JRNRCV(server-library/receiver-name) THRESHOLD(500000)
  2. CHGJRN JRN(server-library/RSCSPCTLJ) JRNRCV(server-library/receiver-name) SEQOPT(\*RESET) MNGRCV(\*SYSTEM) DLTRCV(\*YES)
  3. CRTJRNRCV JRNRCV(server-library/receiver-name) THRESHOLD(500000)
  4. CHGJRN JRN(server-library/RSCSCCTLJ) JRNRCV(server-library/receiver-name) SEQOPT(\*RESET) MNGRCV(\*SYSTEM) DLTRCV(\*YES)

---

## Warehouse Manager and AS/400 Work Management

This section describes how Warehouse Manager is influenced by AS/400 work management controls and provides information on how to adjust these controls. Before reading this section, you should be familiar with AS/400 work management concepts. For information on AS/400 work management see *Work Management*(IBM Document No. SC41-4306-00).

### Work Management Controls

Following is an overview of what you can control with each of the work management constructs as they relate to Warehouse Manager and running queries. Included is information for changing the constructs to influence Warehouse Manager behavior.

In general, you should not need to adjust the work management behavior of Warehouse Manager. DB2 OLAP Server for AS/400 is providing this information only to give you a basic understanding of how Warehouse Manager is interacting with your system and to provide the information

necessary for advanced systems management, such as system tuning or configuring job classes to control temporary storage limits.

### **Job Class**

The job class used by Warehouse Manager can be used to control the following attributes with minimum impact to other jobs. Other attributes can also be controlled through the class description, such as time slice.

- Job run priority
- Maximum temporary storage
- Default wait time

*Job run priority* can be used to control Warehouse Manager's run priority relative to other jobs on the system. If this is the only attribute you want to control, using the Resource Settings feature of Warehouse Manager client is an easier alternative.

*Maximum temporary storage* can be used to control the amount of storage a job uses at run time. Temporary storage is the storage required for Warehouse Manager and system programs at run time, internal system objects used by the system for this job, and temporary objects used when running a query.

*Default wait time* can be used to control the maximum time in seconds that the Warehouse Manager job will allow for an instruction that performs a wait to finish running. Most typically, this will be the time Warehouse Manager waits to obtain either an object or record lock while processing a query.

Use the default wait time if locks are common in your environment and are affecting your query. Reducing the wait time will allow Warehouse Manager to return record and object lock errors more quickly. Lengthening the wait time will allow Warehouse Manager additional time for the object to become available.

In most environments a wait time of 30 seconds to 2 minutes is recommended. A wait time of \*NOMAX is allowed but is not recommended as lock conditions may cause the PC to appear stopped for long periods of time.

To create a job class, use the CRTCLS command. Add or update the routing entries and prestart job entries to use the new job class.

### **Storage Pools**

The storage pool on the AS/400 is a portion of the main memory isolated for a set of jobs. All jobs using a storage pool compete for the resource but do not compete against jobs using a different storage pool. Storage pools can be used to:

- Isolate memory-intensive jobs from other jobs on the system
- Give particular jobs more memory by reducing contention
- Keep highly used objects in main memory

Use caution when creating storage pools since the memory will be subdivided and will not be used unless a job is actively using the storage pool.

More than one storage pool can be associated with a subsystem. The routing entry or prestart job entry indicates the storage pool that should be used by the job when running in the subsystem.

### **Subsystems**

The subsystem Warehouse Manager runs in can be used to control the storage pools available to the server and to configure job classes within the subsystem.

To ensure the storage pool you are using has expert cache enabled, see the “Enabling Expert Cache” on page 5.

The subsystem the server job uses is determined as follows.

**APPC:** For APPC, the subsystem Warehouse Manager jobs use is based on the subsystem to which the APPC device description is allocated. This will be either QCMN or QBASE by default on most systems. You can control this allocation by adding the appropriate communication or remote location entries to your subsystems using ADDCMNE. When altering the allocation of devices to a subsystem, keep the following in mind:

- Controlling device allocation is not trivial; make sure you have a thorough understanding of your system configuration before beginning.
- The order in which your subsystems are started influences device allocation.
- Once a device is allocated to a subsystem, it cannot be reallocated until the subsystem ends or the device is varied-off. Simply disconnecting the PC will not deallocate the device.

**TCP/IP:** TCP/IP uses the subsystem specified during the Warehouse Manager installation or with the ADDSCTCP command. This will be either QCMN or QBASE by default on most systems.

For TCP/IP it is easier to choose the subsystem the server uses; however, you do not have the flexibility to use multiple subsystems with a single Warehouse Manager library as in APPC. If you need different PC workstations in different subsystems using TCP/IP, you must have multiple Warehouse Manager libraries installed.

**Batch Queries:** Batch queries use the subsystem servicing the job queue. This job queue was specified in the job description and is used when the query is scheduled by the user.

### **Prestart Jobs**

Prestart jobs reduce the amount of time required to start a program on your AS/400, which in turn saves valuable processing time in program initialization. Prestart jobs work only when using the APPC communications protocol and typically provide the best benefit when used on a very slow system. It is not a recommended method if you have a reasonably fast machine. For more information on using prestart jobs with APPC, see “Routing Entries and Prestart Job Entries” on page 37.

Routing entries and prestart jobs are mutually exclusive, so when trying to improve performance by using prestart job entries, you do not need to configure routing entries.

A prestart job is an AS/400 job that starts running before a remote system program sends a request to start the AS/400 program. A prestart job entry tells the operating subsystem which program to start when the operating subsystem starts. The program does not run at this time. It simply performs all the preparatory steps and initializes the job named in the prestart job entry. It then waits for a request to start the program. When it receives the start request from the PC, it starts the job by using the prestart program that is ready and waiting, thus saving valuable time in program initialization.

To add prestart jobs when using APPC:

1. Record the name of your communications subsystem. If you do not know the name of the communications subsystem, run the command `DSPSYSVAL QCTLSBSD`.

The name of the controlling subsystem appears. The following table lists the controlling subsystems and the corresponding communications subsystems.

*Table 2. Controlling and Corresponding Communications Subsystems*

Controlling Subsystem	Communications Subsystem
QCTL	QCMN
QBASE	QBASE

2. To add a prestart job, type the command `server-library/ADDSCPJE`  
Where *server-library* is the library in which you installed Warehouse Manager

3. Press **Enter**, then press **F10** to view additional parameters. The Add Prestart Job Entry (ADDPJE) screen appears. Change the defaults if necessary.
4. Press **Page Down**. The remaining Additional Parameter entries appear.
5. In the Subsystem Description parameter, type the name of the communications subsystem.
6. Change the defaults if necessary.
7. Press **Enter**.

For more information on prestart jobs, see the *IBM AS/400 Programming: Work Management Guide*.

### Routing Entries and Prestart Job Entries

Routing entries and prestart job entries allow you to choose the job class used by Warehouse Manager and the storage pool within the subsystem the job uses.

**APPC Without Prestart Jobs:** For APPC *without* prestart jobs, routing entries are used to choose the job class of the server job. To add routing entries for Warehouse Manager:

1. Determine the subsystem your users will use. It will most likely be QCMN or QBASE.
2. Within the subsystem, you will need to add one or more routing entries. By default, Warehouse Manager will use the default routing entry for the mode description QPCSUPP. This will give a job class of QWCPCSUP and storage pool 1 within the subsystem.
3. The following information will help you add a routing entry. Use the APPC program start request routing data. The routing data format is shown in the following table.

*Table 3. Routing Data Format*

Data	Position	Length
Mode name	1	8
Device name	9	10
User profile	19	10
PGMEVOKE	29	8
Program	37	10
Library	47	10

*Program* will contain 'RSCGWY0' for Warehouse Manager.

*Library* will be the server installation library.

4. Choose the job class and storage pool with the subsystem to use on the routing entry. Following are several examples of routing entries that can be added.

### **Routing Entry Based on Mode**

In this case we want all jobs that use a mode of QPCSUPP to use job class MYCLASS in QGPL and storage pool 1. Note that this routing entry will apply to all programs using QPCSUPP, not just DB2 OLAP Server for AS/400 jobs.

```
ADDRTGE SBSD(QCMN) SEQNBR(1) CMPVAL(QPCSUPP 1) PGM(*RTGDTA) CLS(QGPL/MYCLASS) POOLID(1)
```

### **Routing Entry Based on User**

In this case we want each user to have his/her own job class. We will add an entry similar to the following for each user. Note that this routing entry will apply to all APPC programs for the user QUSER, not just Warehouse Manager.

```
ADDRTGE SBSD(QCMN) SEQNBR(1) CMPVAL(QUSER 19) PGM(*RTGDTA) CLS(QGPL/MYCLASS) POOLID(1)
```

### **Routing Entry Based on Warehouse Manager**

In this case we want all Warehouse Manager jobs to use their own job class.

```
ADDRTGE SBSD(QCMN) SEQNBR(1) CMPVAL(RSCGWY0 37) PGM(*RTGDTA) CLS(QGPL/MYCLASS) POOLID(1)
```

When the system searches for a routing entry, it uses the first routing entry that matches. This means you should put your most restrictive routing entries first and least restrictive routing entries last. The SEQNBR() keyword on the command allows you to control the search order used by the system when searching routing entries.

**APPC with Prestart Jobs:** When you are using prestart jobs in your subsystem, routing entries are not used. Instead, the information associated with the prestart job is used. Because of this, you do not have the flexibility of allowing different users to have different job classes when using prestart jobs.

To change the job class and storage pool used by a prestart job you can either specify the information on the ADDSCPJE command or change the information using the CHGPJE command. Prestart jobs give you the flexibility to have the first number of jobs use one job class and any additional jobs use another job class. This is done by specifying a threshold and two classes on the ADDSCPJE or CHGPJE commands.

**TCP/IP:** TCP/IP uses the job class associated with the routing entry added when DB2 OLAP Server for AS/400 was installed or when the ADDSCTCP

command was issued. By default, DB2 OLAP Server for AS/400 uses the QWCPCSUP job class for TCP/IP connections.

To change the job class used by TCP/IP, use the CHGRTGE command. The routing data format of the ShowCase TCP/IP routing data is listed in the following table.

*Table 4. ShowCase TCP/IP Routing Data Format*

Data	Position	Length
Library	1	10
SCTCPSRV	10	8

TCP/IP also has a manager on the AS/400. This job simply starts the associated DB2 OLAP Server for AS/400 client jobs when a connection is made to the AS/400 using ShowCase ODBC. The routing data for this job is as follows.

*Table 5. AS/400 TCP Manager Job Routing Data*

Data	Position	Length
Library	1	10
SCTCPMGR	10	8

**Batch Queries:** For batch queries, you can add your own routing data in the job description. When that job description is specified batch job schedule, the routing data will be passed to the system. You can set up routing entries based on the routing data you define to assign the appropriate job class.

### **Job Descriptions**

The job description allows you to control the initial library list used by the server, job log logging options, and job accounting.

**APPC:** For APPC, the job description specified in the user profile is used.

**TCP/IP and APPC with Prestart Jobs:** For TCP/IP and prestart jobs, the job description of QUSER is used initially. The library list of the job description specified in the actual user's user profile is then used to replace the library list of QUSER. The job log logging and job accounting are not updated.

**Batch Queries:** For batch queries, it is the job description specified when scheduling the batch job.

### **User Profiles**

The user profile allows you to control:

- Object level security and management attributes applied when running the server.
- The job description used for the user.
- Permanent storage limits.
- NLS attributes of the job.
- Output queue used when producing a job log or other spooled output.

Warehouse Manager will always run using the user profile specified at ODBC login when connecting to the AS/400.

When using WRKACTJOB, WRKUSRJOB, or a similar command, the job name may contain the actual user profile or it may contain QUSER depending on how the job was started. QUSER is the user profile that many DB2 OLAP Server for AS/400 jobs run under. QUSER must not have a storage limit. To ensure that QUSER has no limit, run the following command:

```
CHGUSRPRF USRPRF(QUSER) MAXSTG(*NOMAX)
```

To identify a DB2 OLAP Server for AS/400 job by user you can use one of the following techniques.

To list all jobs by user, use the command WRKOBJLCK OBJ(myuser) OBJTYPE(\*USRPRF) Where *myuser* is the user profile for which you want to list jobs

To list all DB2 OLAP Server for AS/400 jobs for a particular library, use the command WRKOBJLCK OBJ(library-name/SCCONNECT) OBJTYPE(\*USRSPC)

---

## Security Issues

This section provides information on Warehouse Manager security.

### User Exit Programs

DB2 OLAP Server for AS/400 supports the PCSACC exit program on the CHGNETA command. DB2 OLAP Server for AS/400 does support \*REGFAC, but if the PCSACC value is \*REGFAC the user will need to run the ADDEXITPGM command with SC\_QUERY\_ROW\_SEC for the Exit point parameter and SCRS0100 for the Exit point format parameter to use User Exit Programs.

A user exit program can be specified on the AS/400 to accept or reject a request before the function is actually carried out. This allows for additional user-defined security not provided by the AS/400 or Warehouse Manager. Specify the user exit program name on the PC Support Request Access (PCSACC) parameter of the AS/400 change network attributes (CHGNETA) command. For example:



This tells PC support to call this program before it executes a File Transfer, Virtual Printer, or Message function. Warehouse Manager uses the same network attribute and program parameters as PC Support before it prepares each SQL statement or before submitting a batch query.

Following are descriptions of the two parameters that are passed to the user exit program when using Warehouse Manager or PC Support. More detailed documentation on user exit programs can be found in *OS/400 Server Concepts and Administration* (IBM Document No. SC41-3740).

- Parameter one: One byte exit program return code. The request is allowed if the user exit program sets this value to hex F1 (the character “1”). The request is rejected if this parameter is set to any other value.
- Parameter two: This parameter contains information about the request. The following table describes the structure used by DB2 OLAP Server for AS/400 and is identical to the PC Support File Transfer structure.

Table 6. Structure of the Second Parameter Passed to a User Exit Program<sup>1</sup>

Value	Type	Description
User profile	Character 10	Contains the user profile making the request.
Application name	Character 10	Contains the name of the PC application making the request. <sup>2</sup>
Requested function	Character 10	Contains the type of SQL statement, SELECT, JOIN, INSERT, UPDATE, DELETE, CREATE, DROP, GRANT, etc. <sup>3</sup>
Object name	Character 10	Contains the object (table) name. <sup>4,5</sup>
Library name	Character 10	Contains the library name or *LIBL. <sup>4,5</sup>
Member name	Character 10	Contains the member name, *FIRST or *LAST. <sup>4,5</sup> *ALL is not supported by AS/400 SQL.
Record format name	Character 10	Contains the record format name. DB2 OLAP Server for AS/400 passes a value of *ONLY for the record format name. <sup>5</sup>
Length of next field	Zoned 5,0	Length of the statement in the following request.
Entire Request	char(32500)	SQL statement

Table 6. Structure of the Second Parameter Passed to a User Exit Program<sup>1</sup> (continued)

1. When developing a user exit program be aware that PC Support Virtual Print may continue using the first user exit program it calls even though you use CHGNETA to change the user exit program.
2. An asterisk for the first character of the name signifies a DB2 OLAP Server for AS/400 client application. Otherwise, the name is the ODBC client application name specified by the PC on the ODBC connection string.
3. For SELECT statements involving multiple files, the requested function value is passed as JOIN and the user exit program is called once for each table involved in the join select.
4. The library name does not appear in the SQL statement when \*LIBL is implied. The member name never appears in the SQL statement (members are handled through file overrides).
5. Blanks are passed for this field for any SQL statement other than SELECT, INSERT, UPDATE or DELETE.

## Distributed Relational Database Architecture (DRDA)

DRDA is the IBM AS/400 implementation for database operations on remote AS/400s. The correct operation of Warehouse Builder requires that connections to remote AS/400 databases be set up correctly. If there are errors with the configuration or if the IBM DDM server is not running, Warehouse Builder sets will not extract and/or write data from one AS/400 database to another. The following steps will help ensure that remote databases are configured properly.

1. Type the command WRKRDBDIRE and press Enter.
2. There should be a list of remote database names that correspond to other AS/400 systems. If the required systems are present, the following steps may be skipped.
3. The first step in adding a remote database entry is to decide which communications protocol to use. You have the choice of SNA or TCP/IP.
4. When configuring the entry for SNA, the APPC information for the remote system is required. Consult the IBM APPC manual to determine and configure the name, the device, the local location, the remote network identifier, and the mode. When configuring the entry for TCP/IP, the TCP/IP address is required. Determine the TCP/IP address of the remote AS/400.
5. Use the command ADDRDBDIRE and fill in the required parameters depending on the communications protocol chosen. Usually, the relational database being added is the serial number of the remote AS/400.

If the communications protocol for the remote database entry is TCP/IP, the DRDA server must be started on the remote AS/400 system. Use the command `STRTCPSVR SERVER(*DDM)` on the remote AS/400 to ensure that the server is running.

If the communications protocol for the remote database entry is SNA, ensure that the device being used is in the varied-on state and is active. Consult the IBM APPC manual for instructions if necessary.



---

## Chapter 5. OLAP Maintenance

This chapter discusses typical maintenance procedures that system administrators must perform on the OLAP server. It also includes special considerations when running the Essbase server on an AS/400 system and other miscellaneous notes.

---

### Understanding OLAP Server Configuration

The OLAP server provides two mechanisms for configuring information about the server on the AS/400. Users familiar with Essbase servers on other platforms have traditionally used the Essbase configuration file to configure virtually all Essbase server default characteristics. The OLAP server uses the Essbase configuration file in combination with Essbase environment settings to completely control the OLAP server environment. The following sections explain the AS/400 objects that DB2 OLAP Server for AS/400 installs, the OLAP environment, how to use the AS/400 configuration file, and how the configuration file relates to the Essbase environment settings.

#### Using OLAP Configuration File Settings

The OLAP configuration file contains keywords and values that define characteristics about the OLAP server. The *Essbase Technical Reference*, available online, provides specific information about each of the keywords and values that can be used within the configuration file. The OLAP configuration file can contain the same keywords and values used by Essbase on other platforms. Keywords and values not supported by the OLAP server are ignored.

The largest difference between the OLAP configuration file and the configuration files used on other platforms is file location. The OLAP configuration file is kept in the AS/400 QSYS.LIB file system within the OLAP installation library. The name of the configuration file is CFG. The settings that are used when the server is started and while the server is active are the settings specified within the ESSBASE member. Most Essbase documentation refers to the configuration file as ESSBASE.CFG. Where ESSBASE.CFG is referenced throughout the documentation, substitute the OLAP equivalent, Essbase-lib/CFG member ESSBASE (where *Essbase-lib* is the OLAP installation library).

When the OLAP server component is installed, the configuration file is created empty as a source physical file unless it already exists. If the file exists, it is converted to a source physical file, maintaining the settings it contains.

If the file is deleted, use the following command to recreate the file:

```
CRTSRCPF FILE(Essbase-lib/CFG) RCDLEN(240) MBR(ESSBASE) ACCPTHsiz(*MAX4GB) SIZE(*NOMAX) C
```

Where *Essbase-lib* is the OLAP installation library

For more information on creating AS/400 physical or source physical files, see “Using QSYS.LIB Files from a 5250 Display Session” on page 54. For more information on viewing or editing these files, see “Displaying QSYS.LIB Files from a 5250 Display Session” on page 54 and “Editing QSYS.LIB Files from a 5250 Display Session” on page 55.

## Using OLAP Environment Settings

Each OLAP installation library has its own settings. You can tailor the OLAP environment using a number of parameters. Some parameters are configured during installation while others are configured only when you need them. The environment parameters are implemented through the use of OS/400 environment variables. Each installation of the OLAP server has its own set of environment variables.

Two Essbase commands allow you to work with OLAP environment variables:

- **CHGESSENV** command allows you to change the standard OLAP environment variables and additional generic Essbase environment variables.
- **DSPESENV** command displays the currently configured Essbase environment.

See the following sections for details about configuring, displaying, and removing OLAP environment parameters.

### Understanding When the Environment is Used

It is important to remember that though the Essbase environment can be changed while Essbase operations are active, the changes are not active until all Essbase operations are ended and restarted.

### Configuring OLAP Environment Parameters

A standard Essbase environment is configured at installation time, but it can be changed after installation using the CHGESSENV command. The following table lists the standard environment parameters and their associated command parameters.

