EDMSuite OnDemand

Administrator's Reference

Version 2.2

S544-5293-08

EDMSuite OnDemand

Administrator's Reference

Version 2.2

S544-5293-08

Ninth Edition (June 1999)

This edition of *IBM EDMSuite OnDemand: Administrator's Reference* applies to IBM EDMSuite OnDemand, Version 2 Release 2 and to all subsequent releases of this product until otherwise indicated in new releases or technical newsletters.

Requests for copies of this publication and for technical information about IBM products should be made to your IBM authorized Dealer, your IBM Marketing Representative, or your IBM Software Solutions Representative.

IBM Software Solutions welcomes your comments. For your convenience, a form for reader's comments is provided at the back of this publication. You may send your comments by fax to 1-303-924-7522, by e-mail to Ondemand@us.ibm.com, or mail your comments to:

INFORMATION DEVELOPMENT
IBM SOFTWARE SOLUTIONS
DEPARTMENT MYGA BUILDING 001L
PO BOX 1900
BOULDER CO 80301-9191

Note: Visit our home page at http://www.software.ibm.com/data/edmsuite.

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

Contents

Notices . . . . . **ix**
Trademarks and Service Marks ix

About this publication . . . . . **xiii**
Who should use this publication xiii
How this publication is organized xiii
Product support xiv
Our use of typefaces xiv
OnDemand documentation xiv
Related documentation. xv

Related IBM EDMSuite Products . . . . **xvii**
ImagePlus VisualInfo xvii
ContentConnect xvii
MQSeries Workflow (formerly FlowMark) xviii
Domino Doc xviii

Part 1. Loading and Maintaining Data **1**

Chapter 1. Loading reports **3**
Overview 3
Preparing to load reports 4
Estimating and allocating storage space 4
Defining the application group 5
Defining the application 5
Accessing reports, index data, and resources 5
Configuring arsload. 7
Indexing reports 9
Indexing reports on OnDemand servers 10
Indexing reports on MVS and OS/390 systems 11
Processing index data, reports, and resources 12
Processing index data 12
Processing reports and resources . . . 13
Loading index data 13
Loading storage objects 14
Cache storage 14
Primary storage node 14
Secondary storage node 15
Resources 15
Backing up databases 15
Backup the OnDemand database . . . 16
Backup the ADSM database 16
Protecting the document cache 17
Checking messages and verifying totals 17

Chapter 2. Loading image files **19**
Overview 19
Defining the application group 19
Database organization 20
Expiration type 20
Permissions 20
Field definition 20
Defining the application 21
Application group 21
Data format 21
Indexer 21
Data compression 21
Defining the folder 21
Application group 21
Permissions 22
Folder fields 22
Acquiring the images 22
Accessing the image files 23
Creating index data 23
Configuring arsload. 24
Processing the index data and the image files 25
Processing index data 26
Processing the image files. 26
Verifying processing. 26

Chapter 3. Loading user-defined data **29**
Overview 29
About the example 29
Defining the application group 30
Database organization 30
Expiration type 30
Permissions 30
Field definition 30
Defining the application 31
Application Group 31
Data Format 31
File Extension. 31
Indexer 32
Data compression 32

Defining the folder 32
Application group 32
Permissions 32
Folder fields 32
Field Mapping 33
Accessing the input files 33
Creating the index data 33
Configuring arsload 34
Processing the input data 35
Processing index data 35
Processing the input files 36
Verifying processing. 36

Chapter 4. Restarting a load process 39

Chapter 5. Deleting a report 41

Chapter 6. Download exits and processing 43
Overview 43
Using Download 43
Invoking a data download exit program 44

Chapter 7. Maintaining migrated index data 45
Configuring the system 45
Logging and routing messages 45
Defining a system log user exit 46
Configuring ADSM 46
Defining storage sets 46
Configuring application groups 46
What happens when a user queries migrated data 47
Message to the user 47
Message to the system log 47
Message to the system 48
Message to the system log user exit . . 48
Importing the index data 48
Verify temporary work space. 48
Verify database storage space 48
Verify database log file space. 48
Run the arsadmin command 49
After you import index data 49
Querying data. 49
Expiring imported migrated indexes 49
Configuring index migration 50

Chapter 8. Backup and recovery 51
Overview 51
Server software 51
Server information 51
The OnDemand database 52
DB2 UDB table spaces 52
Database backup 53
Creating a full offline backup of the database 54
Database logging. 55
Database recovery 56
The ADSM database 58
ADSM recovery log 59
ADSM volume history file 60
ADSM device configuration file 60
ADSM database recovery 61
Archived reports and resources 62
The document cache 62
Reports on archive media 63

Part 2. Managing ADSM storage 65

Chapter 9. Overview 67
Using magnetic disk devices 67
Using removable media devices. 68
Managing removable media operations 68
Defining drives and libraries 69
Defining device classes. 69
Managing storage pools 70
Managing storage pool volumes. 70
Managing policies 71
Managing client nodes 71
Automating server operations 71
Managing server operations 72
Managing the database and recovery log 72
Monitoring the server 72
Protecting and recovering data 73

Chapter 10. Starting, halting, and restarting the server 75
Starting the server 75
Starting the ADSM server command line interface. 75
Halting the server 76
Restarting the server 76

Chapter 11. Using scratch and private volumes 77

Chapter 12. Labeling storage volumes 79
Overwriting volume labels 79
Identifying volume labels 79

Labeling volumes one at a time 80
Searching the library 80

Chapter 13. Adding storage volumes . . 81
Checking in volumes one at a time. . . . 81
Searching the library 82
Allowing swapping of volumes 82

Chapter 14. Removing storage volumes 83
Removing volumes from a library 83
Returning volumes to a library 84
Managing a full library. 84
Offline storage of storage volumes 85