*Table 7. Standard Essbase Environment Parameters*

Command parameter	Environment variable
LIB	ARBORLIB
PATH	ARBORPATH

*Table 7. Standard Essbase Environment Parameters (continued)*

SERIALNUM	ESSBASE_SERIAL
IPADDR	ESSBASE_IP_ADDR
THREADS	SERVER_THREADS
EBCCSID	ESSBASE_EBCDIC_CCSID
ASCCSID	ESSBASE_ASCII_CCSID
SCRSYNTAX	N/A
ALLOBJ	Run with all object authority

The following generic variables can also configure OLAP operations.

Table 8. Generic Essbase Environment Variables

Environment variable	Description
ESSBASE_UPPERCASE	When this variable is defined with a non-NULL value, all text output Essbase generates in messages and log files is displayed in uppercase English. This environment variable is most useful when Essbase is used on systems outside the United States that do not support lowercase English characters.
ESSBASE_LINES	Used to specify the number of lines to be displayed by the help command in the ESSCMD interface.
ESSBASE_JOURNAL	Used to specify the library and name of a journal where Essbase will log additional debug information. The information is used by ShowCase development if necessary when investigating problems. <b>Note:</b> This variable should only be used at the direction of the ShowCase technical support staff.

### Displaying OLAP Environment Parameters

To display the currently configured environment, run the DSPESSENV command. This command displays all of the defined environment variables and their values.

### Removing OLAP Environment Parameters

The standard Essbase environment parameters cannot be removed. Once set, they can only be changed to another value. The generic variables can be removed by specifying \*REMOVE as the value on the CHGESSENV command.

## Understanding the Configuration and Environment Relationship

Since DB2 OLAP Server for AS/400 configures operating characteristics in two ways, it is important to understand how the characteristics relate to one another. The Essbase configuration file allows characteristics common among all Essbase platforms to be defined. The Essbase environment settings define either characteristics that are unique to DB2 OLAP Server for AS/400, characteristics that have different value ranges, or defaults for OLAP. Server threads can be configured in both places. If a value is configured in both places, the environment setting takes precedence over the configuration file setting. To reduce user confusion, it is recommended that server threads be configured only using the environment settings.

---

## Using OLAP National Language Support (NLS)

The OLAP server interacts with PC client workstations using an ASCII CCSID to represent character data. To represent character data with DB2/400 and OS/400, DB2 OLAP Server for AS/400 uses EBCDIC CCSID. Defaults for both CCSID values are calculated at installation time based on your AS/400 system settings. For additional information, including the supported EBCDIC and



ASCII CCSIDs, see “Displaying the Current EBCDIC CCSID” and “Displaying the Current ASCII CCSID” on page 50. To view and/or change the EBCDIC and ASCII CCSIDs for the OLAP server, see the following sections.

## Migrating to the Euro CCSID

If you need support for the euro, you will need to change the EBCDIC CCSID. See the table below for easy conversions. If the conversion you need is not in the table, see “Displaying the Current EBCDIC CCSID”.

Table 9. Euro CCSID's

Country, Region or Language group	EBCDIC CCSID	Euro CCSID
United States/Canada/Brazil	37	1140
Germany/Austria	273	1141
Denmark/Norway	277	1142
Finland/Sweden	278	1143
Italy	280	1144
Spain/Latin America	284	1145
United Kingdom	285	1146
France	297	1147
Multi-national #5	500	1148
Iceland	871	1149

To make the CCSID change:

1. Run the AS/400 command *library*/CHGESSENV EBCCSID(*euro CCSID*)

Where:

*library* is the name of the OLAP installation library and *euro CCSID* is the euro CCSID

2. Stop and restart the server by following the instructions in “OLAP Server Component” on page 1.

## Displaying the Current EBCDIC CCSID

1. To display the current EBCDIC CCSID value, run the AS/400 command *library*/DSPESENV INFO (\*ALL)

Where:

*library* is the name of the OLAP installation library

2. Locate the ESSBASE\_EBCDIC\_CCSID variable. The value displayed is the current EBCDIC CCSID.
3. To display 2nd level help text, set the cursor on the Server Attributes message and press **F1**.

## Displaying the Current ASCII CCSID

1. To display the current ASCII CCSID value, run the AS/400 command `library/DSPESENV INFO (*ALL)`

Where:

*library* is the name of the OLAP installation library.

2. Locate the `ESSBASE_ASCII_CCSID` variable. The value displayed is the current ASCII CCSID.
3. To display 2nd level help text, set the cursor on the OLAP Server Attributes message and press **F1**.

## Changing the ASCII and EBCDIC CCSIDs

If you need to change OLAP server CCSID values, be aware of the potential impact on your OLAP environment. You may be required to recreate some of your OLAP environment because certain characters may be interpreted differently once the change is complete. In most environments, an EBCDIC CCSID change will not impact the interpretation of character data stored within the OLAP server. If the ASCII CCSID is changed and only invariant (universal codes amongst CCSIDs) alphabetic characters are used within DB2 OLAP Server for AS/400, the change will have little or no impact. However, if the ASCII CCSID is changed and language-specific characters are used within OLAP, you will be required to recreate some information.

Before attempting a change, examine the differences between the CCSIDs that you are considering changing and the importance of the differences. Consult the IBM manual *AS/400 National Language Support* to assist in your research. If you still believe a CCSID change is necessary, follow the steps below:

1. Save outlines, rules files, calculation scripts, and report scripts to the client PC.
2. As a precautionary measure, export all your crucial data.
3. Make your CCSID change. Run the AS/400 command `library/CHGESSENV ASCCCSSID(ccsid1) EBCCCSID(ccsid2)`  
Where: *library* is the name of the OLAP installation library *ccsid1* is the new ASCII CCSID *ccsid2* is the new EBCDIC CCSID
4. Validate that the change accomplishes your goal.
5. Reload all outlines, rules files, calculation scripts, and report scripts from the client PC to the OLAP server.
6. Import any OLAP data that was interpreted differently as a result of the CCSID change.

---

## Understanding OLAP File System Usage

The IFS is a grouping of several file systems that include the traditionally supported AS/400 file systems and several new file systems. While most of the OS/400 operating system still uses the traditional file systems, the OS/400 IFS can support file systems that use input/output and file structures similar to those used by UNIX and personal computer operating systems. In addition, the file systems that the AS/400 has traditionally used can be accessed using the new interfaces that the IFS support provides.

The IFS lets you store files in a hierarchical directory structure. You can organize data in tree-like structures.

The IFS also provides support for storing information in stream files. A stream file is very similar to a file stored on a personal computer or UNIX-based system. It consists of a stream or a group of bytes that can represent text or binary data. OS/400 limits the size of a stream file to 2 gigabytes.

The last feature that IFS provides is a set of common interfaces that allows users and applications to access IFS objects consistently. It provides interfaces to access stream files as well as database files, documents, and all objects stored on the AS/400.

Essbase was designed to use the simple file systems found on operating systems such as Microsoft Windows NT and UNIX. When Essbase was enabled for the AS/400 by ShowCase, it was designed to use two of the file systems supported by OS/400 IFS: the root file system and the QSYS file system.

Because Essbase was originally designed and built around a simple file system model, most of the OLAP implementation uses the IFS root file system. The root file system uses a hierarchical directory structure and stream-based files. As the file system name implies, the directories are created from the root level of the IFS.

While the root file system is good for most OLAP tasks, some operations are better implemented using the QSYS file system. The QSYS file system is the traditional OS/400 file system where most OS/400 objects are found, including database files, programs, message queues, etc. It is also the file system where most traditional AS/400 applications run, and it has the largest set of tools available to users managing the system.

The following sections explain in detail the syntax of the OLAP file systems. They also provide general information about how to find, view, and edit files within each file system.

## OLAP Installation Library and Path

At installation time, you are asked to specify an OLAP installation library name. By default, the name is ESSBASE. The OLAP installation program uses the installation library name to create both a QSYS.LIB library and a root file system directory. Thus, the default root file system directory name is also ESSBASE.

For instance, if you specified MYBASE as the installation library name, the installation program creates a library named MYBASE in the QSYS.LIB file system and a directory named MYBASE within the root file system.

To differentiate between the areas the OLAP Server uses to store information, we will refer to the library as the *Essbase library* and the root file system directory as the *Essbase directory*.

## OLAP QSYS.LIB Objects Created During Installation

The OLAP server uses the root file system for virtually all user data associated with Essbase application databases. The data in the Essbase library consists of configuration and program information.

The following sections explain which objects are stored in the QSYS.LIB file system. If you are familiar with an Essbase file not mentioned here, it can probably be found in the Essbase directory structure within the root file system.

### Files

On platforms other than the AS/400, Essbase uses text files for setting configuration values. The OLAP Server stores the configuration file in the Essbase library. The application log files are stored in IFS.

The configuration file is a source physical file named CFG and contains a member called ESSBASE. In most of the Essbase documentation, the file is referred to as ESSBASE.CFG. For more information about the Essbase configuration file used with OLAP, see “Using OLAP Configuration File Settings” on page 45.

In 2.5 and earlier, the server and application logs are written to a physical file named LOG. Beginning with release 3.0, these logs are written in ASCII format to IFS. To view the logs from an AS/400 command line, use the DSPESLOG command. DSPESLOG can also be used to copy the logs to the qsys.lib file system.

### Miscellaneous Objects

The OLAP server also includes several miscellaneous objects that implement functions specific to the AS/400 platform. The miscellaneous objects include user spaces to store environment information, commands to provide user

interfaces, a message queue for additional logging support, and a message file that logs AS/400-specific error messages.

### Specifying IFS Syntax for QSYS.LIB Objects

All AS/400 libraries exist in the QSYS.LIB file system. IFS enforces a syntax with each file system to impose a hierarchical format on the file system. For QSYS.LIB objects, this becomes a hierarchy of object names. Each object name consists of the actual object name with an extension that indicates the object type. Therefore, for example, the IFS syntax for specifying the member MYMEMBER in MYLIBRARY/MYFILE is:

```
/qsys.lib/MYLIBRARY.lib/MYFILE.file/MYMEMBER.mbr
```

To look at the list of objects in the Essbase installation library, use the following command:

```
WRKLNK '/QSYS.LIB/Essbase-lib.LIB/*'
```

Where *Essbase-lib* is the Essbase installation library

You can also look at the contents of the same library using the command:

```
DSPLIB Essbase-lib
```

Where *Essbase-lib* is the Essbase installation library

### Specifying IFS Syntax for Root File System Objects

The root file system provides a syntax for specifying objects that is similar to the directories and files on personal computers or UNIX operating systems. In addition, you can have symbolic links to other objects on the system. To look at the objects within the Essbase directory, use the following command:

```
WRKLNK '/Essbase-lib/*'
```

Where *Essbase-lib* is the Essbase installation library

### Finding Files Within the Integrated File System (IFS)

As noted in previous sections, all AS/400 file systems are included in the IFS. OS/400 provides the WRKLNK utility to view the file structures and their attributes independent of the specific file system. If an OLAP administrator needs to find the location of a particular Essbase object, the WRKLNK command is very useful.

When you issue the WRKLNK command, a list of directories and objects for the current directory is shown. The utility also includes options to look at authorities, attributes, next level directories, and several others.

When you are using OLAP and want to display Essbase root file system directories and objects, you will normally use the command:

```
WRKLNK '/Essbase-lib/*'
```

Where *Essbase-lib* is the Essbase installation library and the \* indicates a wild card notation for all files and directories within *Essbase-lib*

If you specify just the '/' on the WRKLNK command, all root directories in the root file system are displayed, including the Essbase directory. In the IFS, all other file systems that the AS/400 supports have a root directory within the root file system as well. Once you are within a root directory for a particular file system, the rules that apply to the file system associated with the directory are enforced. The special directories that designate file system root directories typically start with the letter Q. However, this does not mean that every root directory which starts with the letter Q is a new file system. In order to fully use the WRKLNK command and to understand the file locations, you must first understand the syntax of each file system. OLAP uses the QSYS.LIB file system and the root file systems; they are the only two that we will address. For more information about IFS, see the *IBM OS/400 Integrated File System Introduction* manual.

## Using QSYS.LIB Files from a 5250 Display Session

OLAP uses files within the QSYS.LIB file system to store information that an administrator or user may want to view or edit using a 5250 display session. The following sections provide AS/400 information that is useful for displaying or editing the QSYS.LIB files from a 5250 display session.

### Displaying QSYS.LIB Files from a 5250 Display Session

There are several ways to display QSYS.LIB files from a 5250 display session. The simplest is to use the DSPPFM command, which presents the file contents in a raw format. It also provides a simple string search mechanism and locks the file with a shared read lock. To display the contents of the ESSBASE.CFG file, use the following command:

```
DSPPFM Essbase-lib/CFG ESSBASE
```

Where *Essbase-lib* is the Essbase installation library

When you use the DSPPFM command to display source physical files, you will notice that the data within the file is concatenated with sequence information. This can make reading the file somewhat difficult at times. If you have Query/400 installed, another way to display QSYS.LIB files is using the RUNQRY command. To display the ESSBASE.CFG file using RUNQRY, use the following command:

- RUNQRY () ((Essbase-lib/CFG ESSBASE))

Where *Essbase-lib* is the Essbase installation library

- The RUNQRY interface displays information separated into fields, which makes it easier to read. The disadvantage of using RUNQRY is that it does not have a search mechanism.

- The DSPPFM and RUNQRY commands are AS/400 commands. For more information, see the IBM AS/400 command reference manuals.

### **Editing QSYS.LIB Files from a 5250 Display Session**

There are several ways to edit QSYS.LIB files from a 5250 display session. Each method differs depending on the type of QSYS.LIB file and by the products you have installed on the AS/400. For source physical files, the simplest method is to use the Source Edit Utility (SEU). The Source Edit Utility is not part of the base operating system but is a common licensed program many customers have. It is installed as part of the Application Development Toolset/400 licensed program and can be purchased separately.

**Using the SEU to Edit the ESSBASE.CFG File:** Issue the STRSEU command as shown here: STRSEU Essbase-lib/CFG ESSBASE

Where *Essbase-lib* is the Essbase installation library

SEU provides a full screen editor interface that provides a good set of tools for editing text files. For more information about SEU, see the *IBM Source Edit Utility* manual provided with the Application Development ToolSet/400 licensed program.

**Using a Text Editor to Edit the ESSBASE.CFG File:** If the SEU is not available, the easiest way to edit the QSYS.LIB files is to open a DOS text editor on your PC (e.g., Notepad), add the necessary entries, and save the entries in a text file.

1. In a text editor, type the appropriate entries as in the following example:  
NETDELAY 2000 NETRETRYCOUNT 2000
2. Name the file CFG and save it in the root of your C: drive. By default, a text editor will add the .txt file extension to the filename.
3. At a DOS prompt, type the following commands: FTP 400ipaddress  
profile password PUT C:\CFG.TXT ESSBASE/CFG.ESSBASE QUIT  
If there are already entries in the existing ESSBASE.CFG file, you must add these entries to the new entries in the CFG.txt file before you FTP the file to the AS/400.

---

## **Using ESSCMD/400**

ESSCMD/400 is an OLAP component that lets AS/400-based users administer the OLAP server from a command line. The functionality that ESSCMD/400 provides is consistent with the operations that ESSCMD users of the PC versions of ESSCMD (ESSCMD and ESSCMDW) are familiar with.

ESSCMD/400 uses the AS/400 standard input, output, and error devices to retrieve input, write output, and write errors, respectively. Any one of these

can be switched from using the device interface to using an AS/400 QSYS.LIB physical or source physical file. This is accomplished with the parameters provided on the ESSCMD/400 command. Such flexibility lets an administrator build an input file that is a script of Essbase commands run in sequence. From this point forward, we will refer to running ESSCMD/400 with an input file as the “scripted interface” and running ESSCMD/400 without an input file as the “non-scripted” or “command line interface” to ESSCMD/400.

The default interface that ESSCMD/400 uses is the command line interface. In addition to the different interfaces that can drive ESSCMD/400, you can run it in either interactive or batch mode. The combination of different ESSCMD modes and interfaces make ESSCMD/400 a powerful administrative tool.

An alternative to ESSCMD/400 is the **ESSCMD API Toolkit**, which provides OS/400 command interfaces and OS/400 error handling for all ESSCMD/400 operations. For more information, see “Using ESSCMD API Toolkit” on page 62.

## Using ESSCMD/400 in Interactive Mode

To use ESSCMD/400 from an interactive display session, you can use either the command line or script interfaces. If the command line interface is used, you should not specify an output or error file when running ESSCMD/400. This allows all input, output, and error information to be read from and written to the screen on the workstation running the command. If the user specifies an output or error file when running ESSCMD/400, it will read input from the workstation that issued the command, but will direct the output or error information to a file rather than the display. This type of usage could confuse users since they will be prompted for input but will not know what to enter or if an error occurs because of their input. In summary, it always makes sense when using the ESSCMD/400 command line interface to omit specifying an output or error file.

To run ESSCMD/400 interactively using the command line interface, issue the STRESSCMD command from a 5250 command line without specifying any parameters.

If you choose to use the scripted interface, you need to specify an output or error file depending on the intent of the script. If you have a short running script or a script that is being tested, you could run the script interactively while directing the output and error information to the workstation. This ensures that the script is properly coded and displays on the screen what the script did or did not do. While running scripts interactively and displaying output and error information to the screen is useful for testing scripts, script information is most often directed to a file. This allows you to start scripts, let them run unattended, and easily verify at a later time that they finished correctly. When an output or error file is specified, the output or error



information that is normally displayed on the screen is written to the specified file. For information about creating and viewing ESSCMD/400 input, output, and error files, see “Using QSYS.LIB Files from a 5250 Display Session” on page 54.

The syntax for running ESSCMD/400 with an input, output, and error file is:

```
Essbase-lib/STRESSCMD INFILE(inlibrary/infile) INMBR(inmember)  
OUTFILE(outlibrary/outfile) OUTMBR(outmember) ERRFILE(errlibrary/errfile)  
ERRMBR(errmember)
```

Where: *Essbase-lib* is the library where OLAP is installed *inlibrary* is the library where the input script file is located *infile* is the input script file that contains the input script file member *inmember* is the input script file member that contains valid OLAP commands *outlibrary* is the library where the output file is located *outfile* is the file that contains the output file member *outmember* is the output member where ESSCMD/400 screen output is written *errlibrary* is the library where the error file is located *errfile* is the file that contains the error file member *errmember* is the error file member where ESSCMD/400 error information is written.

## Using ESSCMD/400 in Batch Mode

To use ESSCMD/400 in batch mode, you can only use the script interface, since there is no interaction with a display device in batch mode. This means when the ESSCMD/400 command is submitted to AS/400 batch, you must specify an input file. If an input file is not specified and an ESSCMD request is submitted as an AS/400 batch job, unpredictable results will occur.

In batch mode, you have the option of specifying an output or error file when running ESSCMD/400. If you do not specify an output or error file on the ESSCMD/400 command, the output and error information will be written to AS/400 spool files. If you specify an output or error file on the ESSCMD/400 command, the output or error information is directed to a file rather than the spool file. It also causes the output and error information to be written to a predefined, easy-to-find location so the user does not have to completely understand the AS/400 work management. Though using output and error files can be very useful, each method has benefits that can be useful.

While there are several ways to schedule AS/400 work and to run AS/400 jobs automatically, the simplest method is to use the AS/400 SBMJOB command.

Following is an example that submits an ESSCMD/400 request to batch using the SBMJOB command.

```
SBMJOB CMD(Essbase-lib/STRESSCMD INFILE(inlibrary/infile) INMBR(inmember)  
OUTFILE(outlibrary/outfile) OUTMBR(outmember) ERRFILE(errlibrary/errfile)  
ERRMBR(errmember))
```

Where: *Essbase-lib* is the library where OLAP is installed *inlibrary* is the library where the input script file is located *infile* is the input script file that contains the input script file member *inmember* is the input script file member that contains valid OLAP commands *outlibrary* is the library name where the output file is located *outfile* is the file that contains the output file member *outmember* is the output member where ESSCMD/400 screen output is written *errlibrary* is the library where the error file is located *errfile* is the file that contains the error file member *errmember* is the error file member where ESSCMD/400 error information is written

## Using an ESSCMD/400 Input Script File

In order to create ESSCMD/400 script files, a user must create an AS/400 QSYS.LIB file and member to contain the Essbase commands. For specific information about AS/400 file systems used by OLAP and their differences, see “Understanding OLAP File System Usage” on page 51. The file can be created as a physical or source physical file. It is recommended that the record size be between 80 and 240 characters to allow easy AS/400-based editing and viewing. For more information about creating AS/400 physical or source physical files, see “OLAP QSYS.LIB Objects Created During Installation” on page 52.

Once the file and member are created, the file must be updated to contain the Essbase commands that ESSCMD/400 will run. For more information about viewing and editing AS/400 physical or source physical files, see “Displaying QSYS.LIB Files from a 5250 Display Session” on page 54 and “Editing QSYS.LIB Files from a 5250 Display Session” on page 55.

## ESSCMD/400 Script File Differences

The Essbase commands that are supported by ESSCMD/400 are almost identical to the commands supported by ESSCMD interfaces on the PC. Although the interface that ESSCMD/400 provides is very similar to the interfaces provided by ESSCMD utilities on other platforms; there are a few subtle differences in coding an ESSCMD/400 script file.

### Specifying Output and Error Files within ESSCMD/400 Script Files

When you are creating ESSCMD/400 scripts, several ESSCMD commands allow you to specify input, output, and error files to be used by the given command. If a command you are running requires a file name as a parameter, you must enter the file name in the syntax that Essbase understands.

For example, when using the BUILDDIM or IMPORT commands, you have the option of specifying a file name to be used for recording errors found

during the operation. You must enter an IFS path name. You have the option of storing the file in the root file system or the QSYS.LIB file system.

The following path example stores the information in a library named MYLIB, file named MYFILE, and member named MYMEMBER within the QSYS file system. For more information about specifying QSYS.LIB file syntax, see “Specifying IFS Syntax for QSYS.LIB Objects” on page 53.

*/qsys.lib/MYLIB.lib/MYFILE.file/MYMEMBER.mbr*

However, if you specify a file in the QSYS.LIB file system as shown in the example, the library and file must exist prior to running ESSCMD/400. ESSCMD/400 does not create the library and file. ESSCMD/400 will create the specified member if it does not exist. However, if the member does not exist, you should verify that the MAXMBRS setting on the file will allow the creation of a new member. If the MAXMBRS limit is reached, the member cannot be created. You should also verify that the file is created with a record size large enough to accommodate the error messages and any failing input records that may be logged to the file.

The following path example stores the information in the DIMBUILD.ERR file within the Essbase directory for the Sample Basic application database of the root file system. For more information about specifying root file system file syntax, see “Specifying IFS Syntax for QSYS.LIB Objects” on page 53.

*/ESSBASE/app/Sample/Basic/dimbuild.err*

When a root file system path is specified, the directories must exist prior to running ESSCMD/400. The file will automatically be created if it does not exist. If the file exists, it is overwritten.

### **Considerations When Specifying Dimension Build and Data Load Error Files**

ESSCMD/400 allows you to specify an error file that receives rejected records if a dimension build or data load error occurs. This file can reside in either the root file system or the QSYS file system. For information on specifying an error file, see “Specifying Output and Error Files within ESSCMD/400 Script Files” on page 58. If you want to use the error file to reload rejected records, direct it to a stream file in the root file system. If you only want to display the error file, store the file in the QSYS file system. In this situation, tabs in the file are expanded to spaces, making the file easier to read.

If you direct the error file to the root file system, you can use the CPYFRMSTMF command to copy the error file to the QSYS file system to read.

Prior to creating or viewing a QSYS file that holds error records, you must determine the EBCDIC server CCSID that OLAP is configured to use. To do so:

1. Find the server CCSID. The server CCSID was configured during installation or using the CHGESSENV interface when you set up the server. For more information, see the *Using DB2 OLAP Server* manual. If you do not know the server CCSID OLAP is configured to use, run the DSPESSENV command to determine the value of the ESSBASE\_EBCDIC\_CCSID variable. This is the server CCSID.
2. If the server CCSID is a mixed CCSID (i.e., contains single-byte and double-byte characters), refer to the following table to determine the associated single-byte CCSID. For more information, see the IBM manual *AS/400 National Language Support*.
3. To display 2nd level help text, set the cursor on the OLAP Server Attributes message and press F1.

Table 10. Mixed CCSID Associations

Mixed CCSID	Associated Single-Byte CCSID
05035	01027
00939	01027
00937	00037
00935	00836
00933	00833

If you choose to log errors directly to the QSYS file system, the file must exist prior to running ESSCMD/400. To create the source file in QSYS, use the command:

```
CRTSRCPF FILE(mylib/pf) RCDLEN(length) CCSID(value)
```

Where: *mylib* is the QSYS library name *pf* is the QSYS file name *length* is the record length. (Be certain to specify a record length that is long enough to handle the longest rejected record, keeping in mind that tab characters will be expanded to spaces at a ratio of eight spaces per tab character) *value* is the single byte CCSID that the OLAP server uses. (On servers using double byte character sets, *value* must be the single byte CCSID that is associated with the mixed CCSID of the OLAP server)

If you wrote the error file to the root file system and want to view the error file, copy the file from the root file system to the QSYS file system, then use Query, RUNQRY, or DSPPFM OS/400 commands to view the file. To copy the file into the QSYS file system:

1. Create a source physical file using the CRTSRCPF command described above.
2. Copy the file from the root file system to the QSYS file system using the command:

```
CPYFRMSTMF FROMSTMF('file name')
TOMBR('/qsys.lib/mylib.lib/pf.file/member.mbr') TABEXPN(*NO)
```

Where: *file name* is the full path and name of the file in the root file system where the error records were originally logged

(e.g., /essbase/app/smstbc/dbcs/dimerr.txt) *mylib* is the QSYS library name *pf* is the QSYS file name *member* is the name of the member to create in the QSYS file system (e.g. dimerr.mbr)

3. Use Query or the OS/400 RUNQRY command to look at the file. For example:

```
RUNQRY QRYFILE((mylib/pf member))
```

Where: *mylib* is the QSYS library name *pf* is the QSYS file name *member* is the QSYS member name

To view files with a record length less than 240, you can also use the Display Physical File Member command (DSPPFM). For example:

```
DSPPFM mylib/pf member
```

Where: *mylib* is the library name *pf* is the file name *member* is the member name

### Output Command

The OUTPUT command directs process information output from the ESSCMD/400 session to a text file.

Output *lib/file(member)*

Where *lib/file(member)* is the library and file name of the text file where the output is directed

## Using an ESSCMD/400 Output or Error File Specified from Within ESSCMD

As explained previously, output and error files can be created to capture ESSCMD/400 output and error information that would otherwise be sent to the screen. It is recommended that these files be created prior to running ESSCMD/400. The files can be created as physical or source physical files. It is recommended that the record size be between 80 and 240 characters to allow easy AS/400-based editing and viewing. If the ESSCMD/400 output file is not created in advance, the file is created as a source physical file with a data field of 254 characters.

The size of the data field makes viewing or editing the output difficult. If the record size is not large enough to hold an entire ESSCMD/400 output record, the output record is truncated to fit within the file. For more information about creating AS/400 physical or source physical files, see “OLAP QSYS.LIB Objects Created During Installation” on page 52. For more information about viewing or editing these files, see “Displaying QSYS.LIB Files from a 5250 Display Session” on page 54 and “Editing QSYS.LIB Files from a 5250 Display Session” on page 55.

## Getting Help ESSCMD Commands

You can install Hyperion documentation with DB2 OLAP Server for AS/400. For help about ESSCMD/400 commands, see the Hyperion Essbase HTML Documentation Help guide. To view this guide on your PC, select **Essbase Books** from the STRATEGY drop-down menu.

---

## Using ESSCMD API Toolkit

### Introduction

The ESSCMD API Toolkit lets an OLAP administrator automate Essbase. ESSCMD API Toolkit provides OS/400 command interfaces and OS/400 error handling for all of the ESSCMD/400 operations. The ESSCMD API Toolkit is automatically enabled and sits on top of the ESSCMD/400 function, transforming the script based interface of ESSCMD/400 into an OS/400 command interface.

### Where Can I Learn More?

The Quick Technical Reference includes information that allows you to take advantage of the ESSCMD API Toolkit. It is available online under Essbase Books on the STRATEGY menu.

For details about individual ESSCMD API commands, see the AS/400 online help. To use the help, enter an ESSCMD API command, press **F4**, then press **F1** for help about a parameter.

### Overview

The ESSCMD API Toolkit lets you open and maintain an ESSCMD/400 session across multiple OS/400 commands. This lets you easily intersperse ESSCMD/400 commands with other OS/400 commands. You can also code your CL program to respond to errors reported by Essbase on a command by command basis, using standard OS/400 error and message handling mechanisms.

### Command Structure

The ESSCMD API Toolkit consists of four base OS/400 commands, along with numerous supplemental commands that are discussed later. OS/400 command

level help information is available for each of the base and supplemental commands. The four base commands are:

- LOGINESS
- RUNESSCMD
- LOGOUTESS
- RTVESSSTS

The STRESSCMD command is not an ESSCMD API command. STRESSCMD acts independently from the ESSCMD API commands.

### **LOGINESS**

Issue LOGINESS before running other ESSCMD API commands. After issuing the LOGINESS command, RUNESSCMD (or a supplemental command) interacts with the OLAP server under the session opened by the LOGINESS command. Multiple logins with LOGINESS are not allowed.

### **RUNESSCMD**

RUNESSCMD lets you run any ESSCMD command directly from the OS/400 command line, or from within a CL or other program. With RUNESSCMD, enter the underlying ESSCMD/400 command as follows:

```
RUNESSCMD COMMAND('UNLOADAPP MYAPP')
```

Typically, you will use the LOGINESS command (described earlier) to initiate the session prior to using RUNESSCMD.

### **LOGOUTESS**

LOGOUTESS ends the connection and closes the ESSCMD API session by issuing the ESSCMD EXIT command within the regular ESSCMD interface.

### **RTVESSSTS**

RTVESSSTS can be used from within a CL program to retrieve information about a previous ESSCMD API Toolkit command if the session associated with that command has not been closed.

There are three pieces of information that can be retrieved:

- Essbase message number
- Essbase message severity
- Essbase message text

See the online command help for more information on the data types of these variables and the possible return values.

## **Auto-logoff Considerations**

When you use ESSCMD API commands within another program, you must consider how the Auto-logoff feature of Essbase will affect the program.

Essbase is installed with a default auto-logoff time of 60 minutes, but this may have been changed to a smaller amount. This might be a problem in a program such as the following:

- LOGIN SVRUSR(*userid*) SVRPW(*password*)
- ENDESSAPP APPNAME(*mycube*)
- SAVLIB DEV(TAP01) ...
- STRESSAPP APPNAME(*mycube*)

The SAVLIB operation may take a long time, particularly when loading or changing a tape is required to complete the operation. If the time between the completion of the ENDESSAPP and start of the STRESSAPP command is longer than the auto-logoff setting, the STRESSAPP will fail.

Unfortunately, the auto-logoff setting applies to all users, so you can't adjust it for the user ID used in the program. If you prefer not to change the time-out for all Essbase users (it can be set to no auto logoff), run LOGOUTESS and LOGINESS again wherever a problem might occur. For instance:

- LOGIN SVRUSR(*userid*) SVRPW(*password*)
- ENDESSAPP APPNAME(*mycube*)
- LOGOUTESS
- SAVLIB DEV(TAP01) ...
- LOGINESS *userid password*
- STRESSAPP APPNAME(*mycube*)

## Supplemental Commands

In addition to the commands described earlier, ESSCMD API Toolkit provides alternative commands to RUNESSCMD. There's no need to remember the ESSCMD command syntax. The underlying ESSCMD syntax is mapped to OS/400 command keywords.

Each supplemental command corresponds to an ESSCMD/400 command. For instance, the ESSCMD API command STRESSAPP corresponds to the ESSCMD/400 LOADAPP command. These OS/400 commands have the look and feel of standard OS/400 commands, complete with prompted keywords, help, and so on. For those ESSCMD/400 commands that do not have a corresponding ESSCMD API command, you can use RUNESSCMD.

With ESSCMD/400, choose an application and database using the SELECT command so that subsequent commands (that do not specify an application or database) can operate upon the selected database. The ESSCMD API supplemental commands require that you specify the application and database on the command itself. The appropriate SELECT command is issued implicitly for you.



Most of the supplemental commands let you utilize the full power of the underlying ESSCMD/400 commands.

In a few cases, the ESSCMD API Toolkit supplemental command does not permit all of the underlying command's forms to be used. If you need access to the other forms, they are always available via the RUNESSCMDC command.

This list contains all of the Essbase ESSCMD API supplemental commands and indicates which corresponding ESSCMD command is issued. See the ESSCMD documentation in the "Quick Reference Guide" for more information on what each ESSCMD command does.

The IMPESSSQL and BLDESSDIM commands do not permit all forms of their respective ESSCMD commands to be issued. In each case, only the SQL form with a server-based rule file is supported, as this is the most common usage. To handle other cases, use RUNESSCMDC.

*Table 11. ESSCMD API supplemental commands*

ESSCMD/400 API Command	ESSCMD/400 Command
ADDESSGRPU	ADDUSER
BLDESSDIM	BUILD DIM
CLCESSDFT	CALCDEFAULT
CPYESSAPP	COPYAPP
CPYESSDB	COPYDB
CRTESSAPP	CREATEAPP
CRTESSDB	CREATEDB
CRTESSGRP	CREATEGROUP
CRTESSUSR	CREATEUSER
CRTESSVAR	CREATEVARIABLE
DISESSLGN	DISABLELOGIN
DLTESSAPP	DELETEAPP
DLTESSDB	DELETEDB
DLTESSGRP	DELETEDGROUP
DLTESSUSR	DELETEUSER
DLTESSVAR	DELETEVARIABLE
ENBESSLGN	ENABLELOGIN
GETESSAPP	GETAPPSTATE
IMPESSSQL	IMPORT

Table 11. ESSCMD API supplemental commands (continued)

STRESSAPP	LOADAPP
LOGINESS	LOGIN
LOGOUTESS	EXIT
LSTESSAPP	LISTAPP
LSTESSDB	LISTDB
LSTESSGRPS	LISTGROUPS
LSTESSGRPU	LISTGROUPUSERS
LSTESSUSR	LISTUSERS
LSTESSVAR	LISTVARIABLES
CLRESSDB	RESETDB
RNMESSAPP	RENAMEAPP
RNMESSDB	RENAMEDB
RNMESSUSR	RENAMEUSER
RMVESSGRPU	REMOVEUSER
RUNESSCLC	RUNCALC
ENDESSSVR	SHUTDOWNSERVER
ENDESSAPP	UNLOADAPP
UPDESSVAR	UPDATEVARIABLE

## Output Processing

RUNESSCMD and other ESSCMD/400 API commands each have an `OUTFILE` keyword. This keyword follows normal OS/400 `OUTFILE` conventions. If the file does not exist, it will be created with a record length of 240 characters.

## Controlling Error Handling When Programming

The OLAP messaging mechanism defines five message severity levels:

1. Informational
2. Warning
3. Error
4. Serious
5. Fatal

If Essbase returns only *Informational* and/or *Warning* messages, by default an ESSCMD/400 API command will end with a normal OS/400 completion message. Conversely, if Essbase returns an *Error*, *Serious*, or *Fatal* message, the ESSCMD/400 API command issues an OS/400 Escape message.

To override this behavior use the optional ESCLEVEL keyword on most ESSCMD/400 API commands. For example, you could use the following sequence of commands to delete an existing application and then recreate it by copying another application:

```
LOGINESS myuserid mypassword
DLTESSAPP myapp ESCLEVEL(*SERIOUS)
CPYESSAPP yourapp myapp ESCLEVEL(*ALL)
LOGOUTESS
```

The DLTESSAPP uses ESCLEVEL(\*SERIOUS) so that the sometimes expected error message doesn't cause the program to fail:

```
(1051030) ERROR: Applicationmyapp does not exist.
```

In the example above the Essbase ERROR level message will behave the same way, resulting in an OS/400 Completion message. Serious and Fatal messages will still result in an OS/400 Escape message.

In the example above, the CPYESSAPP command uses ESCLEVEL(\*ALL) because the copy should always work after successfully deleting the application. CPYESSAPP doesn't return any Essbase messages, not even an Informational one. By using \*ALL, the program will fail if anything other than what is expected (no messages) occurs. See the online help for more information about the ESCLEVEL keyword.

In addition to the ESCLEVEL keyword, you can use the RTVESSSTS command within your program to retrieve status information about the most recent Essbase message issued. This lets you programmatically retrieve the level of the message (e.g., Informational, Warning), the Essbase message number, and the message text. For example, the following message:

```
(1051030) ERROR: Application myapp does not exist.
```

Would return 1051030 for the STATUS (message number), "ERROR" for the MSGLEVEL and "Applicationmyapp does not exist." for the MESSAGE.

See the online help for RTVESSSTS for more information about the types and sizes of the keywords for RTVESSSTS.

## Debugging

It is possible to see the commands ESSCMD/400 API passes to the ESSCMD/400 engine. To log each ESSCMD/400 command, issue the following command in the job ESSCMD/400 is running under. This command will also log the password.

```
ADDENVVAR ENVVAR(ESSBASE_CMDAPI_LOG) VALUE('your_library/your_file')
```

Once the environment variable is set, turn off logging with the next command:

```
CHGENVVAR ENVVAR(ESSBASE_CMDAPI_LOG) VALUE(*NULL)
```

Logging can be turned on and off again with the CHGENVVAR command.

You can use all the standard OS/400 built-in debugging support for your CL program, RPG program, or other program that uses the ESSCMD/400 API.

To more easily debug CL programs that use Esscmd API, you may want to create your CL program to log CL Program Commands:

```
CRTCLPGM ... LOG(*YES)
```

In addition, change your job to default to not log CL Program Commands by issuing:

```
CHGJOB LOGCLPGM(*NO)
```

The combination of these changes will result in a job log (via DSPJOBLOG) which shows only the commands issued directly by your CL program, and not those issued internally by ShowCase software.

---

## Maintaining OLAP Log Files

While OLAP is running, entries are added to various log files. These files all reside in the IFS root file system in either the BIN or application directories. Over time, these files can grow very large.

To keep the size of these files relatively small, you should clear the server and application logs occasionally. To do so, in the Essbase Application Manager, on the **Server** menu, click **Delete Event Log**, or, on the **Application** menu, click **Delete Event Log**. If the logs are not properly maintained, they may fill up or consume an unnecessary amount of storage space. For more information, see “Understanding OLAP File System Usage” on page 51.

---

## Index Cache Size

The index cache size of an OLAP database depends on the size of the database index page. For every 1 kilobyte (KB) of index page size, you can specify up to 125 megabytes (MB) for the index cache size. The next table shows the maximum sizes allowed for the index cache, depending on the size of the index page:

*Table 12. Maximum Sizes for the Index Cache*

Index Page Size	Maximum Index Cache Size
1 KB (1024 B)	125 MB (131072000 B)

Table 12. Maximum Sizes for the Index Cache (continued)

2 KB (2048 B)	250 MB (262144000 B)
4 KB (4096 B)	500 MB (524288000 B)
8 KB (8192 B)	1000 MB (1048576000 B)

If the index cache size limit is exceeded, the OLAP server will reset the cache size to the maximum size allowed for the index page size. For example, if you specify a cache size of 1000 MB, but your index page size is 2 KB, OLAP will reset the cache size to 250 MB.

---

## Loading DB2/400 Data Into OLAP

OLAP data load and dimension build operations are most efficient when done using the SQL interface support. The SQL interface support is shipped as part of the base OLAP package; you do not need to install a separate SQL interface module.

### Setting Up Your AS/400 to Allow SQL Loads

1. Type WRKRDBDIRE and press **Enter**.
2. If you do not see an entry for \*LOCAL, use option 1 to add a \*LOCAL entry.

### Loading Data Using the SQL Interface Support

1. Using the Essbase Application Manager, go to a Data Prep Editor window and on the **File** menu, click **Open SQL**.
2. Select your application and database in the **Select Server, Application and Database** dialog box and click **OK**.
3. In the **SQL Data Sources** selection box of the **Define SQL** dialog box, you should see the \*LOCAL Relational Database Directory entry highlighted. The **Connect** information should NOT be filled in.

The \*LOCAL entry allows your rules files to be portable between OLAP servers on different systems. You can also select the name of your system from the list, but your rules file must always be used only on that system.