Part 3. Operating UNIX Servers 87

Chapter 15. Starting and stopping processes 89
Starting OnDemand. 89
Starting the database 89
Starting ADSM 89
Starting Download 90
Starting related MVS and OS/390 programs 90
Starting the server program 90
Starting arsload 91
Stopping OnDemand 91
Stopping the OnDemand server program 93
Stopping ADSM 93
Stopping DB2 UDB 94
Restarting OnDemand 94

Chapter 16. Monitoring the server . . . 95
Errors and alerts 95
OnDemand system logging facility. . . . 96
Searching for and viewing messages 96
Monitoring users. 97
Reviewing a CD-ROM creation manifest 97
System log user exit. 98
Monitoring processes 98
Displaying OnDemand process information 98
Displaying DB2 UDB processes 100
Displaying ADSM processes 101
Monitoring system load 101
Monitoring memory usage 102
Monitoring system paging space 102
Monitoring disks and file systems 103
Monitoring I/O activity 103
Monitoring storage volumes 104
Monitoring the network 105

Part 4. Operating Windows NT Servers 107

Chapter 17. Administrative tools and procedures 109
Administering servers with OnDemand tools 109
OnDemand configurator 109
Administrative client 109
Administering databases with DB2 tools 110
Administering SQL Server 110
Administering ADSM 111
Administering Windows NT 111

Chapter 18. Readying the system for use 113
Install and configure the system. 113
Backup databases and other information 113
Control access to the system 114
Creating users and groups 114
Assigning permissions 114
Software licensing 115
OnDemand 115
ADSM 115
DB2 UDB 115
Windows NT Server. 116

Chapter 19. Managing servers 117
Starting and stopping services 117
When do you start and stop services? 118
Monitoring servers 119
Errors and alerts 119
System logging facility 119
Monitoring performance 121
Database administration 122
Backing up the database 123
Optimizing the database 124
Monitoring DB2 database file systems 126
Managing DB2 log files 126
Monitoring the SQL Server database and transaction log 127
Database migration 128
Configuring and managing cache storage 129
Starting the migration utility 130
Scheduling the migration utility. 130

Part 5. Command Reference . . . 133

Chapter 20. Overview 135
How to read a syntax diagram 135

Chapter 21. arsadm command reference 139
Purpose 139
Syntax 139
arsadm USER command function . . . 139
arsadm GROUP command function . . 141
arsadm PRINTER command function 141
Description. 142
Parameters 144
Examples 149
Notes 151
Files 151

Chapter 22. arsadmin command reference 153
Purpose 153
Syntax 153
Description. 156
Parameters 159
Examples 163
Files 164

Chapter 23. ars_adsm command reference 165
Purpose 165
Syntax 165
Description. 165
Parameters 165
Examples 166
Notes 166
Files 166

Chapter 24. arsdate command reference 169
Purpose 169
Syntax 169
Description. 169
Parameters 169
Examples 171
Files 172

Chapter 25. arsdb command reference 173
Purpose 173
Syntax 173
Description. 173
Parameters 174
Examples 178
Notes 179
Files 179

Chapter 26. arsdoc command reference 181
Purpose 181
Syntax 181
arsdoc ADD command function 181
arsdoc DELETE command function . . 182
arsdoc GET command function 182
arsdoc PRINT command function . . . 183
arsdoc QUERY command function . . . 184
arsdoc UPDATE command function . . 185
Description. 185
Parameters 187
Examples 196
Notes 199
Files 199

Chapter 27. arsjesd command reference 201
Purpose 201
Syntax 201
Description. 201
Parameters 202
Examples 203
Notes 203
Files 204

Chapter 28. arsload command reference 205
Purpose 205
Syntax 205
Description. 205
Parameters 206
Examples 211
Notes 212
Files 212

Chapter 29. arsmaint command reference 213
Purpose 213
Syntax 213
Description. 213
Parameters 214
Examples 218
Notes 218
Files 218

Chapter 30. arstblsp command reference 219
Purpose 219
Syntax 219
Description. 219
Parameters 220
Examples 221
Notes 221
Files 222

Chapter 31. arsview command reference 223
Purpose 223
Syntax 223
Description 223
Parameters 224
Examples 225
Files 225

Part 6. Appendixes 227

Index 229

Readers' Comments — We'd Like to Hear from You 233

Notices

References in this publication to products or services of IBM do not suggest or imply that IBM will make them available in all countries where IBM does business or that only products or services of IBM may be used. Noninfringing equivalents may be substituted, but the user must verify that such substitutes, unless expressly designated by IBM, work correctly. No license, expressed or implied, to patents or copyrights of IBM is granted by furnishing this document. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594, USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (1) the exchange of information between independently created programs and other programs (including this one) and (2) the mutual use of the information, which has been exchanged, should contact: SWS General Legal Counsel, IBM Corporation, Department TL3 Building 062, P.O. Box 12195, Research Triangle Park, NC 27709–2195 USA. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks and Service Marks

The following terms, used in this publication and other books in the OnDemand library, are trademarks, registered trademarks, or service marks of the IBM Corporation in the United States and other countries:

ADSTAR
Advanced Function Printing
AFP
AIX
AIX/6000
AIXwindows
BookManager
BookMaster
CICS
CICS/ESA
DATABASE 2
DB2
DB2 Universal Database

eNetwork

IBM

Infoprint

IPDS

MVS

MVS/XA

MVS/ESA

OS/2

OS/390

Print Services Facility

RS/6000

S/370

S/390

SP2

SP

System/370

System/390

Ultrastar

VisualInfo

WIN-OS2

The following terms, used in this publication and other books in the OnDemand library, are trademarks of other companies as listed:

Acrobat, Acrobat Exchange, Adobe, Adobe Type Manager, ATM, Distiller, and PostScript are trademarks or registered trademarks of Adobe Systems Incorporated

Cygnet is a registered trademark of Cygnet Systems, Inc.