4. The **Select** information allows you to specify the SQL statement to be used to load data into the Data Prep Editor. In the **From** box, enter the names of the tables in the SQL naming convention as opposed to the AS/400 naming convention. In the SQL naming convention, a period (.) is used between a library and file name. For example: library.filename You cannot specify a specific OS/400 database file member name on this interface.
5. If you wish to add GROUP BY or ORDER BY clauses, you may do so in the **Where** box; however, you must first specify at least one WHERE

condition. If you do not have a WHERE condition to specify, simply use 1 = 1 as a placeholder WHERE condition in the **Where** box. For example: 1 = 1 ORDER BY Product

6. Click **OK/Retrieve**.

## Substitution Variables in Rule File SQL Statements

Essbase substitution variables can be used in calc scripts, report scripts, and spreadsheets. With OLAP, you can also use substitution variables within the SQL statement specified in a Data Load or Dimension Build rule file. This allows easy re-use of rule files for multiple data files of the same format or for re-targeting loads for test versus production libraries. The value of substitution variables can be modified as needed, which will in turn modify the behavior of the scripts and rule files that use the variables.

To use substitution variables in the SQL statement, simply specify the variable prefixed with an ampersand (&). If a variable is used in the WHERE box, and the result of the entire WHERE clause after all variables are substituted is all blanks or an empty string, the entire WHERE clause will be omitted from the SQL statement.

If a variable reference does not have a value assigned to it, no substitution will occur and the original variable name along with the ampersand will be passed in as part of the SQL statement.

The following message is an example of the message that will be logged if the variable reference is not assigned a value:

```
Local/Appname/Dbname/Userid/Info(1012042) Substitution variable <xy> doesn't exist.
```

Substitution variables can be managed by selecting the **Substitution Variables** option from the **Server** menu in OLAP Application Manager, or via the ESSCMD commands: CREATEVARIABLE, UPDATEVARIABLE, and DELETEVARIABLE.

See Hyperion Solutions Essbase documentation for more information about substitution variables.

## Flat File Support

Many non-AS/400 systems support character delimited files, also known as flat files. Flat files often use a delimiter character such as a tab or space to separate individual fields within a record.

OLAP can load flat files if they are stored with a .txt extension in the IFS application/database directory. ESSCMD/400 can also read from flat files from IFS.

The recommended way to load data is to use a load rule with a SQL statement. To load data from DB2 files, specify the appropriate SQL statement to select the fields you want to load. You can also load a flat file using this same technique. In this case, the standard OS/400 database file contains a single character field. This field is used to hold character-delimited data.

For example, suppose you have a tab delimited text file on a PC. You can load data directly into OLAP from this file using ESSCMD on the PC. As an alternative, you can copy this file to the AS/400 and load the data using SQL. The following illustrates the steps that would be used:

1. Create a database file on the AS/400 using the appropriate record length:  
`CRTPF FILE(MYLIB/FILE) RCDLEN(132)`
2. Using the FTP PUT command, copy the data from the PC to the AS/400:  
`put c:\pcdata.txt mylib/file.file`
3. The file MYLIB/FILE now contains the tab delimited data. Using Application Manager, create or edit the load rule to read this data by using the following select statement:  
`SELECT * FROM MYLIB.FILE`
4. On the **Options** menu, click **Data File Attributes** and use the **File Delimiter** tab of the **Data File Attributes** dialog box to set the appropriate delimiter.

---

## Using Warehouse Builder with OLAP

Using Warehouse Builder, you can perform several operations against an OLAP database (e.g., clear a database, build dimensions, load a database, calculate a database). When using Warehouse Builder to build dimensions or load data, OLAP uses SQL to read records and load them into the database. Essbase validates each record as it is processed, and OLAP may reject records if it encounters an error.

### Dimension Build and Data Load Rejected Records

The member names identified by each record must match the member names specified in the OLAP outline. If the member names do not match (or some other error occurs), OLAP rejects the record. In this situation, if the OLAP build plan defined in Warehouse Builder is set to "Continue on Error," Warehouse Builder logs the rejected record in the JOBMSGs table and processing continues.

The records causing an error are logged with a MSG\_ID of 'TXT0208' (in the case of a data load) or 'TXT0207' (in the case of a dimension build). The record itself is logged in the MSG\_HELP field. The object ID for the OLAP

build plan is stored in the OBJECTID field. The sequence number of the OLAP build plan identifies the OLAP operations within the OLAP build plan and is stored in SEQ\_NUMBER.

## Reloading Rejected Records

After fixing your OLAP outline, it is possible to reprocess only the rejected records (as opposed to reprocessing all of the data). To reprocess the rejected records, use the Essbase Application Manager to create a dimension build or data load rule that points to the JOBMSGs table. Following is an example SQL statement that reprocesses the rejected records of an OLAP build plan whose object ID is 101 and whose sequence number is 1:

```
SELECT MSG_HELP FROM SCSERVER.JOBMSGs WHERE MSG_ID = 'TXT0208' AND OBJECT_ID = 101 AND S
```

To reprocess the rejected records for a dimension build, the MSG\_ID value is 'TXT0207'.

To determine the correct object ID and sequence number to use, run the queries (\*.DBQ) shipped with Warehouse Builder against the JOBMSGs table. The queries are located in the *REPORTS* folder where Warehouse Builder is installed.

## Naming the OLAP Server

When using Warehouse Builder with OLAP, the value you specify for the OLAP server on the **Build Plan** dialog box must be either an IP address in numeric format or a valid (or useable) name on the control server AS/400. To check whether a name is valid, enter the AS/400 command CFGTCP from an AS/400 session window on the control server AS/400 and choose option 10. In the screen that appears, you can verify that the name associated with the TCP/IP address for the OLAP server is the same as the name defined on your PC.

For additional information on using Warehouse Builder with OLAP, consult the Warehouse Builder online help.

---

## Security

All Essbase security functions are supported by OLAP. However, individual OLAP data files reside in the OS/400 Integrated File System (IFS) and may be accessible via other OS/400 interfaces. You should verify that these files are properly secured with respect to the AS/400 defined security mechanisms. In particular, the Essbase security file itself, named ESSBASE.SEC, is stored within the IFS file system.

### ALLOBJ Authority

With a profile that has \*ALLOBJ authority, a user has full AS/400 access, including all databases, programs, folders, and files. If you want a user profile to have access to certain AS/400 objects, but do not want to assign \*ALLOBJ



authority, you can grant users authority access to certain objects and not to others. For example, a profile at the \*USER level can be granted access to library ABC, but not to library XYZ.

Essbase security controls user access. Using a profile that has been assigned the \*SECOFR class is the easy way to start the OLAP server. However, to enable a user to run OLAP with anything less than \*ALLOBJ authority, follow these steps:

1. Create a user profile with the desired security class to be dedicated to starting Essbase. Make sure that Essbase will always be started with this user profile.
2. Run the GRTESSAUT command to grant the user profile all the necessary authorities to start Essbase.
3. Using the CHGESSENV command, change the "Run with All Object Authority" field to \*NO.
4. To verify if a user profile is authorized to start Essbase, run the CHKESSENV command.

---

## Integrated Security

OLAP provides support for Integrated Security. By using OS/400 security controls, Integrated Security lets you control who has access to OLAP. As a result, your security scheme for OLAP, Analyzer, and Analyzer for the Web is greatly simplified. Whether you're using OLAP alone, or implementing Analyzer OLAP Server, you can take advantage of Integrated Security.

Once you set up the Integrated Security environment between OLAP and OS/400, OS/400 user profile operations will be applied to OLAP security. Then you can perform most routine OLAP user security changes from OS/400 security interfaces, or from Warehouse Manager. You will no longer need to add or remove users or change group memberships via the OLAP security interfaces. However, you will still use OLAP security interfaces to establish which applications and databases a group can access.

If you choose *not* to use the Integrated Security feature, you need not do anything. By default, OLAP runs with its own security system, independent of OS/400 security.

Whether or not Integrated Security is enabled, access to DB2/400 for data loading and dimension building will still require OS/400 authorization information to be entered.

### Setting up OLAP Integrated Security

To set up OLAP Integrated Security, you should be familiar with basic Essbase security and basic OS/400 security, including OS/400's group profile support.

This includes being familiar with how to create users and groups in OLAP using Application Manager and OS/400 using Warehouse Manager.

For detailed information about basic OLAP and OS/400 security, see the *Essbase Database Administrator's Guide* and the *OS/400 Security Reference* (IBM Document No. SC41-4302-00). For more information about Warehouse Manager, see the Warehouse Manager online help. You should also be familiar with editing the OLAP configuration files. For information about editing the configuration file, see “Editing QSYS.LIB Files from a 5250 Display Session” on page 55.

OLAP requires that actual Essbase user IDs be defined for many operations. For example, locks on objects such as outlines, rule files, or calc scripts must be associated with an OLAP user ID. For this reason, the Integrated Security feature of OLAP will automatically create and delete OLAP user IDs in response to OS/400 user profile creations, deletions, and other operations.

OLAP Integrated Security can be configured in one of two ways: you may enable *all* valid OS/400 users to also have access to OLAP, or you may enable only a *subset* of OS/400 users to have access to OLAP.

To implement Integrated Security, do the following:

1. Start the OLAP server.

At this point, you must decide which OS/400 user profiles should have access to Essbase. You can give *all* valid OS/400 user profiles (both existing profiles and those created in the future) access to Essbase, or you can give a *subset* of OS/400 users access to Essbase. Regardless of your choice, you will still be able to give non-OS/400 users access to Essbase as well.

If you wish to give *all* current and future OS/400 users access to OLAP, skip to step 5; if you wish to only give a *subset* of OS/400 users access to OLAP continue with the next step.

2. First, you'll create an OS/400 group, using Warehouse Manager. Open Warehouse Manager. To add a Group User Profile, follow these steps:
  - a. In Explorer, connect to the AS/400 data source on which you want to create the profile.
  - b. On the **Manage** menu, choose **Profile**, then choose **Create Group**. The **Create Group Profile** dialog box appears.
  - c. Type a name and description for the profile.

The name for the group profile can be anything you wish. In this guide, we will assume you are using the name **ESSUSERS**.
  - d. Define any other settings on the other tabs. When setting up Integrated Security, you can accept the defaults that are displayed on these tabs. No special authorities are required.

- e. Click **Update** to update the AS/400 and refresh the profiles list in Explorer. Then, click **Close** to close the dialog box.

You have now created a Group User Profile (ESSUSERS) which identifies those OS/400 users who will have access to OLAP. Now, you will update the OLAP configuration file to identify the OS/400 group (and corresponding OLAP group, to be created later) which will specify the OS/400 users who will have OLAP access.

3. Without stopping the OLAP server, add the following line to the Essbase configuration file. (For information about editing the configuration file, see “Editing QSYS.LIB Files from a 5250 Display Session” on page 55.)

```
OS400USERSGROUP ESSUSERS
```

Where ESSUSERS refers to the name of the OS/400 group you just created

4. Now you can add OS/400 users to your new group, ESSUSERS. Each OS/400 user profile can be associated with one *primary* Group User Profile and several *supplemental* Group User Profiles. By establishing the ESSUSERS group as either the primary or supplemental group for an OS/400 user, you will indicate that the user has access to OLAP.

ESSUSERS can be either the primary OR supplemental group for any of your users. For example, it could be the primary group for Joe and the supplemental group for Bob.

To associate a Group User Profile with a User Profile, use Warehouse Manager to add users to the ESSUSERS group. How you do this depends on whether ESSUSERS will be the primary or secondary groups for a user:

- For those users who do not have a primary group, find ESSUSERS in the Explorer view. Right-click and then choose **Change Profile**. The **Create Group Profile** dialog will appear. Go to the **Members** tab and add all the necessary members. When you're finished, click **Update**.
- For those users who already have a primary group, you must add the users individually via the **Change User Profile** dialog box. To do this, locate the user in the Explorer view, right-click, and then choose **Change Profile**. The **Change Profile** dialog box will appear. Go to the **Group** tab.

From the list of all groups, choose **ESSUSERS**, and then click the double arrows button to add the group to the user's list of Current Supplemental Groups. Click **Update**. Repeat this for as many users as necessary.

5. **If you have completed steps 2-4, skip to step 6.**

You have chosen to allow all valid OS/400 users to access OLAP.

To give *all* valid OS/400 users access to OLAP, *without* stopping the OLAP server, add the following line to the Essbase configuration file. (For information about editing the configuration file, see “Editing QSYS.LIB Files from a 5250 Display Session” on page 55.)

```
OS400UsersGroup *ALL ESSUSERS
```

Where *ESSUSERS* is the group name for OLAP

Be sure to choose a group name that is NOT currently in use by Essbase.

6. At this point, you have specified which OS/400 users can access OLAP (either a subset of them, or all of them.) Now, using Essbase Application Manager, create the OLAP group (which was specified in the OLAP configuration file in Step 3 or Step 5) using the following steps:
  - a. Open Essbase Application Manager on your PC. Connect to the OLAP server. From the **Security** menu, choose **Users/Groups**. The **User/Group Security** dialog box will appear.
  - b. To create a new Essbase group, click **New Group**. The **New Group** dialog box will appear.
  - c. Create an OLAP Group that matches the group name specified in the OS400UsersGroup configuration file entry (in this case **ESSUSERS**). Specify the group type as User.
  - d. Click **OK**. The new Essbase group will be added. Click **Close** to close the **User/Group Security** dialog box.

Implementing Integrated Security shifts user management to OS/400 and Warehouse Manager. As a result, you will no longer add users to Essbase via Application Manager security. You should only use Application Manager security to create the Essbase group; after that, any user maintenance (e.g., adding, deleting users) should be done via OS/400.

1. Stop and restart the OLAP server. Integrated Security is now in effect.
2. Using Application Manager, from the **Security Menu**, choose **View Event Log**. Scroll to the bottom of the log to ensure there are no errors related to Integrated Security. If you had restarted the server prior to completing the necessary steps (such as creating the OS/400 or OLAP groups), you may see some error messages. All necessary groups must be created prior to stopping the server.
3. When you're finished viewing the log file, from the **File** menu, choose **Close**.

Congratulations! Now, as users log in to OLAP, OLAP will verify that the user name is an eligible Integrated Security OS/400 user, and if so, the password will be validated against the OS/400 user password. If the OS/400 user is deleted, the OLAP user will be automatically deleted. If the OS/400 user is disabled, the OLAP user will be disabled. Likewise, if the OS/400 password is expired, the OLAP password will be expired. In addition, the OS/400 user profile description will be reflected in the OLAP user description.

At this point, all Integrated Security users will have access only to the OLAP applications and databases to which the OLAP ESSUSERS group

has access. For instructions on controlling which OS/400 users have access to which applications and databases, see “Setting Up Group-Level Control”.

If all Essbase users will have the same level of access, you may choose to modify the rights of the OLAP ESSUSERS group so that as each Integrated Security user is automatically added to OLAP, the appropriate rights will be granted via the ESSUSERS group. See the Essbase Database Administrator’s Guide for information on how to do this. If, however, you wish to have different levels of access for different groups, do not give ESSUSERS any additional access; instead, continue with the following section.

## Setting Up Group-Level Control

Integrated Security can give you even more control by enabling you to specify at a group level which OS/400 users have access to particular applications and databases. For example, members of the Sales group can see the Sales database while members of the Human Resources group can see the Human Resources database.

To implement group-level security, you must define an OS400USERSGROUP (such as ESSUSERS). If you haven’t done that yet, see “Setting up OLAP Integrated Security” on page 73.

You may already have OS/400 group user profiles established for different departments or roles within your company. For example, you may have an OS/400 group user profile for access to Sales data, one for access to Accounting information, and another for access to HR data. If these group user profiles provide an appropriate method of defining who should have access to OLAP applications, you can use Integrated Security so that as users are added or removed from the OS/400 groups. Access to OLAP applications and databases will be adjusted accordingly. If you already have these groups defined, skip to step 2. Remember that members of these OS/400 groups must also be members of OS/400 group ESSUSERS (or the equivalent) to participate in OLAP Integrated Security.

1. If you have not yet defined OS/400 groups that make sense for your OLAP application and database security requirements, you will need to do so. Open Warehouse Manager. To add a Group User Profile, follow these steps:
  - a. In Explorer, connect to the AS/400 data source on which you want to create the profile.
  - b. On the **Manage menu**, choose **Profile**, then choose **Create Group**. The **Create Group Profile** dialog box appears.
  - c. Type a name and description for the profile.
  - d. The name for the group profile is your choice. In this guide, we will assume you are creating two new groups, SALES and HR.

- e. Define any other settings on the **General** through **Workstation** tabs. These settings will not have any effect on OLAP usage rights. Generally, you may accept the defaults.
  - f. Click **Update** to update the AS/400 and refresh the profiles list in Explorer. Then, click **Close** to close the dialog box.
  - g. Repeat steps A through E for each group you wish to create.
  - h. You can add users to these groups now or later. For those users who do not have a primary group, find ESSUSERS in the Explorer view. Right-click and then choose **Change Profile**. The **Create Group Profile** dialog will appear. Go to the **Members** tab and add all the necessary members. When you're finished, click **Update**. Remember that members in these groups must also be members of ESSUSERS or the equivalent.
2. At this point, you have created Group User Profiles (SALES, HR) that identify users who will have access to the Sales and HR applications and databases. Now, you will create these groups in OLAP.
    - a. Open Essbase Application Manager on your PC. Connect to the OLAP server. From the **Security** menu, choose **Users/Groups**. The **User/Group Security** dialog box will appear.
    - b. To create a new Essbase group, click **New Group**. The **New Group** dialog box will appear.
    - c. Create an OLAP Group that matches the name of the group you just created in Warehouse Manager (in this case HR). Specify the group type as User.
    - d. Click **OK**. The new Essbase group will be added.
    - e. Repeat steps A through D to create the Sales group.
  3. Next, you will adjust the application and database access as needed for each group. For example, the Sales group should have access to the Sales database; the HR group should have access to the HR database.
    - a. In Application Manager, on the **User/Group Security** dialog box, highlight a group in the **Group** box and click **Edit Group**. The **Edit Group** dialog box will appear.
    - b. In the **Group Type** box, choose **User**.
    - c. To specify the group's application access, click **App Access**. The **User/Group Application Access** dialog box will appear.
    - d. Choose an application from the **Applications** box and then set the access in the **Access** box. If you have created databases under the application, the **DB Access** button will be enabled if you choose **Access DBs**. To set the database access, click **DB Access**. The **User Database Access** dialog box will appear.
    - e. Choose a database in the **Database** box and set the access level for the group in the **Access** box.

- f. When you're finished, click **OK**. The **User/Group Application Access** dialog box will appear. Click **OK**. The **Edit Group** dialog box will appear. Click **OK**. The **User/Group Security** dialog box will appear. Click **Close**.
  - g. Repeat steps A through F for each group you want to set access for.
4. Next, add entries to the Essbase configuration file to tell OLAP which OLAP groups are participating in Integrated Security. Without stopping the OLAP server, add one line as shown below to the Essbase configuration file for each group you have defined: (For information about editing the configuration file, see "Editing QSYS.LIB Files from a 5250 Display Session" on page 55.)
 

```
OS400COMMONGROUP GROUPNAME
```

Where *GROUPNAME* refers to the name of the OLAP group you just created, for example, SALES or HR
  5. Stop and restart the OLAP server.
  6. Using Application Manager, from the **Security** menu, choose **View Event Log**. Scroll to the bottom of the log to ensure there are no errors related to Integrated Security. If you had restarted the server prior to completing the necessary steps (such as creating the OS/400 or OLAP groups), you may see some error messages. All necessary groups must be created prior to stopping the server.
  7. When you're finished viewing the log file, from the **File** menu, choose **Close**.

Congratulations! You have completed setup for Integrated Security.

For those users who are members of the OS/400USERSGROUP (i.e., ESSUSERS), membership in the "common" OLAP groups (e.g., Sales, HR) will be dynamically synchronized with the current membership status in the corresponding OS/400 common group.

Automatic creation of an OLAP user occurs when the user first logs into OLAP. Automatic deletion of users (if required), and synchronization of group memberships occurs at every log in, at OLAP startup, and when the **Security** dialog in Application Manager is opened.

### Troubleshooting Integrated Security

If you've followed all of the instructions above and Integrated Security still doesn't seem to be working, check this list for common problems and their fixes. After you fix any problems, you must shut down and restart the OLAP server in order to apply your changes.