Ethernet is a trademark of Xerox Corporation

Exabyte is a trademark of Exabyte Corporation

FaxBox is a trademark of DCE Corporation

Hewlett-Packard, HP-UX, HP VUE, and PCL are trademarks of Hewlett-Packard Company

Intel, MMX, and Pentium are trademarks or registered trademarks of Intel Corporation

Lotus is a trademark of Lotus Development Corporation

Microsoft, TrueType, Windows, Windows 95, and Windows NT are registered trademarks of Microsoft Corporation

Motif is a trademark of Open Software Foundation, Inc.

Netscape and Netscape Navigator are trademarks of Netscape Communications Corporation

NFS is a trademark of SUN Microsystems Incorporated

Novell is a trademark of Novell Inc.

PC/TCP is a trademark of FTP Software, Inc.

Post-it is a registered trademark of 3M

Solaris is a trademark of Sun Microsystems, Inc.

Tivoli is a trademark of Tivoli Systems, Inc.

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

WinFax Lite is a trademark of Delrina Corporation

X Windows is a trademark of the Massachusetts Institute of Technology

Portions of the OnDemand Windows client program contain licensed software from Adobe Systems Incorporated, © Adobe Systems Incorporated 1987-1997. All rights reserved.

Portions of the OnDemand Windows client program contain licensed software from Pixel Translations Incorporated, © Pixel Translations Incorporated 1990, 1997. All rights reserved.

About this publication

This book contains information that can help you operate and administer an EDMSuite OnDemand Version 2.2 (OnDemand) system. It provides an overview of the tasks an administrator is expected to perform and describes the tools available to assist the administrator. OnDemand requires a database manager product, such as IBM's DB2 UDB Universal Database (DB2). If you plan to maintain reports on archive media, OnDemand requires the IBM ADSTAR Distributed Storage Manager (ADSM) product. If you plan to use the OnDemand server print facility, you must install Print Services Facility for AIX (PSF for AIX) on a workstation connected to the OnDemand network. If you plan to transmit files from an MVS or OS/390 system to an OnDemand server, we recommend that you use MVS Download or Download for OS/390.

Note: PSF for AIX, PSF/MVS, and PSF for OS/390 are commonly referred to as PSF throughout this book. MVS Download and Download for OS/390 are commonly referred to as Download throughout this book.

Who should use this publication

This reference is of primary interest to administrators of OnDemand servers. While OnDemand servers can be managed by a single administrator, administrative responsibilities can be divided among any number of people your organization requires. For example, in a distributed computing environment, one administrator can maintain the central library server while other administrators can load reports and maintain data on distributed object servers. Hopefully you will find using this book and administering OnDemand straightforward.

How this publication is organized

"Loading and Maintaining Data," beginning on page 1 provides information about loading reports into the system, including creating the index data that OnDemand client programs use to search for and retrieve documents. This section also describes how to maintain migrated index data and contains information about backup and recovery.

"Part 2. Managing ADSM storage" on page 65 provides information about managing ADSM devices and storage objects.

"Operating UNIX Servers," beginning on page 87 contains information that can help you maintain servers that run a UNIX operating system.

"Operating Windows NT Servers," beginning on page 107 contains information that can help you maintain servers that run the Windows NT Server operating system.

"Command Reference," beginning on page 133 provides detailed information about OnDemand commands.

Product support

The IBM Support Center maintains current information about OnDemand, including program temporary fixes (PTFs).

Before you install OnDemand, contact the IBM Support Center or your IBM software service representative to obtain the latest maintenance level of OnDemand.

If you encounter problems or errors running any of the OnDemand programs, you can call the IBM Support Center to obtain software problem and defect support. The phone number for the IBM Support Center is 1-800-237-5511. The OnDemand program number is 5622–622 for AIX, 5765–D60 for HP-UX, 5765–E20 for Solaris, and 5639–E12 for Windows NT. The OnDemand component ID is 5622-66200.

Our use of typefaces

Throughout this book, words and phrases appear in **Bold**, *Italic*, and other fonts. The following explains our convention when using these fonts.

Bold	Used for paragraphs that call attention to especially relevant information about a topic or command.
Italic	Used to emphasize concepts and terms.
`Monospace`	Indicates output of commands and programs in examples. Also used for information you are instructed to type.
UPPERCASE	Indicates parameter or command names and sometimes file and directory names.

OnDemand documentation

The following publications contain information about OnDemand Version 2.2:

Introduction and Planning Guide, G544-5281

Installation and Configuration Guide for UNIX Servers, G544-5598

Installation and Configuration Guide for Windows NT Servers, G544-5526

Administrator's Reference, S544-5293

Indexing Reference, S544-5489

Getting Started with the Administrative Client, S544-5463

User's Guide, SC26-9810

OS/2 Client Customization Guide, S544-5465

OLE Control and Win32 Client Customization Guide and Reference, S544-5466

Related documentation

The following publications contain information about ADSM Version 3, DB2 Universal Database Version 5, and PSF:

ADSM Messages, GC35-0271

ADSM for AIX Version 3 Quick Start, GC35-0273

ADSM for AIX Version 3 Administrator's Guide, GC35-0274

ADSM for AIX Version 3 Administrator's Reference, GC35-0275

ADSM for Windows NT Version 3 Administrator's Guide, GC35-0292

ADSM for Windows NT Version 3 Administrator's Reference, GC35-0293

ADSM for Windows NT Version 3 Quick Start, GC35-0295

ADSM for HP-UX Version 3 Administrator's Reference, GC35-0321

ADSM for HP-UX Version 3 Administrator's Guide, GC35-0320

ADSM for HP-UX Version 3 Quick Start, GC35-0322

ADSM for Sun Solaris Version 3 Administrator's Guide, GC35-0328

ADSM for Sun Solaris Version 3 Administrator's Reference, GC35-0329

ADSM for Sun Solaris Version 3 Quick Start, GC35-0330

DB2 Universal Database Extended Enterprise Edition for UNIX Version 5 Quick Beginnings, S99H-8314