- Your log file has any of the following error messages: 1180002:  
OS400USERSGROUP (ESSUSERS) is not an OS/400 Group User Profile,  
OS400USERSGROUP disabled 1180003: OS400USERSGROUP (ESSUSERS) is not an OLAP group, OS400USERSGROUP disabled 1180004:  
OS400COMMONGROUP (HR) ignored. It is not an OS/400 Group User

Profile. 1180005: OS400COMMONGROUP (HR) ignored. It is not an OLAP group. Verify that you have created the group profile in Warehouse Manager, entered the group in the Essbase configuration file, and added the group in Application Manager. Remember that the group name must be the same in all three locations.

- You've set up your special groups "HR" and "Sales" but the users are rejected when they try to log in.

Verify that the users are also members of the OS/400 ESSUSERS group. If you are enabling a subset of your users to access Essbase, those users must first be members of OS/400 ESSUSERS. Then you can add them to any additional OS/400 groups (e.g., Marketing, Sales) that you define.

- Your error file includes the following message: 1180008:  
OS400USERSGROUP (ESSUSERS) has been deleted, OS400USERSGroup disabled Verify that the OS400USERSGROUP is still defined in Application Manager. If the group has been deleted, it must be recreated for Integrated Security to work properly.

### **Automatic Deletion of OLAP Users**

There are two ways that OLAP users will be automatically deleted.

- If you have specified only a group name (e.g., ESSUSERS) in the OS400UsersGroup configuration entry (i.e., you did not specify \*ALL), then any user who is found to be in the OLAP group specified on the configuration entry, but is not a member of the OS/400 group, will be automatically deleted from Essbase. This assures that when an OS/400 user is deleted, the OLAP user will also be deleted.
- If you have specified \*ALL on the OS400UsersGroup entry, automatic deletion will occur when the OS/400 user profile is deleted.

### **What About Passwords?**

Passwords for OLAP user IDs created by the Integrated Security feature must be changed by changing the OS/400 password via the CHGPWD or CHGUSRPRF command. However, some OLAP client programs (including Application Manager and Spreadsheet Add-ins) will allow the end user to change a password at OLAP login time if the password has expired. For an Integrated Security User, this will change the OS/400 password as well.

Changing a password via the OLAP Application Manager **Security** menu option or via ESSCMD will not affect the OS/400 password.

OLAP normally requires that passwords be at least six characters long. OS/400 allows a wide variety of password rules to be optionally enforced and even allows a user exit for password screening. When using the Integrated Security feature, the normal OLAP requirement of six character passwords is no longer enforced. This will allow existing OS/400 users with shorter passwords to use Essbase without needing a new password. Please note that



any rules or exits defined under OS/400 will also apply to Integrated Security users when they attempt change an expired password via an Essbase interface.

### **Renaming Users and Groups**

When Integrated Security is enabled, certain renaming operations within OLAP are not permitted. The special purpose groups specified within the Essbase configuration file entries for OS400UsersGroup and OS400CommonGroup may not be renamed. In addition, any OLAP integrated user may not be renamed. These restrictions prevent the Integrated Security configuration from becoming inconsistent.

---

## **OLAP Jobs**

You can use the WRKACTJOB command to verify that your OLAP server is running. The server job will have a name identical to the job description name, a job type of BCH, and a function of PGM-ESSBASE. Applications run under separate jobs with the job name identical to the application name and a job type of BCI.

---

## **SQL Drill-Through Using Essbase Add-ins**

When creating a SQL Drill-Through profile, you must qualify an OLAP table with its library. For example, simply entering tablename will fail. Instead you must enter libname.tablename.

---

## **Message Logging**

When running the Essbase Application Server on platforms other than the AS/400, messages that track the requests being handled and any error messages are displayed to the terminal. Because of internal restrictions on the AS/400, most of these messages are recorded in the OLAP log file and are not displayed at the terminal. To access the OLAP log file on the AS/400, use the DSPESSLOG command.

---

## **Backing Up OLAP Files**

As with Essbase running on other platforms, OLAP does not provide any direct backup capability. Rather, it assumes the underlying operating system will provide adequate backup mechanisms.

You can use the SAVLIB or SAV commands to save OLAP applications and libraries. The following procedure creates two save files in QGPL which hold the QSYS file system and IFS file system information important to OLAP. This

procedure assumes OLAP is installed in the default library of ESSBASE. Substitute the name of your installation library for ESSBASE if you did not install OLAP into the default library.

Be sure to shut down the OLAP server before backing up OLAP files.

1. Type CRTSAVF FILE(QGPL/SAVEQSYS)
  2. Type SAVLIB LIB(ESSBASE) DEV(\*SAVF) SAVF(QGPL/SAVEQSYS) **or** SAV DEV('/QSYS.LIB/QGPL.LIB/SAVEQSYS.FILE') OBJ('/QSYS.LIB/ESSBASE.LIB')
  3. Type CRTSAVF FILE(QGPL/SAVEIFS)
  4. Type SAV DEV('/QSYS.LIB/QGPL.LIB/SAVEIFS.FILE') OBJ('/ESSBASE')
- All files found in the APP directory as well as the ESSBASE.SEC file must be backed up and restored together to ensure consistency.

---

## Restoring OLAP Files

If you used the procedure described in “Backing Up OLAP Files” on page 81, use the following procedure to restore your OLAP files.

1. Type RSTLIB SAVLIB(ESSBASE) DEV(\*SAVF) SAVF(QGPL/SAVEQSYS) **or** RST DEV('/QSYS.LIB/QGPL.LIB/SAVEQSYS.FILE') OBJ('/QSYS.LIB/ESSBASE.LIB')
2. Type RST DEV('/QSYS.LIB/QGPL.LIB/SAVEIFS.FILE') OBJ('/ESSBASE')

---

## Appendix A. Commands

This appendix lists many of the common AS/400 commands, Warehouse Manager commands, and OLAP commands that you will need to maintain your ShowCase AS/400 software.

---

### AS/400 Commands

The following table lists common AS/400 commands used to maintain a ShowCase library:

Table 13. Common AS/400 Commands

Command	Description	Usage
ADDLIBLE	Add a library to your library list	Adding a Warehouse Manager library to the library list when you need to run a ShowCase command from the AS/400 command line.
CFGTCP	Configure TCP/IP	To configure AS/400 TCP/IP for the ShowCase AS/400 software.
DSPPTF	Display PTFs	To find the latest PTFs installed on your system.
DSPSYSVAL	Find system information	To find system information, such as model and serial number. For example, to find the model number, enter DSPSYSVAL SYSVAL(QMODEL); to find the serial number, use QSRLNBR for the system value.
DSPUSRPRF	Display user profile	To display a user's profile.
RSTSMPLB	Restore sample database	This command will install the sample database if not previously installed, or it will restore the sample database back to its original structure (if database files were altered or destroyed).
STRTCP	Start AS/400 TCP	To start AS/400 TCP/IP. AS/400 TCP/IP must be started before you can enable or start ShowCase TCP/IP support.
WRKACTJOB	View active jobs	To see if ShowCase TCP/IP is running or to view active jobs, such as OLAP and Query jobs.
WRKOBJLCK	Check for jobs active against a ShowCase library	When checking for AS/400 jobs that may be active against a ShowCase library. In particular, checking for ShowCase TCP/IP jobs. Useful when installing over the top of an existing ShowCase library or debugging system-related problems (e.g., TCP/IP connections fail).

Table 13. Common AS/400 Commands (continued)

WRKSRVTBLE	View port numbers already in use.	When configuring TCP/IP, use this command to find an available port number.
------------	-----------------------------------	---

## Warehouse Manager Commands

The following tables list the common AS/400 Warehouse Manager commands used to maintain a ShowCase library. For related security information, see “ShowCase Command Authorities” on page -19.

Before using the following commands, add the ShowCase library to your AS/400 library list; ensure that you only have one ShowCase library in your library list. Run the command ADDLIBLE server-library, where server-library is the name of the ShowCase library with which you want to work

Table 14. Common Warehouse Manager Save/Restore Commands

Command	Description	Usage
SAVSRVRINF	Save server information	To back up the Warehouse Manager and Analyzer OLAP servers or move data stored on these servers to another library on the same AS/400 or a different AS/400.
RSTSRVRINF	Restore server information	To restore the server data saved using the SAVSRVRINF command.

Table 15. Common Warehouse Manager TCP/IP Commands

Command	Description	Usage
ADDSCTCP	Add ShowCase TCP/IP connectivity support	When you want to enable ShowCase TCP/IP after an install has finished. Requires *ALLOBJ, *SECADM, *IOSYSCFG, *JOBCTL, and *SAVSYS authorities.
ENDSCTCP	End ShowCase TCP/IP	To end ShowCase TCP/IP. For example, if you need to restart TCP/IP you should end ShowCase TCP/IP first. Requires *ALLOBJ, *SECADM, *IOSYSCFG, *JOBCTL, and *SAVSYS authorities.
RMVSCTCP	Remove ShowCase TCP/IP connectivity support	When you no longer want ShowCase TCP/IP support. Requires *ALLOBJ, *SECADM, *IOSYSCFG, *JOBCTL, and *SAVSYS authorities.

Table 15. Common Warehouse Manager TCP/IP Commands (continued)

STRSCTCP	Start ShowCase TCP/IP connectivity support	After an IPL and ShowCase TCP/IP has not started or if you have removed ShowCase TCP/IP and re-added it. Also used to restart ShowCase TCP/IP after issuing the ENDSCTCP command. Requires at least *ALLOBJ, *IOSYSCFG, and *JOBCTL authorities.
----------	--	--

Table 16. Common Warehouse Manager Server Information Commands

Command	Description	Usage
CHGSCAUT	Change/enter ShowCase passwords	To enter a password.
CHKSRVRCAT	Check Server Catalog Information	To inspect Warehouse Manager catalogs and delete obsolete Warehouse Manager settings.
CHGSRVRSTS	Change server status	To change the status of the Warehouse Manager server library (for example, SCSEVER) to and from a restricted state. A restricted state prevents users from connecting to the server library.
DSPSCAUT	Display ShowCase authority	To see the version of Warehouse Manager installed and the products and passwords.
DSPSCINF	Display ShowCase hints and tips	To locate hints and tips provided by ShowCase.

Table 17. Common Warehouse Manager Server Options Commands

Command	Description	Usage
CHGSRVRDFT	Change server defaults	To change the default *PUBLIC authority to all libraries or tables and enforce ShowCase security rules on *ALLOBJ profiles.
SETLIBACC	Set library list access	To restrict users so they can only access the libraries in their AS/400 library lists, or to give users access to all libraries on the AS/400 to which they are authorized.

## OLAP Server Commands

The following tables list the common OLAP commands used to maintain an OLAP library.

Before using the following commands, add the Essbase library to your AS/400 library list; ensure that you only have one Essbase library in your library list. Run the command ADDLIBLE essbase-library, where essbase-library is the name of the Essbase library with which you want to work

Table 18. Common General AS/400 Commands

Command	Description	Usage
CHGESSENV	Change OLAP environment variables	To change the environment
DSPESENV	Display OLAP environment variables	To know the version or the environment
DSPESSLOG	Display OLAP Logs	To read the server and application logs from the AS/400
ENDESSSVR	Stop the OLAP Server	To stop the server
STRESSCMD	Start OLAP command processor	To run any OLAP command or execute scripts
STRESSSVR	Start the OLAP Server	To start the server

Table 19. OLAP Uninstall Command

Command	Description	Usage
UNINSTSC	Uninstall OLAP	To remove an OLAP library from the system. For more information about the UNINSTSC save file, see “Removing Warehouse Manager Server Libraries” on page -6.

The following table lists keywords used for implementing Integrated Security. These keywords are added to the Essbase configuration file. See “Integrated Security” on page 73 for the context in which these keywords are used.

Table 20. Common OLAP Integrated Security Configuration File Settings

Command	Description	Usage
OS400UsersGroup	Enable all OS/400 users or a subset of OS/400 users to access OLAP	When setting up Integrated Security. Gives OLAP access to all users or a specified group of users.
OS400CommonGroup	Assign OS/400 user profiles to OLAP groups	When setting up Integrated Security. As OS/400 users are added or removed from OS/400 groups, they will be automatically added or removed from the corresponding OLAP group.

Table 20. Common OLAP Integrated Security Configuration File Settings (continued)

OS400AutoDelete	Enable automatic deletion of OLAP users	When setting up Integrated Security. Automatically deletes all users in the auto-delete group each time the OLAP server is started.
-----------------	---	---





---

## Appendix B. Sample Database

This appendix describes the sample database that you can install with Warehouse Manager server. The sample database tables are installed into a library called SCSAMPLExx, where xx is the current version of STRATEGY (e.g., SCSAMPLE30).

The sample database is the basis for the product tutorials. It contains data for a fictitious company called The Outdoor Connection. The Outdoor Connection is a sporting goods company that sells through three channels: retail stores, specialty stores, and catalogs. Its primary product lines are skiing equipment, biking equipment, camping gear, and clothing. The Outdoor Connection is based in the United States. It also does business in Canada, Belgium, Germany, France, the United Kingdom, Japan, Australia, and New Zealand.

---

### RSTSMPLB (Restore Sample Library)

This command will install the sample database if it was not previously installed. This command can also be used to restore the sample database back to its original structure (if database files were altered or destroyed). To use simply type RSTSMPLB at the command prompt. The command has two parameters.

1. CLRLIB (\*YES, \*NO) If \*YES is chosen then the sample database library (if it exists) is cleared (all files within destroyed) before the sample files are restored. Sample files are restored to their original version installed with this release.
2. PRVLGL (\*YES,\*NO) If \*YES is chosen then all logical files on the system that are dependent on the Builder sample files within the sample database library are preserved. Logical files that are dependent on NON-Builder files within the sample library are not preserved.

If the user wants to preserve logical files not dependant on sample files, choose \*NO for CLRLIB.

---

### The ORDERS Table

The ORDERS table is the basis of all transaction information. It contains the order number, the location placing the order, the date of the order, the promised delivery date, the date shipped, etc. The table also includes a column that records the location to which each order is shipped. This accommodates the situations in which the order is shipped to a location other than that where the order originates.

The ORDERS table also contains the product-related specifics of each order. Such information includes the quantity ordered, quantity shipped, quantity backlogged, and quantity canceled for each product. In addition, the table is used to track the unit price, unit cost, extended price, and extended cost for each product ordered. Finally, for international sites, the table includes exchange rates. The rates are used to calculate the foreign extended price and foreign extended cost for each product based on the exchange rate at the time of the order.

The exchange rate is an unsigned percentage and can be mapped to a currency code for each country in which The Outdoor Connection conducts business. It is assumed that all monetary amounts entered into the database are entered in U.S. dollars. Therefore, by definition, the exchange rate for U.S. dollars is 1.0. The exchange rate for other currencies is based on the amount that one U.S. dollar will buy. For example, \$1 US recently bought \$1.4575 CA. Therefore, the exchange rate for Canadian dollars is 1.4575. To convert an amount in the database into local values, multiply the amount by the exchange rate: (\$10 US X 1.4575 = \$14.58 CA or \$10 US X 0.6105 = £6.11).

This table contains 14,379 rows of data.

Table 21. The ORDERS Table

Column	Description	Heading	Data Type	Contents
ORDNUM	Order Number	Order Number	Decimal(8)	Order number
CUST_ID	Customer ID	Customer ID	Numeric(8)	Account ID of client
SHIP_TO	Shipping Destination	Ship To	Numeric(8)	Account ID of shipping destination
PRODUCTID	Product ID (SKU)	Product ID	Numeric(8)	Product ID
PRDLVL03	Product Level 3	Product Level 3	Char(8)	Third-tier product category
QTYORD	Quantity Ordered	Quantity Ordered	Decimal(15)	Quantity ordered
QTYSHIP	Quantity Shipped	Quantity Shipped	Decimal(15)	Quantity shipped
QTYBACK	Quantity Backordered	Quantity Backordered	Decimal(15)	Quantity back-ordered
QTYCAN	Quantity Cancelled	Quantity Cancelled	Decimal(15)	Quantity cancelled
DATEREQ	Date Requested	Date Requested	Date	Date requested for delivery
DATEORD	Date Ordered	Date Ordered	Date	Date ordered
DATEPROM	Date Promised	Date Promised	Date	Date promised for delivery

Table 21. The ORDERS Table (continued)

DATESHIP	Date Shipped	Date Shipped	Date	Date shipped
DATEINV	Date Invoiced	Date Invoiced	Date	Date invoiced
UNIT_PRICE	Unit Price	Unit Price	Decimal(15,4)	Unit retail price
UNIT_COST	Unit Cost	Unit Cost	Decimal(15,4)	Unit wholesale cost (cost of goods sold)
EXTPRICE	Extended Price	Extended Price	Decimal(15,4)	Extended retail total (the product of QTYORD and UNIT_PRICE)
EXTCOST	Extended Cost	Extended Cost	Decimal(15,4)	Extended wholesale total (the product of QTYORD and UNIT_COST)
MEASURE	Unit of Measure	Unit of Measure	Char(4)	Unit of measure code
CURR_CODE	Currency Code	Currency Code	Char(3)	Currency code
EXCHG_RATE	Exchange Rate	Exchange Rate	Decimal(15,7)	Exchange rate
FRGN_XPRIC	Foreign Extended Price	Foreign Ext Price	Decimal(15,4)	Foreign extended retail total
FRGN_XCOST	Foreign Extended Cost	Foreign Ext Cost	Decimal(15,4)	Foreign extended wholesale total
DIVSN_ID	Division ID	Division ID	Char(8)	Division code
REGION_ID	Region ID	Region ID	Char(8)	Region code
AREA_ID	Area ID	Area ID	Char(8)	Area code
DIST_ID	Distribution Center ID	Distribution Cntr ID	Char(8)	Distribution center code
REPCODE	Sales Rep Code	Rep Code	Char(4)	Account representative's code

## The CUSTOMERS Table

The CUSTOMERS table contains information about The Outdoor Connection's clients. It includes name and address information and it divides the clients into various geographical categories. The Outdoor Connection consists of three divisions (North America, Europe, and Asia/Pacific); 12 regions (East, West, South, Midwest, Canada, Belgium, Germany, France, United Kingdom, Japan, Australia, and New Zealand); 70 areas (one for each state and province, including the District of Columbia, plus one for each European and Asia/Pacific country); and three distribution facilities (Seattle, Kansas City, and New York). In addition, the Customers table includes various other information, including a currency code for each client, a status code that tracks whether a client is active or inactive, and a sales representative code that identifies each client's sales representative.

The CUSTOMERS table contains sufficient data to analyze customers according to two separate hierarchical structures:

Sales organization structure

- Division
- Region
- Area

Geographic location

- Country
- State/Province
- City
- Postal code

This table contains 157 rows of data.

Table 22. The CUSTOMERS Table

Column	Description	Heading	Data Type	Contents
CUST_ID	Customer ID	Customer ID	Numeric(8)	Account ID of client
CUSTNAME	Customer Name	Customer Name	Char(100)	Name of client
LNAME	Surname	Last Name	Char(50)	Surname of client contact
FNAME	Given Name	First Name	Char(50)	Given name of client contact
ADDRESS1	Address 1	Address 1	Char(100)	Address line 1
ADDRESS2	Address 2	Address 2	Char(100)	Address line 2
CITY	City	City	Char(50)	City name
STATE	State/Province Abbreviation	State/Province Abbr	Char(2)	State/province postal abbreviation
STATEDESC	State/Province Description	State/Province	Varchar(80)	State/province name
POSTALCODE	Postal Code	Postal Code	Char(16)	Postal code
COUNTY	County	County	Varchar(80)	County name
COUNTRY	Country Code	Country Code	Varchar(12)	Country code
COUNTRYDESC	Country Description	Country	Varchar(80)	Country name
PHONE	Phone Number	Phone Number	Char(24)	Phone number
E_ADDRESS	E-mail Address	E-mail Address	Char(100)	E-mail address of client contact
LOB	LOB	LOB Code	Char(8)	Line of business code

Table 22. The CUSTOMERS Table (continued)

LOBDESC	Line of Business Description	Line of Business	Char(80)	Line of business description
REPCODE	Sales Rep Code	Rep Code	Char(4)	Account representative's code
ENTRY_DATE	Entry Date	Entry Date	Date	Date added
ACTIVE	Active Code	Active	Char(1)	Active status (Y or N)
DIVSN_ID	Division ID	Division ID	Char(8)	Division code
DIVSN	Division	Division	Char(80)	Division name
REGION_ID	Region ID	Region ID	Char(8)	Region code
REGION	Region	Region	Char(80)	Region name
AREA_ID	Area ID	Area ID	Char(8)	Area code
AREA	Area	Area	Char(80)	Area name
DIST_ID	Distribution Center ID	Distribution Cntr ID	Char(8)	Distribution center code
DIST_LOC	Distribution Center	Distribution Center	Char(80)	Distribution center name
CURR_CODE	Currency Code	Currency Code	Char(3)	Currency code

## The PRODUCTS Table

The PRODUCTS table contains all of The Outdoor Connection's product information and groups products hierarchically. Each product has three levels associated with it. The first identifies a product group (e.g.; clothing, biking, skiing, etc.), the second a type (e.g.; outerwear, shirts, pants, etc.), the third a style (e.g.; black, red, yellow, etc.). In addition, the PRODUCTS table includes a unit price and a unit cost for each product. It also contains a currency code so that the unit price and cost can be converted for various countries. Finally, it includes a measures column that a unit type (e.g.; *one* bicycle, *a pair* of skis, etc.) and a date that records when information was last changed.