DB2 Universal Database Version 5 Administration Getting Started, S10J-8147

DB2 Universal Database for UNIX Version 5 Quick Beginnings, S10J-8148

DB2 Universal Database for Windows NT Version 5 Quick Beginnings, S10J-8149

DB2 Universal Database Version 5 Administration Guide, S10J-8157

DB2 Universal Database Version 5 Command Reference, S10J-8166

DB2 Universal Database Version 5 Message Reference, S10J-8168

PSF for AIX: Print Submission, S544-3878

PSF for AIX: Print Administration, S544-3817

PSF for MVS: MVS Download Guide, G544-5294

PSF for OS/390: Download for OS/390, S544-5624

Related IBM EDMSuite Products

OnDemand is one of the web-enabled products included in the IBM Enterprise Document Management Suite (EDMSuite), which is a portfolio of IBM software that includes imaging, computer output to laser disk (COLD) document management, and workflow. In addition to OnDemand, EDMSuite contains four more products:

- ImagePlus VisualInfo
- ContentConnect
- MQSeries Workflow
- Domino Doc

For more information on EDMSuite, visit the EDMSuite homepage at http://www.software.ibm.com/data/edmsuite.

ImagePlus VisualInfo

ImagePlus VisualInfo is a distributed and scalable client/server solution that enables the management and storage of a broad array of document types, such as document images, graphics, spreadsheets, text, audio, and video. ImagePlus consists of scalable, multiplatform offerings for organizations of all sizes. ImagePlus implementations range from simple store-and-retrieve applications to image and workflow-enabling, very high-volume, transaction-oriented business processes. Some ImagePlus capabilities allow you to:

- Develop a robust library to store documents
- Handle multiple file types in addition to images, such as word processing documents, audio and video clips, and spreadsheets
- Manage and migrate documents across magnetic, optical, and tape storage automatically

ContentConnect

ContentConnect is a client toolkit based on Java that provides single-query access to multiple repositories from Web browsers, Lotus Notes clients, and stand-alone clients. ContentConnect provides a search engine that supplies reliable and immediate access to advanced document imaging, workflow, COLD archiving, and collaborative document management components. ContentConnect presents its search results in an organized, seamless, easy-to-read manner. ContentConnect not only connects document archives

and processes to other components of EDMSuite, but also connects its users to the document management world outside EDMSuite.

MQSeries Workflow (formerly FlowMark)

MQSeries Workflow is a workflow engine designed for the client/server environment. MQSeries Workflow is dedicated to managing the flow of work, allowing companies to integrate the applications required to meet the needs of their business processes. MQSeries Workflow allows you to:

- Separate application logic from business process rules
- Tie legacy and client/server applications to business process steps
- Monitor processes to show where a specific piece of work is in the overall process
- Record live production process data for analysis by management
- Model, animate, and simulate your business processes rapidly
- Enable processes across the Internet and Intranets

Domino Doc

Domino.doc is a Web-based document and content management solution that allows organizations to capture, file, retrieve, and distribute content across the Internet using desktop applications, Web browsers, or any Lotus Notes client. Domino.doc allows you to:

- Create document profiles that enforce the capture of key document search terms, so documents can be searched and retrieved easily
- Create a common library to provide a knowledge base of documents for your enterprise
- Use version control to provide a history of previous changes
- Use check-in and check-out controls to ensure data integrity

Part 1. Loading and Maintaining Data

This part of the book describes how to store data in OnDemand. The OnDemand data loading programs can process many types of data, including AFP data, line data, unformatted ASCII data, Adobe PDF files, and other types of user-defined data. The data loading programs also support many image formats, including BMP, GIF, JFIF, PCX, and TIFF. When you store data in OnDemand, the data loading programs typically generate the index data that is loaded into the OnDemand database, segment and compress the input data, and store the input data and resources on storage volumes. The OnDemand database is updated with the index information that allows users to search for and retrieve data using OnDemand client programs. OnDemand supports storing copies of input data on cache storage volumes and archive media. You can optionally create a backup copy of data stored on archive media.

This section also describes how to maintain migrated index data and contains information about backup and recovery.

Topic	Page
Loading reports	3
Loading image files	19
Loading user-defined data	29
Restarting a load process	39
Deleting a report	41
Data download exits and processing	43
Maintaining migrated index data	45
Backup and recovery	51

Chapter 1. Loading reports

Overview

You can archive the print output of existing application programs in OnDemand without changing the print data stream or writing programs to process the data. In OnDemand, the print output of applications is called a report. OnDemand provides programs that can automatically receive reports from MVS and OS/390 systems, index the reports, store the index information in the database, segment and compress report data, and load the compressed segments on storage volumes. After you store a report in OnDemand, users can query, retrieve, and view or print pages of the report using one of the OnDemand client programs.

The OnDemand data indexing and loading programs process files that reside on an OnDemand server. If you generate reports on an MVS or OS/390 system, you typically transfer the reports to the server and use the indexing and data loading programs to process the report. When you index a report, you must make sure the indexing program can access resources required by the report. Resources include page segments and fonts. You can transfer the resources to the server or you can mount a resource library using the Networked File System (NFS). Alternatively, you can index the report on the MVS or OS/390 system and then transmit the report, index, and resource files to the server.

You can create up to 32 index or filter fields for each type of report that you define to OnDemand, providing many ways for users to query information contained in a report. The number of index or filter fields that you define depends on the organization of the data in the report. For example, when you index a report that contains logical items, such as policies and statements, you might define index and filter fields for the statement date, customer name, customer number, balance due, transaction number, and amount. When you index a report that contains transaction data, such as a general ledger, you might define index and filter fields for the report date and transaction number. After you determine and define the index fields, OnDemand automatically extracts the index values from the report and stores them in the database.

Based on information you define for the application, OnDemand segments and compresses report data into storage objects. Depending on the storage management settings for the application group where you want OnDemand to

store a particular report, OnDemand can automatically store a copy of the report in cache storage and create a backup copy of the report on archive media.

The data loading program writes output messages to the OnDemand system logging facility at the end of each processing run. You can open the System Log folder and review information, such as the run date, the names of the input files, indexing information, and the number of records processed.

The data loading program provides checkpoints so that you can restart a previously interrupted load. You can recover or restart the steps that store index rows into the database and write storage objects on storage volumes.

Preparing to load reports

Estimating and allocating storage space

When you configure an OnDemand system, you calculate the total amount of magnetic, optical, and tape storage required to support the database, log files, archived reports, and the ADSM database and recovery log, along with temporary space needed by OnDemand programs. The amount of storage space required is a factor of how much data that you plan to archive in OnDemand, how long you need to maintain the data in OnDemand, the compression ratio achieved, how long you want OnDemand to maintain index data on magnetic storage volumes, and whether you maintain copies of reports in cache storage and on archive media. If you plan to index data on the server, you must allocate temporary work space for the data indexing program. Temporary work space is also required for the data loading program.

After you determine how much storage space you need and configure your OnDemand system, you organize the physical storage devices to support the various components of OnDemand, such as the database manager, data download, and cache storage and prepare storage devices for use by ADSM. Physically separating program directories, the database, cache storage, and temporary work spaces can improve performance and availability.

The *Introduction and Planning Guide* provides information, formulas, and worksheets that can help you calculate your storage requirements and contains recommendations for organizing storage devices to support OnDemand. The *Installation and Configuration Guide* describes how to define database, cache, and temporary storage areas to OnDemand and configure storage libraries and devices and define them to ADSM.

Defining the application group

When you store reports in OnDemand, you load index data in an application group and store compressed report data into storage objects associated with the application group. As part of defining a report to OnDemand, you specify the application group where you want OnDemand to store the index data and report data. Use the OnDemand administrator interface to define an application group.

If you plan to store reports on archive media, you must assign the application group to a storage set that identifies a client node in ADSM. The *Installation and Configuration Guide* provides information about ADSM client nodes.

Defining the application

Typically you define an application for each type of report (source of data) that you plan to store in OnDemand. When you create an application, you define information about the report, such as:

- The application group where you plan to store the data.
- The data format, for example, the type of data and the code page of the data.
- Indexing information, if you plan to index the report on the server.
- Load information that determines special processing that OnDemand must perform on the index data before loading it into the database

Use the OnDemand administrator interface to define an application.

Accessing reports, index data, and resources

The typical OnDemand system processes reports created by application programs that run on MVS and OS/390 systems. However, OnDemand supports processing almost any type of input data.

You must create index data for a report before you load the report into OnDemand. If you require resources to display and reprint pages of a report, you must also store the resources in OnDemand. There are several tools provided with OnDemand to help you extract index data from and generate index data about input files and collect resources:

- Enhanced ACIF, to index AFP and line data files
- OnDemand Generic Indexer, to index almost any type of input data
- OnDemand PDF Indexer, to index Adobe PDF files

The information in this section of the book shows examples of using ACIF to index reports. For details about indexing data, including how to use the Generic Indexer and the PDF Indexer, see the *Indexing Reference*.

You can use the enhanced AFP Conversion and Indexing Facility (ACIF) provided with OnDemand to index reports and collect the required resources into a resource group. ACIF processes the report and produces an output file to load, an index file for the database manager, and a resource group file that contains the resources required to view and reprint pages of the report.

ACIF can run on the server, an MVS system, or an OS/390 system. If you run ACIF on the server, you must transfer the report to the server and provide access to the resources required to process the report. If you run ACIF on an MVS or OS/390 system, you must transfer the output file, index file, and resources to the server.

You can use several methods to transfer files between the system and the server where you plan to process the data. For example:

- You can use Download to transmit a print dataset from the JES Spool to file systems on the server. The MVS (or OS/390) component of Download operates as one or more JES writers. You configure the writers to interpret JCL parameters, such as CLASS and DEST, and route spool files to an OnDemand server. You can specify other JCL parameters, such as FORMS and DATASET, and invoke a user exit program to determine the name of the application group and application so that OnDemand correctly processes the input data. The arsjesd command starts the server component of Download and monitors a specific TCP/IP port for data from an MVS or OS/390 system. You can start one or more occurrences of the arsjesd command, each monitoring a different port, receiving datasets from different Download writers, and storing data on different file systems. You can configure Download to automatically transmit reports created by all of the application programs that generate input to be stored in OnDemand. See the Download guide for your operating system for information about installing, configuring, and operating Download on the MVS or OS/390 system. The Download guide lists the supported JCL parameters and documents the user exit program that can be used to provide additional job information to OnDemand. Refer to “Chapter 6. Download exits and processing” on page 43 for more information about using the arsjesd command to invoke a user-defined command to process a downloaded file, the JCL parameters, and the additional job information. If you followed the instructions in the *Installation and Configuration Guide*, you configured the server to start the arsjesd command whenever the system is initialized.
- You can access files that reside on an MVS or OS/390 system with NFS. For example, if you plan to index reports on the server, you can mount the resource library on the server and access the resources required by a report. However, for performance reasons, we do not recommend using NFS to access datasets that contain report or index data.

- You can write reports to tape. For example, if the system that generates a report resides in a different location than the system where OnDemand processes the data, you can save a copy of a report on a tape volume and process it at a later time.

Configuring arsload

The arsload command is the primary OnDemand indexing and data loading program. The arsload command determines if the input file needs to be indexed, and if so, calls the required indexing program. The arsload command then calls programs that process the index data, load index rows into the database, compress report file segments into storage objects, and load the storage objects on storage volumes.