This table contains 120 rows of data.

Table 23. The PRODUCTS Table

Column	Description	Heading	Data Type	Contents
PRODUCTID	Product ID (SKU)	Product ID	Numeric(8)	Product ID
PRODUCT	Product Description	Product Description	Char(80)	Product description
PRDLVL01	Product Level 1	Product Level 1	Char(8)	First tier product category

Table 23. The PRODUCTS Table (continued)

PRDDESC01	Product Description 1	Description Lvl 1	Char(80)	First tier product description
PRDLVL02	Product Level 2	Product Level 2	Char(8)	Second tier product category
PRDDESC02	Product Description 2	Description Lvl 2	Char(80)	Second tier product description
PRDLV03	Product Level 3	Product Level 3	Char(8)	Third tier product category
PRDDESC03	Product Description 3	Description Lvl 3	Char(80)	Third tier product description
PRDLVL04	Product Level 4	Product Level 4	Varchar(8)	Fourth tier product category
PRDDESC04	Product Description 4	Description Lvl 4	Varchar(27)	Fourth tier product description
PRDLV05	Product Level 5	Product Level 5	Varchar(8)	Fifth tier product category
PRDDESC05	Product Description 5	Description Lvl 5	Varchar(27)	Fifth tier product description
UNITPRICE	Unit Price	Unit Price	Decimal(15,4)	Unit retail price
UNITCOST	Unit Cost	Unit Cost	Decimal(15,4)	Unit wholesale cost (cost of goods sold)
CURR_CODE	Currency Code	Currency Code	Char(3)	Currency code
MEASURE	UOM Code	UOM Code	Char(4)	Unit of measure code
MEASURE_DESC	Unit of Measure	Unit of Measure	Char(80)	Unit of measure description
ENTRY_DATE	Entry Date	Entry Date	Date	Entry/update date

## The SALESORG Table

The SALESORG table provides a key to the geographic structure used in the database. Essentially, it maps the various divisions, regions, and areas to each other. It also indicates which distribution facility serves each area or foreign region. As explained previously, the database contains three divisions, 12 regions, 70 areas, and three distribution facilities.

This table contains 70 rows of data.

Table 24. The SALESORG Table

Column	Description	Heading	Data Type	Contents
DIVSN_ID	Division ID	Division ID	Char(8)	Division code
DIVSN	Division	Division	Char(80)	Division name
REGION_ID	Region ID	Region ID	Char(8)	Region code

Table 24. The SALESORG Table (continued)

REGION	Region	Region	Char(80)	Region name
AREA_ID	Area ID	Area ID	Char(8)	Area code
AREA	Area	Area	Char(80)	Area name
DIST_ID	Distribution Center ID	Distribution Cntr ID	Char(8)	Distribution center code
DIST_LOC	Distribution Center	Distribution Center	Char(80)	Distribution center name

## The SALESREPS Table

The SALESREPS table maps a salesperson's code to a name, employee ID, and manager code. It also maps the salespeople to regions so that salespeople can be tracked and evaluated by geography. Each domestic region contains three sales people, Canada and each European region have two, and each Asia/Pacific region has one.

This table contains 25 rows of data.

Table 25. The SALESREPS Table

Column	Description	Heading	Data Type	Contents
EMP_ID	Employee ID	Employee ID	Numeric(8)	Employee number
REPCODE	Sales Rep Code	Rep Code	Char(4)	Account representative's code
MGRCODE	Manager Code	Manager Code	Char(4)	Supervisor's code
LASTNAME	Surname	Last Name	Varchar(100)	Surname of employee
FIRSTNAME	Given Name	First Name	Varchar(100)	Given name of employee
REGION_ID	Region ID	Region ID	Char(8)	Region code
REGION	Region	Region	Varchar(80)	Region name
ENTRY_DATE	Entry Date	Entry Date	Date	Entry date (date of hire)
ACTIVE	Active	Active	Char(1)	Active status (Y or N)

## The DATETYPES Table

The DATETYPES table is used to demonstrate how DB2 OLAP Server for AS/400 handles date values stored as a data type other than DATE. Specifically, the DATEINV column has been replicated into several formats. You can use this table independently or join it to the ORDERS table on ORDNUM. It contains 14,379 rows of data.

Table 26. The DATETYPES Table

Column	Description	Heading	Data Type
ORDNUM	Order Number	Order Number	Decimal(8)
DATEINV	Date Invoiced	Date Invoiced	Date
DATEINV_YY	Two Digit Year Value	YY	Decimal(2)
DATEINV_MM	Two Digit Month Value	MM	Decimal(2)
DATEINV_DD	Two Digit Day Value	DD	Decimal(2)
DATEINV_EXCELSRL	Five Digit Excel Serial Date	EXCELSRL	Decimal(5)
DATEINV_YYDDD	Five Digit YYDDD Date	YYDDD	Decimal(5)
DATEINV_YYMMDD	Six Digit YYMMDD Date	YYMMDD	Decimal(6)
DATEINV_YYDDMM	Six Digit YYDDMM Date	YYDDMM	Decimal(6)
DATEINV_DDMMYY	Six Digit DDMMYY Date	DDMMYY	Decimal(6)
DATEINV_MMDDYY	Six Digit MMDDYY Date	MMDDYY	Decimal(6)
DATEINV_CYYDDD	Six Digit CYYDDD Date	CYYDDD	Decimal(6)
DATEINV_CYYMMDD	Seven Digit CYYMMDD Date	CYYMMDD	Decimal(7)
DATEINV_CYYDDMM	Seven Digit CYYDDMM Date	CYYDDMM	Decimal(7)
DATEINV_YYYYDDD	Seven Digit YYYYDDD Date	YYYYDDD	Decimal(7)
DATEINV_YYYYMMDD	Eight Digit YYYYMMDD Date	YYYYMMDD	Decimal(8)
DATEINV_YYYYDDMM	Eight Digit YYYYDDMM Date	YYYYDDMM	Decimal(8)
DATEINV_MMDDYYYY	Eight Digit MMDDYYYY Date	MMDDYYYY	Decimal(8)
DATEINV_DDMMYYYY	Eight Digit DDMMYYYY Date	DDMMYYYY	Decimal(8)
DATEINV_CHAR	Eight Character YYYYMMDD Date	YYYYMMDD	Char(8)
DATEINV_HYF	Five Digit Infinium Date (EXCELSRL - 1)	HYF	Decimal(5)



---

## The TIMEDIM Table

The TIMEDIM table contains time-related information for specific dates. This table enables you to perform a variety of time-related analyses.

This table contains 2,192 rows of data.

*Table 27. The TIMEDIM Table*

Column	Description	Heading	Data Type	Contents
TDATE	TDATE	TDATE	Date	Date
TYEAR	TYEAR	TYEAR	Integer	Year
TMONTH	TMONTH	TMONTH	Integer	Month number
TDAY	TDAY	TDAY	Integer	Day of month
DAYOFWEEK	DAYOFWEEK	DAYOFWEEK	Integer	Day of week number
DAYNAME	DAYNAME	DAYNAME	Varchar(9)	Day
DAYOFYEAR	DAYOFYEAR	DAYOFYEAR	Integer	Day of year number
TWEEK	TWEEK	TWEEK	Integer	Week number
HOLCODE	HOLCODE	HOLCODE	Integer	Holiday code
HOLIDAY	HOLIDAY	HOLIDAY	Varchar(30)	Holiday name

---

## The SALESPERF Table

The SALESPERF table is used to track a number of performance measurements. It includes sales projections as well as actual sales information. It also includes sales quota information for each sales representative. The table also allows you to determine actual and projected sales, as well as quotas, on a geographical basis.

This table contains 12,008 rows of data.

*Table 28. The SALESPERF Table*

Column	Description	Heading	Data Type	Contents
PERIOD	Period	Period	Date	Time period
REPCODE	Sales Rep Code	Rep Code	Char(4)	Account representative's code
AREA_ID	Area ID	Area ID	Char(8)	Area code
PRODUCTID	Product ID (SKU)	Product ID	Numeric(8)	Product ID

Table 28. The SALES PERF Table (continued)

PRJ_UNITS	Projected Unit Sales	Projected Units	Numeric(15,0)	Projected unit sales
ACT_UNITS	Actual Unit Sales	Actual Units	Numeric(15,0)	Actual units sold
PRJ_SALES	Projected Sales	Projected Sales	Numeric(15,4)	Projected sales amount
ACT_SALES	Actual Sales	Actual Sales	Numeric(15,4)	Actual sales amount
QUOTA	Quota	Quota	Numeric(15,4)	Sales quota

## The EMPLOYEES Table

The EMPLOYEES table contains human resources information about The Outdoor Connection's employees. Specifically, it lists employees' home address and phone number, their e-mail address, their gender, their date of birth, and, where applicable, their social security number. It also contains employees' compensation type and rate, their hire date, and, where applicable, a termination date and reason.

This table contains 39 rows of data.

Table 29. The EMPLOYEES Table

Column	Description	Heading	Data Type	Contents
EMP_ID	Employee ID	Employee ID	Numeric(8)	Employee number
LASTNAME	Surname	Last Name	Varchar(100)	Surname of employee
FIRSTNAME	Given Name	First Name	Varchar(100)	Given name of employee
ADDRESS1	Address 1	Address 1	Char(100)	Address line 1
ADDRESS2	Address 2	Address 2	Char(100)	Address line 2
CITY	City	City	Char(50)	City name
STATE	State/Province Abbreviation	State/Province Abbr	Char(2)	State/province postal abbreviation
POSTALCODE	Postal Code	Postal Code	Char(16)	Postal code
COUNTRY	Country	Country	Char(50)	Country name
PHONE	Phone Number	Phone Number	Char(24)	Phone number
E_ADDRESS	E-mail Address	E-mail Address	Char(100)	E-mail address of client contact
SEX	Gender	Sex	Char(1)	Gender
DOB	Date of Birth	Date of Birth	Date	Date of birth

Table 29. The EMPLOYEES Table (continued)

SSN	Social Security Number	Social Security No.	Char(11)	Social security number
COMPTYPE	Compensation Type	Comp Type	Char(1)	Compensation type (commission, salary, hourly)
SALARY	Base Monthly Salary	Base Salary	Numeric(10,2)	Base monthly salary
HOURLY	Hourly Rate	Hourly Rate	Numeric(4,2)	Hourly wage
COMMISSION	Commission Rate	Commission	Numeric(4,4)	Commission percentage
CURR_CODE	Currency Code	Currency Code	Char(3)	Currency code
HIREDATE	Hire Date	Hire Date	Date	Hire date
TERMDATE	Termination Date	Termination Date	Date	Termination date
TERMTYPE	Termination Reason	Termination Reason	Char(20)	Reason for termination



---

## 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, 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.

Licensees 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.

---

## Trademarks

DB2 OLAP Server	IBM	OS/390
OpenEdition	DB2	AIX
DB2 Universal Database	AS/400	OS/400

Lotus and 1-2-3 are trademarks of Lotus Development Corporation in the United States, or other countries, or both.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Microsoft, Windows, and Windows NT are registered trademarks of Microsoft Corporation.

ShowCase is a registered trademark of ShowCase Corporation, and STRATEGY is a trademark of Showcase Corporation.

Other company, product, and service names may be trademarks or service marks of others.

---

## Glossary

This glossary defines terms that are commonly used in DB2 OLAP Server for AS/400 as well as terms commonly used in discussing data warehousing and information analysis. The *Using DB2 OLAP Server* book includes additional glossary terms.

**access rules.** Restrictions that determine how and whether a user can view AS/400 data. For example, Warehouse Manager access rules can secure data down to the row and column level, can control the amount of AS/400 resources a user can consume, and can create and manage user and group profiles.

**active application.** The application that is affected by all commands issued until you select another application.

**active workbook.** The workbook that is top most in the application.

**active worksheet.** The worksheet that is top most in the active workbook.

**add-ins.** Additional menu items a user can add to Microsoft Excel and Lotus 1-2-3 when installing DB2 OLAP Server for AS/400. These menu items provide additional functions which enable query data to populate Excel and 1-2-3 spreadsheets.

**aggregate.** See *consolidate*.

**alias.** An alternate name for an object. In DB2 OLAP Server for AS/400, aliases can be assigned to database objects such as collections and tables as well as to dimensions and members.

**analysis, multidimensional.** The objective of multidimensional analysis is to allow end users to quickly and easily gain insight from an organization's data. The multidimensional approach to analysis stages data in a manner consistent with the analyst's mental model,

thereby reducing confusion and lowering the incidence of erroneous interpretations. It also makes it faster and easier to navigate the database, screen for a particular subset of information, view the data in a particular orientation, and define analytical calculations. Furthermore, because data is physically stored in a multidimensional structure, these operations are many times faster and more consistent than is possible with other database structures. This combination of simplicity and speed is among the key benefits of multidimensional analysis.

**ancestor.** In a database outline, any member in a branch hierarchically above another member. For example, Year and Quarter are ancestors of April.

**ANSI character set.** The American National Standards Institute character set, consisting of 256 characters.

**API.** See *application program interface*.

**APPC.** The Advanced Program-to-Program Communications protocol. The APPC communications protocol enables programs on different computers to communicate and exchange data directly. A well-designed applet can be invoked from many different applications.

**applet.** A program designed to be executed from within another application. Unlike an application, applets cannot be executed directly from the operating system.

**application.** In DB2 OLAP Server for AS/400, a management structure containing one or more OLAP databases and related files that control many system variables such as memory allocation and autoloading parameters.

**application program interface.** A documented set of functions that allow two software programs to communicate and provide services.

**array, multi-dimensional.** (1) A group of data cells arranged by the hierarchy of the data. A spreadsheet exemplifies a two-dimensional array with the data cells arranged in rows and columns, each being a dimension. A three-dimensional array can be visualized as a cube with each dimension forming a side of the cube, including any slice parallel with that side. Higher dimensional arrays have no physical metaphor, but they organize the data in the way users work in their enterprise. Typical enterprise dimensions are time (e.g., year, quarter, month), measures (e.g., price, cost of goods sold, quantity), products (e.g., skis, tents, bicycles), geographical regions (e.g., domestic, Midwest, Iowa), sales channels, (e.g., mail order, retail, field sales), etc. (2) **Synonyms:** multidimensional data structure, cube, hypercube

**ASCII character set.** The American Standard Code for Information Interchange 7-bit character set, which consists of 128 characters. It is the most universal character-coding set used by computers.

**asynchronously.** Regarding queries, asynchronously refers to a situation in which multiple queries run simultaneously and in no particular order. The query that starts running first will not necessarily finish first.

**attribute.** A column characteristic that describes the data type and length of the column values.

**AVG.** (1) A summary function used to group similar data and find the average of the grouped values in the column where the function was applied. (2) This function first groups the rows with identical data in those columns without any summary functions applied. Then one summarized row is returned for each group of rows which includes identical values and the average of the grouped values in the selected column.

**batch job.** A query that is sent to an AS/400 system and runs as a background task. The AS/400 system then runs the query when enough system resources are available and saves the results to a temporary storage space or intermediate table.

**block.** The primary storage unit within DB2 OLAP Server for AS/400. A block is a multidimensional array representing the cells of all dense dimensions.

**Boolean operator.** One of three operators, AND, OR, and NOT, used by a search condition to define a true or false relationship between two expressions. For example, if you specify X=1 AND Y=5, the search condition is true if both expressions match the values found by the search condition.

**break group.** Rows of returned data that are grouped according to a common column value. For example, in a column of states, the rows of data for each state are grouped together.

**cache.** A special high-speed storage mechanism. It can be either a reserved section of main memory or an independent high-speed storage device.

**calc script.** See *calculation script*.

**calculated member.** (1) A member whose value is determined by the values of other members through the use of a mathematical or logical operation. For example, the member profit can be calculated by subtracting the value of cost of goods sold from price. Calculated members may be calculated and stored in an OLAP server database or may be specified by a user during an interactive session. (2) **Antonym:** input member

**calculation.** An equation within a database outline, a calculation script, or a report script that is used to determine a value for a particular member or point in a report.

**calculation script.** (1) A text file that contains instructions for how specified data sets within an OLAP database should be calculated. (2) **Synonym:** calc script

**CCSID.** See *Coded Character Set Identifier*.

**cell.** (1) A single data point that occurs at the intersection defined by selecting one member from each dimension in a multidimensional array. For example, if the dimensions are measures, time, product, and geography, then the



dimension members Quantity, January 1998, Skis, and Wisconsin specify a precise intersection along all dimensions. That intersection uniquely identifies a single data cell, which in this case contains the number of skis sold in Wisconsin during January 1998. (2) **Synonym:** member combination

**child.** In a database outline, any member that is hierarchically below another member. For example, January is a child of Quarter 1.

**children.** (1) In a database outline, members of a dimension that are included in a calculation to produce a consolidated total for a member that is hierarchically greater. Children may themselves be consolidated, which requires that they have children of their own. For example, Quarter 1, Quarter 2, Quarter 3, and Quarter 4 are children of Year, while January, February, and March are children of Quarter 1. (2) A member may be a child for more than one parent (e.g., January can be a child of Quarter 1 and of Winter), and a child's multiple parents may not necessarily be at the same hierarchical level (e.g., December can be a child of Quarter 4 and of Holiday Season), thereby allowing complex, multiple hierarchical aggregations within any dimension.

**client.** A PC workstation connected to a server, such as an AS/400.

**client/server.** A computing framework where personal computers and database servers cooperate to provide improved end-user access to data.

**Coded Character Set Identifier.** A CCSID is a 2-byte (unsigned) binary number that uniquely identifies an encoding scheme and one or more pairs of character sets and code pages.

**collection.** An object that consists of, and logically classifies, a set of database objects.

**column.** A set of values in a table where all values are of the same data type. These values are displayed vertically. Also known as a field.

**column dimension.** See *page display*.

**column heading.** A title at the top of column of values in a table which describes the values in that column. Also known as a field label.

**communications software.** Software that enables a computer to connect with another computer and to exchange information. Communications software can maintain settings for the connection, coordinate transmission of data and messages, as well as other tasks relating to the connection between the computer systems.

**concatenate.** To combine two pieces of data in sequence. For example, if you concatenated the words "thank" and "you" it would read "thankyou".

**condition.** (1) A selection criterion that specifies the rows to include in data returned from the database. A condition can either be true or false for any row in the applicable table. Only rows for which the condition is true are returned when the query is run. (2) For example, a condition might be used to specify a range of dates for which data is returned or it might specify specific states for which data is returned.

**consolidate.** (1) To compute a relationship for one or more hierarchies of data. For example, consolidating the total sales for January, February and March by adding them together results in the total sales for Quarter 1. While such relationships are normally summations, any type of computational relationship or formula can be defined. Additionally, although consolidate is typically used when talking about multidimensional databases, relational databases commonly contain consolidated data as well. (2) **Synonyms:** roll-up, aggregate

**constant.** A value (string, numeric or date/time) in a table that does not vary and is therefore the same for any row.

**COUNT.** A summary function that returns the number of rows included in a summarized set of data or break group.

**cube.** See *array, multidimensional*.

**currency conversion.** A factor that converts monetary values in an OLAP database from one

country's monetary system into another without altering the original data.

**currency partition.** A dimension type that separates local currency members for a base currency defined in an application. Also identifies currency types such as actual, budget, and forecast.