You run the arsload command each time that you want to store a report in OnDemand. You can either invoke the arsload command from the prompt or run the arsload command as a daemon (or Windows NT service) that periodically checks specified file systems for data to load.

The arsload command can extract the name of the application group to load from the input filename. You can also use the -G parameter to specify a different part of the filename name that identifies the application group. If the application group contains more than one application, you must identify the application to load. Otherwise the load will fail. You can use the -A parameter to specify the part of the filename that identifies the application. “Chapter 6. Download exits and processing” on page 43 provides information about other methods you can use to determine the application group and application to load.

When you load a report, OnDemand requires a userid with administrator authority for the application group and verifies that the user’s password is valid. There are several ways you can provide the userid and password:

- Use the -u and -p parameters each time you run the arsload command
- Use the -U parameter each time you run the arsload command to name a file that contains the userid and password for a user with administrator authority for the application group
- Set a default system administrator userid and password in the arsload.cfg file, which is used whenever you run the arsload command unless you specify otherwise (with the -u and -p parameters or if you specify a different userid/password file with the -U parameter)

The *Installation and Configuration Guide for UNIX Servers* describes how to set a default system administrator userid and password in the arsload.cfg file. The arsload.cfg file can be found in the /usr/lpp/ars/config directory (AIX) or the /opt/ondemand/config directory (HP-UX, Solaris).

Important: After you set the userid and password in the arsload.cfg file, remember to change the password any time you change the user's password in OnDemand. Otherwise the load will fail. The arsload command accepts an expired password. However, the arsload command fails if an incorrect password is specified.

Note: The arsload.cfg file and the -U parameter are not used on Windows NT servers. The OnDemand Load Data service logs on to OnDemand and accesses the application group with the userid and password of the instance owner (typically an OnDemand system administrator user). If you run the arsload command from the prompt, OnDemand unified login should be used to log on to OnDemand and access the application group. The *Installation and Configuration Guide for Windows NT Servers* provides information about configuring services and unified login.

See "Chapter 28. arsload command reference" on page 205 for a complete description of the arsload command, parameters you can specify, and information about the input files that the arsload command can process.

UNIX servers

Use the *init* facility to start the arsload command as a daemon. The init facility ensures that the arsload command starts each time the system is restarted. Each record that the init facility processes defines a specific process and contains runtime parameters. First, determine the parameters and options for the arsload command. Then add a record to the init facility.

The following shows an init record for an AIX server that starts the arsload command to check three different data download directories and create temporary index files in a fourth file system. The arsload command checks the data download directories every thirty minutes. The arsload command uses the FORMS part of the input filename to determine the application group to load. Each application group contains one and only one application. The arsload command logs on to OnDemand and accesses an application group with the userid and password from the arsload.cfg file:

```
arsload:once:/usr/lpp/ars/bin/arsload
     -d /arsacif/acif2 -d /arsacif/acif3 -d /arsacif/acif4
     -c /arsacif/acif1 -t 1800 -v
```

If you followed the instructions in the *Installation and Configuration Guide*, you configured the server to start the arsload command whenever the system is initialized. Consult your operating system documentation for information about the init facility, automating tasks, and operating system initialization.

Windows NT servers

Configure the arsload service to start automatically when the system is started. To configure arsload service:

1. Start the OnDemand configurator program.
2. Expand the server where you want to configure the arsload service.
3. Select Services.
4. Point to OnDemand Load Data and click the right mouse button.
5. From the pop-up menu, select Properties.
6. On the Service page, verify the Startup Type. See the online help for more information.
7. On the Directories page, assign directories to the service. (You add directories to the Available Directories list from the Directory page of the instance properties dialog box.)
8. On the Load Information page, configure other properties of the service. See the online help for information on the data that you can enter.
9. Click OK.

Note: When you configure one of the OnDemand services, you must stop and restart the OnDemand server service and the service you configured.

Indexing reports

You must create index data for a report before you store the report in OnDemand. You can index reports on OnDemand servers, MVS systems, and OS/390 systems.

Note: There are several tools provided with OnDemand to help you extract index data from and generate index data about input files and collect resources:

- Enhanced ACIF, to index AFP and line data files
- OnDemand Generic Indexer, to index almost any type of input data
- OnDemand PDF Indexer, to index Adobe PDF files

The information in this section of the book shows examples of using ACIF to index reports. For details about indexing data, including how to use the Generic Indexer and the PDF Indexer, see the *Indexing Reference*.

Indexing reports on OnDemand servers

If you index reports on the server, you must transfer the report to the server and provide access to the resources required by the report. Figure 1 depicts one method that you can use to index reports on the server.

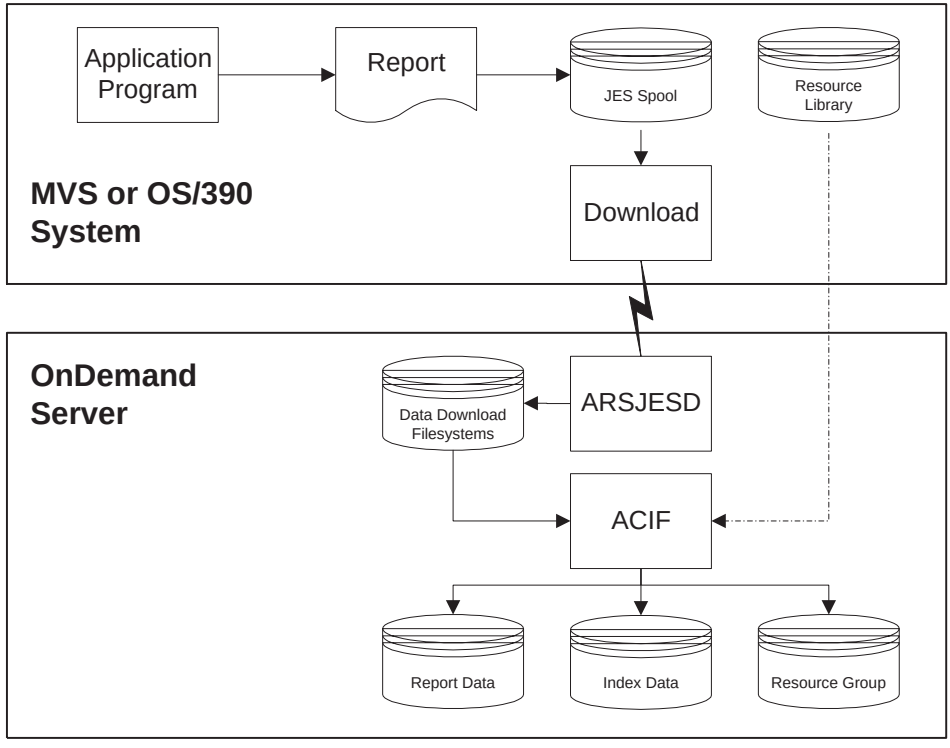


Figure 1. Indexing Reports on OnDemand Servers

In the example, Download is used to automatically transmit the report to the OnDemand server. After the report has been successfully transmitted to the server, the arsload command can process the data. You can either invoke the arsload command from the command line to process a specific file or use facilities of the server operating system to run the arsload command, polling the data download directory and processing files automatically after they are transmitted to the server. Since the report must be indexed before it can be archived in OnDemand, the arsload command automatically calls the data indexing program. In the example, the arsload command calls ACIF to index the report. (In the example, ACIF accesses the resource library through NFS.)

The *Indexing Reference* contains important information about indexing reports, including details about the parameters, options, and data values you can specify to index reports with ACIF.

Indexing reports on MVS and OS/390 systems

If you index reports on MVS and OS/390 systems, you must transfer the output file, index file, and resource group file to the server where you plan to run the arsload command. Figure 2 depicts one method that you can use to index reports on MVS and OS/390 systems.

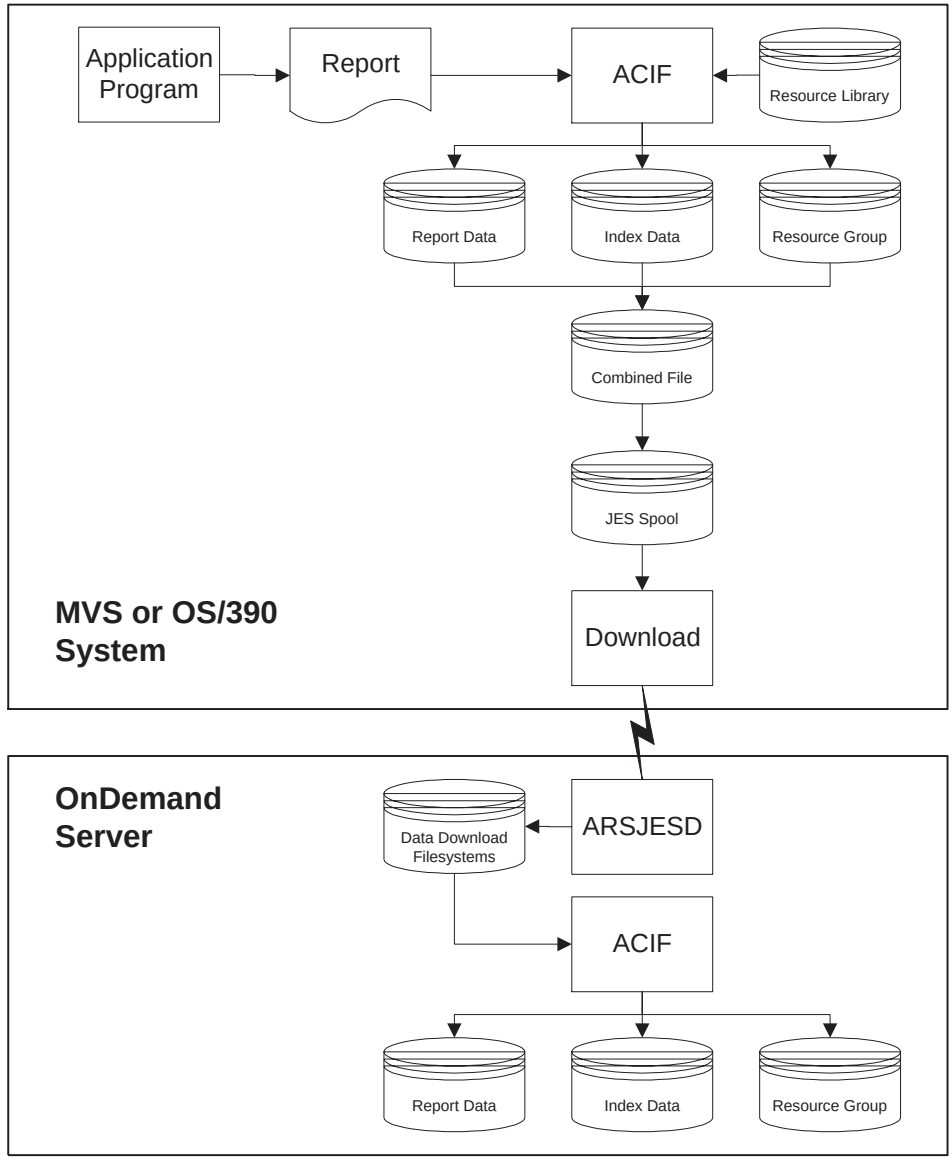


Figure 2. Indexing Reports on MVS and OS/390 Systems

In the example, ACIF processes a line data report, generates index data, converts the input to AFP data, and collects resources from a library on the system. After running ACIF, you must combine the three output files generated by ACIF into one file and write the combined file to the JES Spool. Download can be used to automatically transmit the combined file to the OnDemand server. After the combined file has been successfully transmitted to the server, you must run ACIF on the OnDemand server to process the combined file. Using parameters to convert the data and name a default form definition (**no** trigger, field, or index parameters), ACIF generates separate index, resource, and output files that can be processed by the OnDemand data loading program. Contact the IBM support center if you have questions about indexing data on MVS and OS/390 systems and processing the output on an OnDemand server. "Chapter 6. Download exits and processing" on page 43 provides more information about processing files transmitted by Download.