**data boutique.** Used to describe the smallest decision support database. Typically a data boutique resides on a personal computer hard-drive that can be shared by users.

**data cache.** A buffer in memory that holds data blocks.

**data compression.** Reducing the size of computer data, to enable it to be sent across a network with greater efficiency. The data is later expanded to its original size through decompression.

**data load.** See *load data*.

**data load rules.** A set of operations that DB2 OLAP Server for AS/400 performs on data from an external data source file as it is loaded into an OLAP database.

**data mart.** Used to describe a relatively small, subject-specific database, often departmental or line-of-business. Typically, a data mart resides on a server such as an AS/400 that contains OLTP, decision support or multidimensional databases, access rules and metadata. Data Marts are typically structured and optimized for end-user information analysis as opposed to online transaction processing.

**data mining.** In general, a method of data analysis that identifies relationships within detail data using historical analysis. Data mining often uncovers unexpected or unanticipated relationships. For example, a retail store might never notice that a sale on sleds results in a larger demand for hot cocoa mix. Data mining can uncover such a relationship and prompt a store to never run a promotion on both items at the same time.

**data view.** A data view is a simplified view of an AS/400 database. In practice, data views enable database administrators or business analysts to define table join criteria, build frequently-used result columns, summarize detail data (when necessary), remove unnecessary columns of data from view, build frequently used Sub-SELECT support, and define understandable column names.

**data warehouse.** Used to describe a relatively large database that contains corporate information that has been transferred and transformed from OLTP systems for end-user consumption. Typically a data warehouse resides on a large server, such as an AS/400 or a combination of AS/400s. Data warehouses are often a subject-oriented information store, designed specifically for decision support. A data warehouse can contain data marts and data boutiques.

**database.** A collection of one or more tables and views. In DB2 OLAP Server for AS/400, a repository of data that contains a multi-dimensional data storage array. Each OLAP database consists of a storage structure definition (a database outline), data, security definitions, and optional calculation scripts. An OLAP application contains one or more databases.

**database administrator.** The person whose job encompasses enabling end-user access to data. Often called a DBA.

**database filter layer.** A layer in the DB2 OLAP Server for AS/400 security plan that defines specific settings for database members down to the cell level.

**database management system.** The software between the database and the user. The software tracks physical details of the database and manages all query requests.

**data source.** (1) A system, database, or other information storage method that contains information stored in tables. (2) For example, the IBM AS/400 system, Microsoft SQL Server, Borland dBASE files, and Microsoft Excel files are all common data sources.

**data type.** An attribute that specifies which operations can be performed on a value. Example data types are character, integer, and decimal.

**DB2/400.** DB2 is IBM's standard database architecture across IBM platforms. DB2/400 is the AS/400 implementation of this architecture. This is typically how relational data is stored on the AS/400.

**DBA.** See *database administrator*.

**decision support database.** Databases designed for end-user information analysis that typically provide fast response to ad hoc usage. Decision support databases are generally denormalized.

**decision support systems.** Ad hoc query, reporting and complex analysis tools designed for accessing and transforming raw data into useful information.

**denormalize.** (1) The process of putting normalized data in a physical location and structure that is optimized for performance and data analysis. A denormalized database contains no joins, includes summarized data and includes redundant data. (2) **Antonym:** normalize

**dense.** (1) Describes a dimension in a multidimensional database with a high probability for occupying one or more data point in every combination of dimensions that occurs. For example, in a typical multidimensional database, the measures dimension is dense since it contains accounts data (e.g., quantity sold, cost of goods sold, etc.) and since accounts data exists for virtually all products in all markets. (2) **Antonym:** sparse

**derived data.** (1) Derived data is produced by applying calculations to input data at the time the request for that data is made (i.e., the data has not been pre-computed and stored in the database). The purpose of using derived data is to save storage space and calculation time, particularly for calculated data that may be infrequently called for or that is susceptible to a high degree of interactive personalization by the

user. The trade-off is slower retrievals. (2) **Antonym:** pre-calculated data

**derived members.** Members whose associated data is produced by applying calculations to input data at the time the request for that data is made.

**descendant.** In a database outline, any member that has a parent above it hierarchically. For example, Quarter 1 and January are descendants of Year.

**DDE (Dynamic Data Exchange).** The process of linking two Microsoft Windows compatible applications to share data. For example, you can create a link between a Microsoft Word document and a Query file. Query updates the data in the Word document when the data in the file changes.

**detail member.** See *leaf member*.

**dimension build rules.** Similar to data load rules, but they also modify the outline based on data in the external data source file.

**dimension.** A dimension is a structural attribute of a cube that is a list of members, all of which are of a similar type in the user's perception of the data. For example, all months, quarters, years, etc., make up a time dimension. Likewise all cities, states, regions, countries, etc., make up a geography dimension. A dimension acts as an index for identifying values within a multidimensional array. If one member of the dimension is selected, then the remaining dimensions in which a range of members (or all members) are selected defines a sub-cube. If all but two dimensions have a single member selected, the remaining two dimensions define a spreadsheet (or a slice or a page). If all dimensions have a single member selected, then a single cell is defined. Dimensions offer a very concise, intuitive way of organizing and selecting data for retrieval, exploration and analysis. In an OLAP database outline, the dimensions represent the highest consolidation level.

**distinct.** Refers to the use of the SQL DISTINCT keyword. When this keyword is used in a query,

only one copy of any duplicate rows are returned with the rest of the result data.

**distinct prompt.** (1) When a query is run, a *prompt* asks the user to enter a value that will narrow the amount of data returned from the data source. A *distinct prompt* requires that the value entered match a value defined in the database. (2) For example, a user who queries a database for sales figures can be prompted to enter the sales region for which sales figures will be returned. The region the user enters is required to match a region as defined in the database, such as Upper Midwest.

**Distrubuted Relational Database Architecture.** DRDA is the IBM AS/400 implementation for database operations on remote AS/400s.

**DRDA.** See *Distributed Relational Database Architecture*

**drill down.** Drilling down or up is a specific analytical technique whereby the user navigates among levels of data ranging from the most summarized (up) to the most detailed (down). The drilling paths may be defined by the hierarchies within dimensions or other relationships that may be dynamic within or between dimensions. For example, when viewing sales data for North America, a drill-down operation in the Region dimension would then display Canada, the eastern United States and the Western United States. A further drill-down on Canada might display Toronto, Vancouver, Montreal, etc.

**drill through.** An analytical technique whereby when a user reaches the lowest level of a consolidated multidimensional database the user can seamlessly view associated data that resides in a relational data database. For example, imagine a multidimensional database in which the lowest hierarchy in the time dimension is month. A user looking at sales information for April is said to drill through the database when they are able to automatically retrieve sales data from a relational database for a specific day in April. From the user's perspective, drill through is seamless, although since the detail data is

retrieved from a relational database, response time will likely slow down.

**dynamic calculation.** A calculation that occurs when you report data on a member that has been tagged as Dynamic Calc. The member's values are calculated at retrieval time instead of being precalculated during batch calculation.

**edit mask.** A predefined format that users can apply to data in a viewer.

**EBCDIC.** See *extended binary-code decimal interchange code*.

**EIS.** A generic solution category in the information analysis arena. Formerly executive information system, many companies are redefining EIS as enterprise information system or everyone's information system since the solution category isn't just for executives.

**ESSCMD/400.** A command line interface used to perform OLAP server operations interactively or through a batch file.

**expression.** A series of operations performed on a column value.

**extended binary-code decimal interchange code (EBCDIC).** A coded character set consisting of 8-bit coded characters.

**file extension.** The three characters added to the end of a filename to identify its parent program. For example, Query appends the letters DBQ to each file created.

**filter.** See *selection*.

**formula.** A database object that is a calculation, rule or other expression for manipulating the data within a multidimensional database. Formulae define relationships among members and are used by OLAP database builders to provide great richness of content to the server database. Additionally, formulae are used by end users to model enterprise relationships and to personalize the data for greater visualization and insight.

**formula, cross-dimensional.** A formula that references members from more than one dimension.

**function.** A user-defined set of mathematical or verbal criteria used to obtain highly specific information from a database. Examples of functions include “Average”, “Minimum”, and “Maximum”.

**generation.** In a database outline, the consolidation levels within each dimension. A series of generations begins at the root of an outline, then contains the branches, and ends at the leaf level. For example, in the time dimension, Year is generation 1, Quarter is generation 2 and Month is generation 3.

**global access layer.** A layer in the DB2 OLAP Server for AS/400 security system used to define settings for an application or database.

**global variable.** (1) A variable that is available to all queries. Global variables are used when the user wants the variable value to be available to all queries. (2) For example, a date range can be defined as a global variable so that it doesn't have to be re-keyed for multiple queries.

**group profile.** A user profile that provides the same authority to a group of users.

**group level control.** (1) This level of control allows the administrator to apply security and resource settings to groups of similar user profiles. (2) For example, configuring the SALES group to have Run-time Query Access Only is easier and quicker than applying the same setting to each individual sales person. Group level settings override public level settings, but are overridden by user level settings.

**header record.** In DB2 OLAP Server for AS/400, one or more records at the top of a data source that describe the contents.

**hierarchy.** In DB2 OLAP Server for AS/400, a set of multidimensional relationships created in an outline, usually in a “tree” formation. An example of a hierarchy is “parent, child, and generation.”

**hierarchical relationships.** Any dimension's members may be organized into parent-child relationships, typically where a parent member represents the consolidation of the members that are its children. The result is a hierarchy, and the parent/child relationships are hierarchical relationships.

**horizontal dimension.** See *page display*.

**hypercube.** See *array, multidimensional*.

**index.** (1) On the AS/400, indexes refer to pointers that are logically arranged by the values of a key. Indexes provide quick access and can enforce uniqueness on the rows in a table. (2) In DB2 OLAP Server for AS/400, “indexing” is a method of retrieving data through sparse dimensions.

**index cache.** A buffer in the memory for DB2 OLAP Server for AS/400 that holds index pages.

**index entry.** In DB2 OLAP Server for AS/400, an intersection of two thinly-populated, “sparse” dimensions.

**input block.** A type of data block that has at least one loaded data value.

**input data.** Data in DB2 OLAP Server for AS/400 that originates directly from the server. Input data is not derived from calculations performed on existing data.

**input member.** (1) Members whose values are loaded directly into a database, either through manual entry or by input from another computer-based data source. (2) **Antonym:** calculated member

**intelligent calculation.** A method that tracks which data blocks have been updated since the last calculation. Tracking changed data blocks makes calculation more efficient since it is not necessary to recalculate an entire database if only a limited number of data blocks are changed.

**isolation level.** A setting that specifies the degree to which your data is protected from

other transactions. For more information on isolation levels, see your IBM AS/400 documentation.

**JAR.** See *Java Archive*

**Java Archive.** A file format used to bundle all components required by a Java applet. JAR files simplify the downloading of applets since all the components (.class files, images, sounds, etc.) can be packaged into a single file. In addition, JAR supports data compression, which further decreases download times. By convention, JAR files end with a .jar extension.

**join.** To combine data from two tables into one set of data by specifying a conditional relationship between one or more columns from each table.

**join tables.** To combine data from two tables into one set of data by specifying a conditional relationship between one or more columns from each table.

**journal.** A recorded log that enables data to be recovered when a media failure has destroyed some or all of a database. The journal tracks changes to the database, enabling it to be rebuilt.

**leaf member.** (1) In a database outline, members that have no children. (2) **Synonyms:** detail members, level 0 members, leaf nodes

**leaf nodes.** See *leaf member*.

**level.** In a database outline, the branch within each dimension. Levels count from the leaf node up toward the root. For example, in the time dimension, Year is level 2, Quarter is level 1 and Month is level 0.

**level 0 member.** See *leaf member*.

**level name.** In DB2 OLAP Server for AS/400, a unique name given to a level.

**library.** An AS/400 system object that serves as a directory to other objects.

**list.** A set of values that is displayed to the user. The user selects from the list to respond to a prompt.

**linked partition.** A form of shared partition that provides the ability to link together two different databases with a data cell. When a user clicks on a linked cell in a spreadsheet, for example, drilling across to a second database and a new sheet opens, displaying the dimensions in the second database. The user can then drill down into the available dimensions in the second database.

**list box.** A box that appears within a dialog box. This box contains a list of information that you can select.

**load data.** The process of populating a DB2 OLAP Server for AS/400 database with data. Loading data establishes actual values for the cells defined by the database's structural outline.

**local.** Pertaining to a device accessed directly rather than through communications programs.

**local variable.** (1) A variable that is available only to the query for which it is defined. Local variables are used when the variable only relates to one query. (2) For example, a date range could be defined as a local variable if it is only used in a one query that tracks quarterly revenues.

**lock.** A method to prevent two people from simultaneously altering the same set of data. In DB2 OLAP Server for AS/400, you must lock data before you change it.

**log file.** A file that records actions and commands.

**macro.** A set of instructions that a program uses to automatically perform a task.

**MAX.** (1) A summary function used to group similar data and return the maximum grouped value in the column where the function was applied. (2) This function first groups the rows with identical data in those columns without any summary functions applied. Then one summarized row is returned for each group of rows which includes identical values and the

maximum value of the grouped values in the selected column. (3) When applied to a break group, this summary function returns the maximum value found within the break group.

**member.** A discrete component within a dimension. For example, January 1998 or First Quarter 1998 are typical members of a Time dimension.

**member combination.** See *cell*.

**metadata.** Data about data. For example, an alias name for a confusingly-named collection is a kind of metadata.

**MIN.** (1) A summary function used to group similar data and return only the minimum grouped value in the column where the function was applied. (2) This function first groups the rows with identical data in those columns without any summary functions applied. Then one summarized row is returned for each group of rows which includes identical values and the minimum value of the grouped values in the selected column. (3) When applied to a break group, this summary function returns the minimum value found within the break group.

**missing data.** A special data item which indicates that the data in a given cell does not exist. This may be because the member combination is not meaningful (e.g., a given company doesn't sell toboggans in Miami) or has never been entered. Missing data is similar to a null value or N/A, but is not the same as a zero value.

**multidimensional data structure.** See *array*, *multidimensional*.

**multidimensional.** Referencing data through three or more dimensions. An individual record is the intersection of a point for a set of dimensions.

**multidimensional query language.** A computer language that allows users to specify which data to retrieve out of a multidimensional database. The user process for this type of query is usually called slicing and dicing. The result of a

multidimensional query is either a cell, a two-dimensional slice, or a multi-dimensional sub-cube.

**name.** The identifier for a column or table that matches the data source identifier.

**navigation.** The processes employed by users to explore a multidimensional database interactively by drilling, rotating and screening, usually using a graphical OLAP client connected to an OLAP server.

**nesting.** A display technique used to show the results of a multidimensional query that returns a sub-cube, i.e., more than a two-dimensional slice or page. The column/row labels will display the extra dimensionality of the output by nesting the labels describing the members of each dimension.

**non-missing data.** Data which exists and has values, as opposed to null or missing data.

**normalize.** (1) The process of putting data in a physical location and structure that is optimized for speedy input and the storage of large amounts of data. A normalized database requires tables to be joined for most queries and includes very detailed data, but includes very little redundant data. (2) **Antonym:** denormalize

**object.** A file related to an application or database. Objects can be outlines, rules files, calculation scripts, report scripts, or data sources, and they are stored within the application or database subdirectory on the server or client machine.

**ODBC.** (1) Open Database Connectivity. The ODBC interface allows software applications to access data in different database management systems using Structured Query Language as a standard for accessing the data. (2) Because of this functionality, software developers can create one version of a software application that works with any ODBC-compliant database management system. The application is linked to the various database management systems through the use of database drivers. These drivers process the ODBC instructions from the application, submit

the resulting SQL requests to the data source and then return the query results to the application.

**OEM character set.** The character set as defined by a particular original equipment manufacturer (OEM).

**OLAP.** See *Online Analytical Processing*.

**OLAP client.** End user applications that can request information from OLAP servers and provide two-dimensional or multidimensional displays. OLAP clients typically allow users to modify, select, rank, and calculate information for visualization and navigation purposes. OLAP clients may be as simple as a spreadsheet program or as robust as a financial modeling or sales analysis application.

**OLAP database.** A database structure tuned for data analysis.

**OLAP server.** (1) A high-capacity, multi-user data manipulation engine specifically designed to support and operate on multidimensional data structures. A multidimensional structure is arranged so that every data item is located and accessed based on the intersection of the dimension members that define that item. The design of the server and the structure of the data are optimized for rapid ad-hoc information retrieval in any orientation, as well as for fast, flexible calculation and transformation of raw data based on formulaic relationships. (2) The OLAP Server may either physically stage the processed multidimensional information to deliver consistent and rapid response times to end users, or it may populate its data structures in real-time from relational or other databases, or offer a choice of both. Given the current state of technology and the end user requirement for consistent and rapid response times, staging the multidimensional data in the OLAP server is often the preferred method. It is also the method employed by DB2 OLAP Server for AS/400.

**OLTP.** See *Online Transaction Processing*.

**online analytical processing.** 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. Response time for any information request is from 1-3 seconds because data is stored in summary fashion and is indexed for speedy retrieval.

**online transaction processing.** A computing environment optimized for storage of large amounts of information and fast input of data. OLTP databases are generally normalized and are poorly designed for analyzing or retrieving information. OLTP is typically used to store data created through standard business processes such as order tracking, general ledger, or human resources.

**operator.** A type of expression that can be used with arithmetic expressions (+, -, \*, /, %), comparison expressions (=, >, <, etc.) or logical expressions (AND, OR, NOT) to manipulate data and return other expressions.

**outline.** The structure that defines all elements of a database within DB2 OLAP Server for AS/400. It contains definitions of dimensions and members, dense or sparse dimension tags and attributes, calculations, shared members, and alterations to the basic roll-up structure of the database.

**owners.** The second-level of a SQL Server table reference. For example, in the table reference "cms.dbo.customer", "cms" is the database name, "dbo" is the owner name, and "customer" is the table name.

**page dimension.** A dimension that is not one of the two dimensions of the view being displayed, but for which a member has been selected to define the specific view requested for display. All page dimensions must have a specific member chosen in order to define the appropriate view for display.

**page display.** The current orientation for viewing a multidimensional slice. The horizontal dimension(s) run across the display, defining the column dimension(s). The vertical dimension(s) run down the display, defining the contents of



the row dimension(s). The page dimension-member selections define which page is currently displayed. A page is much like a spreadsheet, and may in fact have been delivered to a spreadsheet product where each cell can be further modified by the user.

**paging.** A data storage method that makes use of spare disk space by increasing the available memory. Also referred to as “swapping” or “virtual memory.”

**parent.** In a database outline, any member that is hierarchically above another member. For example, Quarter 1 is a parent of January.

**partitioning.** Partitioning is the process of distributing a file across the nodes in a node group. Partitioning is done using the hash algorithm. When a new record is added, the hash algorithm is applied to the data in the partitioning key. The result of the hash algorithm, a number between 0 and 1023, is then applied to the partitioning map to determine the node on which the record will reside. Partitioning can affect the performance and scalability of applications.

**pivot.** The ability to alter the perspective of data in a viewer. When DB2 OLAP Server for AS/400 first retrieves a dimension, it expands into rows. A user can pivot or rearrange the data to obtain a different viewpoint.

**pre-calculated data.** (1) Data in output member cells that are computed prior to, and in anticipation of, ad-hoc requests. Pre-calculation usually results in faster response to queries at the expense of storage space. Data that is not pre-calculated must be calculated at query time. (2) **Antonym:** derived data

**precision.** The total number of significant digits in a numeric value.

**pre-consolidated data.** See *pre-calculated data*.

**prompt.** When a query is run, a *prompt* asks the user to enter a value that will narrow the amount of data returned from the data source. For example, when a user runs a query, he might

be prompted to enter the state or fiscal year for which data will be returned.

**prompt from a file.** (1) When a query is run, a *prompt* asks the user to enter a value that will narrow the amount of data returned from the data source. A *prompt from a file* requires that the value entered match a value defined in a personal computer file. (2) For example, a user who queries a database for sales figures can be prompted to enter the item number for which sales figures will be returned. You could build a query to fetch those item numbers and save them to a file, such as a Microsoft Excel spreadsheet. The user would then select a prompt answer from that file containing the item numbers.

**prompt from a user-defined list.** (1) When a query is run, a *prompt* asks the user to enter a value that will narrow the amount of data returned from the data source. A *prompt from a user-defined list* requires the user to choose from a list of possible answers defined within the query. (2) For example, you might choose this option if you are prompting users to enter the name of a division for which they wish to query sales data. You would then define a list of division names from which users would select.

**public level control.** (1) This level of control allows the administrator to apply default security and resource settings to profiles that do not have specific user or group level settings defined. (2) Public level control (designated by the profile \*PUBLIC) provides an easy way for the administrator to specify settings common to all user and group profiles on the AS/400. \*PUBLIC settings are overridden by either user or group level settings. \*PUBLIC settings override the Exclude Users from Libraries and Tables by Default security options.

**query.** A statement that extracts information from the data source. Query stores a query with instructions for handling the data returned from the data source in a file with the extension DBQ.

**query reference data.** (1) Information, including result data, that is returned by a query to an Excel or 1-2-3 workbook. (2) Examples include

column headings, variables, and the SELECT statement used to execute the query.

**reach through.** A means of extending the data accessible to the end user beyond that which is stored in the OLAP server. A reach through is performed when the OLAP server recognizes that it needs additional data and automatically queries and retrieves the data from a data warehouse or OLTP system.

**relational database.** A data structure that appears to the user as a collection of tables.

**reserved words.** A special word that has a specific meaning to the system as defined in a programming language. For example, in OS/400, words such as FORM, QUERY, COUNT, and NULL are reserved words. For a complete list of reserved words, see your IBM documentation.

**restore.** An operation that reloads data and/or structural information after a database has been damaged or destroyed. A restore operation is typically executed after the database has been shut down and restarted.

**result column.** A set of column values generated by an expression that contains one or more columns.

**returned data.** The set of values that Query gathers from the data source in response to a query.

**roll-up.** See *consolidate*.

**root.** In a database outline, the top-most member in a branch.

**rotate.** (1) To change the dimensional orientation of a report or page display. For example, rotating may consist of swapping the rows and columns, or moving one of the row dimensions into the column dimension, or swapping an off-spreadsheet dimension with one of the dimensions in the page display (either to become one of the new rows or columns), etc. A specific example of the first case would be taking a report that has Time across (the columns) and

Products down (the rows) and rotating it into a report that has Product across and Time down. (2) **Synonym:** pivot

**router.** In networks, a router serves as a link enabling messages and other information to be sent between systems. For DB2 OLAP Server for AS/400, a router provides the link between the software on the personal computer and the software on the AS/400.

**row dimension.** See *page display*.

**row.** A set of values in a table, usually displayed horizontally, corresponding to one value per column. Also called a record.

**scale.** The number of digits after the decimal point of a decimal value.

**scoping.** Restricting the view of database objects to a specified subset. Further operations, such as update or retrieve, will affect only the cells in the specified subset. For example, scoping allows users to retrieve or update only the sales data values for the first quarter in the east region, if that is the only data they wish to receive.

**screen.** See *selection*.

**search condition.** A defined test applied to the values of a column that limits the data returned from the database.

**SELECT statement.** A Structured Query Language (SQL) statement that requests data from a data source.

**selection.** (1) A process whereby a criterion is evaluated against the data or members of a dimension in order to restrict the set of data retrieved. Examples of selections include the top ten salespersons by revenue, data only from the east region, and all products with margins greater than 20 percent. (2) **Synonyms:** condition, screen, filter

**server (multidimensional).** A high-capacity, multi-user database server that locates and accesses data values based on their intersection of dimension members.

**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 the outline.

**sibling.** In a database outline, any member that is at the same branch level as another member. For example, Quarter 1, Quarter 2, and Quarter 3 are siblings.

**significant digit.** Numbers to the right of a decimal point.

**single value prompt.** (1) When a query is run, a *prompt* asks the user to enter a value that will narrow the amount of data returned from the data source. A *single value prompt* allows the user to enter any value desired. (2) For example, a user who queries a database for sales figures can be prompted to enter the name of a sales representative for whom sales figures will be returned. The user is allowed to enter any name desired.

**slice.** A subset of a multidimensional array corresponding to a single value for one or more members of the dimensions not in the subset. For example, if the member Actuals is selected from the Scenario dimension, then the sub-cube of all the remaining dimensions is the slice that is specified. The data omitted from this slice would be any data associated with the non-selected members of the Scenario dimension, for example Budget, Variance, Forecast, etc. From an end user perspective, the term slice most often refers to a two-dimensional page selected from the cube.

**slice and dice.** The user-initiated process of navigating by calling for page displays interactively, through the specification of slices via rotations and drill down/up.

**SMP.** See *Symmetric Multiprocessing*.

**sparse dimension.** (1) Describes a dimension in a multidimensional database with a low probability for occupying one or more data point in every combination of dimensions that occurs. For example, in a typical multidimensional

database, the products dimension is often sparse since not every product is sold in every market. (2) **Antonym:** dense

**spreadsheet.** An application program that can display the data returned to Query. The spreadsheet enables you to create charts and to format or print the data. See also *viewer*.

**SQL (Structured Query Language).** A database language used to manipulate information in a database.

**stop and go lights.** See *traffic lighting*.

**Storage Manager.** A layer of the server component of DB2 OLAP Server for AS/400 that handles the mapping of OLAP data to the appropriate location in the server memory. The Storage Manager controls lock, index, data, allocation, and transaction management functions.

**SUM.** (1) A summary function used to group similar data and add the grouped values in the column where the function was applied. (2) The function first groups rows with identical data in columns without any summary functions applied. Then one summarized row is returned for each group of rows which includes identical values and the sum of the grouped values in the selected column.

**summary line.** A row that is returned within a break group that is a result of a summary function applied to the break group data.

**summary text.** The descriptive text that appears to the left of the summary line data within a break group.

**synchronously.** Regarding queries, synchronously refers to a situation in which multiple queries run sequentially and in a predetermined order. Each query will not begin running until the preceding query has finished. This is useful when one query depends on data returned from another query to run.

**Symmetric Multiprocessing.** A computer architecture that provides fast performance by

making multiple CPUs available to complete individual processes simultaneously (multiprocessing).

**system administrator.** A person who maintains the hardware, software, disk space, distributions, and configurations for running DB2 OLAP Server for AS/400.

**table.** A collection of columns and rows of data that resides in the data source. Also called a file.

**table designator.** (1) A qualifier that designates a specific table. The clause that identifies the object tables also establishes the table designators for them. For example, the object tables of an expression in a SELECT clause are named in the FROM clause that follows it: (2) SELECT CUSTID, CUSTNAME FROM SCSAMPLE30.CUSTOMERS. CUSTOMERS (3) In this example, SCSAMPLE30.CUSTOMERS is the table designator for both CUSTID and CUSTNAME. (4) A table designator could also be an alias.

**TCP/IP (Transmission Control Protocol/Internet Protocol).** A standard set of communications protocols adopted by many companies and institutions throughout the world. Allows computers to communicate despite differences in computer types and operating systems.

**template.** A pre-defined screen, designed to retrieve particular data on a regular basis in a consistent format.

**time dimension.** A dimension type that defines how often you collect and update data. You may only tag one dimension as Time, although you do not need to have a Time dimension. Year, Quarter and Month are examples of members typically found in the time dimension.

**timer.** A setting that is used to automatically run a query.

**traffic lighting.** A reporting mechanism that typically highlights good data as green, marginal data as yellow, and problem data as red. In most software packages that support traffic lighting, the user is able to define the numeric ranges that determine the color of a given value.

**two-pass calculation.** An DB2 OLAP Server for AS/400 attribute used for re-calculating an accounts member after a consolidation takes place.

**unary operator.** A group of mathematical indicators (+,-,\*,/) that define how roll-ups are executed in the OLAP database outline.

**unmatched row.** (1) A row of data that does not contain enough information to meet specified join conditions. As a result, the row is blank. (2) For example, if a Customer Name table is joined to an Invoice table, an unmatched row will result for any customers for whom invoice data does not exist. When creating the join, the user can request that such rows either be eliminated from the subsequent query or be returned empty.

**user event.** A macro or Visual Basic function that is executed either before or after a query or series of queries are refreshed.

**user function.** Macros or Visual Basic programs that are callable by the user.

**user level control.** (1) This level of control allows the administrator to apply security and resource settings individually to user profiles. (2) Each user profile on the AS/400 can have its own settings. User level is the highest level of control and overrides any settings specified at the group or public level.

**user profile.** An object with a unique name that contains the user's password, the list of special authorities assigned to a user, and the objects the user owns.

**variable.** (1) A value in the definition of a query that you can change to return different sets of data when Query runs the query. You can change the value with a prompt, in the Set Value dialog box, or with a dynamic data exchange command. (2) In Query, you can specify a variable for a single query or a variable that is shared between multiple queries.

**vertical dimension.** See *page display*.

**view.** A representation of table data that can contain columns from one or more tables.

**viewer.** The vehicle that Query uses to display the data returned from the data source. A viewer can be a spreadsheet, a word processor, or the Query built-in display function.

**virtual data.** Data that DB2 OLAP Server for AS/400 does not calculate until the user reuses the report or view that the data is contained in.

**wizard.** A piece of software which asks a user to enter various pieces of information and then automatically performs some task based on the user's input.

**workbook.** A collection of one or more worksheets. A workbook is associated with an actual file, usually a .XLS (Microsoft Excel) or .WK3 (Lotus 1-2-3) file.

**workgroup folders.** An AS/400-based storage container that can contain Builder Manager Metadata.

**worksheet.** A single page in a spreadsheet file, usually named sheet1, sheet2, etc.

**workstation.** A personal computer that is connected to a network server, mainframe, or other computer at which a user can use and run applications.

**zoom.** (1) The process of retrieving progressively detailed data relative to a selected dimension. Zooming in on a database dimension provides you with greater detail on that dimension, while zooming out moves your perspective to a higher consolidation level. (2) **Synonym:** drill down/up



---

# Index

## Numerics

5250 display sessions 54

## A

ALLOBJ Authority 72

Analyzer OLAP Server 26

APPC 1, 21, 35, 39

with prestart jobs 38

without prestart jobs 37

AS/400

command reference 83

installing DB2 OLAP Server for

AS/400 vi

managing information 13

work management 33

AS/400 commands

reference 83

security issues 40

ASCII CCSID 50

assigning

levels of parallelism 3

authority

required for installation 1

automatic deletion of OLAP

users 80

## B

backing up

a Warehouse Manager library 25

data 13, 15

OLAP files 81

server information 26

batch queries 36, 39

building OLAP dimensions 59

## C

CCSID 60

mixed and single byte

equivalents 60

changed records 7

changing

data views with dependent data

views 20

physical files with dependent

data views 20

checking current usage of a

Warehouse Manager library 22

CHGPF command 28

CHGSRVRSTS command 31

choosing the most efficient type of

distributions 6

CLRPFM 7

commands

security issues 40

configuration after restoring

server 19

configuration file

OLAP 45

configuring

OLAP environment

parameters 46

OLAP server information 45

TCP/IP 21

the Essbase environment 46

Warehouse Manager TCP/IP

support 21

controlling

level of parallelism 3

create custom library lists for

users 15

CUSTOMERS table 91

## D

data

backing up 15

disaster recovery backups 17

loading into OLAP 69, 71

reloading rejected records 72

temporary backups 15

data load

error file considerations 59

errors 59

rejected records 71

data views

changing dependent data

views 20

changing physical files 20

database

sample 89

SCSample 89

DATETYPES table 96

DB2/400

loading data into OLAP 69

DB2 OLAP Server for AS/400

migrating to a new library 22

security issues 40

using TCP/IP with 21

DB2 Symmetric Multiprocessing 6

debugging 67

default wait time 34

defining the uses of an OLAP

application 11

deleting target tables upon

distribution 7

detecting and removing obsolete

information 30

determining level of parallelism 3

dimension build

considerations 59

errors 59

rejected records 71

directing data 14

disaster recovery backups 17

displaying

OLAP environment

parameters 48

QSYS.LIB files from a 5250

display session 54

Distributed Relational Database

Architecture 42

distributions

choosing the most efficient 6

DRDA 42

## E

EBCDIC CCSID 49, 60

editing QSYS.LIB files from a 5250

display session 55

EMPLOYEES table 98

enabling expert cache 5

ending

ShowCase prestarted jobs 23

ShowCase TCP/IP 23

Warehouse Manager 1

Warehouse Manager TCP/IP or

prestarted jobs 23

Enforce AS/400 Security option 31

error logging 59

ESSBASE.CFG file 45

ESSCMD/400 55

error files 58

output files 58

script file differences 58

using in batch mode 57

using interactively 56

ESSCMD API Toolkit 62

expert cache

enabling 5

- F**
  - files 52
  - finding
    - available port numbers 22
    - files within the integrated file system (IFS) 53
  - flat file support 70
- G**
  - getting help on ESSCMD
    - commands 62
  - glossary 103
- I**
  - IFS
    - finding files in 53
    - syntax 53
  - index
    - cache size 68
  - installing
    - DB2 OLAP Server for AS/400 vi
  - Integrated Security 73
- J**
  - job
    - class 34
    - description 39
    - run priority 34
  - journals
    - working with 33
- L**
  - library
    - backing up a server 25
    - checking current usage 22
    - migrating to a new 22
    - releasing the restriction on a server 32
    - restoring a server 28
    - restricting access to a server 31
  - library lists
    - creating for users 15
  - loading
    - data into OLAP 59, 69, 71
    - DB2/400 data into OLAP 69
  - log files 68
  - logging errors 59
  - LOGINESS 63
  - LOGOUTESS 63
- M**
  - maintaining OLAP log files 68
  - managing
    - information on the AS/400 13
  - maximum temporary storage 34
  - message logging 81
- migrating
  - data from test to production 13
  - server information 27
  - server information for backup
    - purposes 15
  - to a new library 22
  - users to a new Warehouse Manager library 22
- miscellaneous objects 52
- mixed CCSID equivalents 60
- moving
  - data from test to production 13
  - server information 27
  - server information for backup
    - purposes 15
- multiprocessing 6
- N**
  - National Language Support
    - OLAP 48
- O**
  - obsolete information
    - removing 30
  - OLAP
    - AS/400 Performance Tuning 10
    - ASCII CCSID 50
    - backing up files 81
    - Calculation Process 10
    - command reference 85
    - configuration and environment relationship 48
    - configuration file 45
    - configuring the Essbase environment 46
    - data load errors 59
    - debugging 67
    - defining uses 11
    - dimension build errors 59
    - EBCDIC CCSID 49
    - ESSCMD/400 55
    - index cache size 68
    - input script files 58
    - installation library and path 52
    - Integrated Security 73
    - jobs 81
    - loading data 69
    - loading databases from Warehouse Builder 71
    - log files 68
    - maintenance 45
    - message logging 81
    - National Language Support 48
    - optimizing 9
    - performance 9
    - QSYS.LIB objects 52, 53, 54
- OLAP (*continued*)
  - QSYS.LIB objects created during installation 52
  - reloading rejected records 72
  - removing Essbase environment
    - parameters 48
  - restoring files 82
  - script files 55, 58
  - security 72
  - special considerations 45
  - SQL Data Load Plan 9
- OLAP server
  - changing the ASCII and EBCDIC CCSIDs 50
  - commands 85
  - configuring server
    - information 45
  - displaying the current ASCII CCSID 50
  - displaying the current EBCDIC CCSID 49
  - starting 1
- OLAP server component 1
- optimizing
  - Warehouse Builder performance 6
- OUTPUT command 61
- P**
  - parallelism 3
    - assigning levels 3
  - performance
    - system tuning 5
  - pre-creating libraries 7
  - prestart jobs 37
  - prestarted jobs
    - ending 23
    - removing 24
  - production environment
    - moving from a test environment 13
  - PRODUCTS table 93
  - pull distributions 7
- Q**
  - QSYS.LIB objects 52, 53, 54
- R**
  - re-directing data 14
  - rejected records 59
    - reloading 72
  - releasing a Warehouse Manager library from a restricted state 32
  - reloading rejected records 72



- removing
  - a Warehouse Manager library 24
  - obsolete information 30
  - OLAP environment
    - parameters 48
  - ShowCase prestarted jobs 24
  - ShowCase TCP/IP 24
  - Warehouse Manager server
    - libraries 24
  - Warehouse Manager TCP/IP or prestarted jobs 24
- renaming users and groups 81
- restoring
  - a Warehouse Manager library 28
  - OLAP files 82
  - server information 28
  - the stored procedure
    - declarations 28
- restoring server
  - followup configuration 19
- restricting access to a Warehouse Manager library 31
- routing entries and prestart job entries 37
- RSTSRVRINF command 26
- RTVESSSTS 63
- RUNESSCMD 63
- S**
  - SALESORG table 94
  - SALESPERF table 97
  - SALESREPS table 95
  - save and restore 13
  - Save and Restore feature 27
  - saving
    - server information 26
    - the user-defined stored procedure
      - declarations 26
  - SAVSRVRINF command 26
  - scenarios
    - test and production on separate AS/400s 14
    - test and production on the same AS/400 15
  - script files 55, 58
  - SCSample 89
  - security 72
    - AS/400 command authorities 40
    - issues 40
    - OLAP 72
  - sending only change records during distributions 7
  - sequence numbers, using 8
  - server information
    - backing up (saving) 26
    - restoring 28
  - servers
    - backing up 15
    - ending 1
    - releasing the restriction on 32
    - restricting access to 31
    - starting 1
  - setting
    - OLAP Integrated Security 73
  - specifying
    - IFS syntax for QSYS.LIB
      - objects 53
    - IFS syntax for root file system
      - objects 53
    - output and error files within
      - ESSCMD/400 script files 58
  - SQL
    - drill-through using Essbase
      - add-ins 81
    - loading data into OLAP 69
    - starting
      - OLAP server component 1
      - Warehouse Manager 1
    - starting and stopping the servers 1
    - stopping
      - Warehouse Manager 1
    - storage pools 34
    - stored procedures
      - restoring 28
      - saving 26
    - substitution variables in rule file
      - SQL statements 70
    - subsystems 35
    - system performance tuning 5
  - T**
    - target tables
      - deleting upon distribution 7
    - TCP/IP 21, 35, 38
      - configuring 21
      - ending 23
      - port numbers 22
      - removing 24
    - TCP/IP and APPC with prestart jobs 39
    - temporary backups 15
    - test environment
      - moving to production 13
    - The Outdoor Connection 89
    - TIMEDIM table 97
    - tuning system performance 5
    - types of distributions 6
  - U**
    - understanding
      - configuration and environment
        - relationship 48
    - understanding (*continued*)
      - OLAP file system usage 51
      - OLAP server configuration 45
    - user
      - exit programs 40
      - profiles 39
    - user exit programs 40
    - using
      - an ESSCMD/400 input script
        - file 58
      - an ESSCMD/400 output or error
        - file specified from within
          - ESSCMD 61
      - CURDATE () 8
      - ESSCMD/400 55
      - ESSCMD/400 in batch mode 57
      - ESSCMD/400 in interactive
        - mode 56
      - ESSCMD API Toolkit 62
      - OLAP configuration file
        - settings 45
      - OLAP environment settings 46
      - OLAP national language support
        - (NLS) 48
      - QSYS.LIB files from a 5250
        - display session 54
      - sequence numbers 8
      - SQL functions to control
        - distributions 8
      - TCP/IP with DB2 OLAP Server
        - for AS/400 21
      - Warehouse Builder with
        - OLAP 71
  - W**
    - Warehouse Builder
      - improving performance 6
      - naming the OLAP server 72
      - tuning issues 9
      - using to load Essbase
        - databases 71
    - Warehouse Manager 1
      - backing up a server library 25
      - checking current usage 22
      - command reference 84
      - commands 84
      - ending prestarted jobs 23
      - ending TCP/IP 23
      - ending the server 1
      - releasing the restriction on a
        - server library 32
      - removing obsolete
        - information 30
      - removing prestarted jobs 24
      - removing TCP/IP 24

- Warehouse Manager 1 *(continued)*
  - restoring a server library 28
  - restricting access to a server library 31
  - security issues 40
  - starting the server 1
  - work management 33
  - working with journals 33
- Warehouse Manager and AS/400
  - work management 33
- Warehouse Manager server
  - starting 1
- what about passwords? 80
- work management 33
- work management controls 33
  - job class 34
  - job description 39
  - prestart jobs 36, 37
  - routing entries 37
  - storage pool 34
  - subsystems 35
  - user profile 39
- working with journals 33





Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC26-9677-00