The *Indexing Reference* provides important information about indexing reports, including details about the parameters, options, and data values that you can specify to index reports with ACIF.

Processing index data, reports, and resources

After the report has been indexed, the arsload command processes the index file to prepare the index data for loading into the application group and the report and resource group files to prepare the storage objects that the storage manager loads on storage volumes.

Processing index data

The index file created by the indexing program is further processed before it is loaded into the database. The arsload command reads the definitions and parameters that you specified when you created the application and application group and does the following processing:

- Reads the application group definitions about the database fields.
- Reads the application definitions and does preprocessing, such as mapping index fields to database fields. This step is required if the field names in the index file are not the same as the database field names. You must map the field names so that OnDemand stores index values in the correct database fields.
- After preprocessing the index file, the arsload command creates the database rows:
  - One row for every group of indexed pages in a report file that contains a sorted transaction value.
  - One row for every indexed item in a report file that contains logical items, such as policies and statements.

- Reads the application definitions and does postprocessing, such as dropping duplicate index records. You specify postprocessing parameters when you define the application.
- Passes the index rows to the database manager. The rows consist of fields that contain the index values that the indexing program extracted from the report and other fields generated by OnDemand. An index row contains:
  - One column for each index and filter field defined in the application group.
  - One or more columns of OnDemand control information.

Processing reports and resources

The arsload command segments and compresses report data into storage objects. Segmenting the report improves the efficiency of queries and can improve the time required to retrieve and display the report. Compression improves the efficiency of the storage manager. In this step, the arsload command:

- Reads the application definitions to determine the compression method for the report and the resources.
- Reads the application group storage management definitions.
- Segments the report and compresses the segments into storage objects. OnDemand compresses report data into approximately 100 KB blocks (a default value) and places the blocks into a storage object. OnDemand uses a 10 MB storage object (a default value) to improve storage efficiency and performance. Compressed data does not span storage objects. OnDemand assigns unique file names to the storage objects, sequentially numbering the objects within an application group.

Loading index data

The arsload command calls the database manager to load the index rows into an application group table. Depending on the database organization that you selected when you defined the application group, OnDemand either creates a new table for each report file load or adds the rows to an existing database table. OnDemand implements a segment table as a high level index to application group data. Rows in the segment table identify the tables of index data for each application group.

Each row in the segment table identifies a different table of application group data. Fields in the segment table contain the application group identifier and a set of dates. The dates represent the earliest and latest date that can be found in that segment of application group data. OnDemand uses the entries in the segment table to limit queries to a specific table or set of application group tables.

When you load a report into OnDemand, the database manager updates the segment table, if the beginning date in the report is earlier than the date already stored in the table.

OnDemand limits the size of an application group table, to provide better performance and storage management. The number of rows in a database table can be set when you define the application group. The default size of a database table is 2.5 million rows. The database manager automatically closes an application group table and opens a new table when this threshold is reached. The database manager updates the segment table with the latest ending date in the segment of data before closing an application group table.

Loading storage objects

The arsload command calls the storage manager to copy storage objects to cache storage and archive media. The storage manager uses application group information to determine where and when to copy storage objects.

Cache storage

Cache storage supports short-term, high-speed retrieval of documents. Cache storage can be located on one or more object servers, depending on how storage sets and primary storage nodes are defined on your system. Documents are always cached on the object server where the primary storage node resides.

If the application group is configured to cache data, the storage manager copies the storage object to cache storage. The `Cache Data for xx Days` setting on the Storage Management page determines whether OnDemand caches documents stored in the application group, and if so, for how long.

Primary storage node

The primary storage node provides two functions. It identifies the object server where documents are cached, and when the storage set supports archive media, it identifies an ADSM client node. OnDemand uses ADSM to maintain storage objects on archive media for long-term storage and for backup copies of reports.

The storage manager can copy the storage object to archive media when the report is loaded into the system or at a later time. If application group is configured to copy data to archive media, OnDemand copies the storage object to archive media.

If the primary storage node is cache-only, the storage manager copies the storage object to a cache file system on the object server. If the primary storage node identifies a client node in ADSM, the storage manager copies the storage object to archive media.

Secondary storage node

A secondary storage node can be used to maintain a copy of files stored on archive media. However, there is no automatic way of accessing the files in a secondary storage node if the files in a primary storage node become corrupted or unavailable. For that reason, if you need to maintain a backup copy of reports on archive media, we recommend that you configure a copy storage pool in ADSM. Refer to the *Installation and Configuration Guide* for information about configuring a copy storage pool.

If the primary storage node has been assigned to a secondary storage node, the storage manager stores a second copy of the storage object on archive media.

Resources

OnDemand always stores resources in cache storage, providing faster retrieval when a user selects an item for viewing. The arsload command stores only unique resources, checking the resource identifier against resources previously stored in OnDemand.

In addition, if the primary storage node identifies a client node in ADSM, the storage manager copies the resources to archive media.

Backing up databases

After you store data into OnDemand, we recommend that you create a backup copy of the OnDemand database and the ADSM database.

- When you backup the OnDemand database, you protect control information and index data that OnDemand and the database manager need to support the system. Also, if you maintain log files on disk storage, the database manager can reclaim the disk space used by the log files.
- When you backup the ADSM database, you protect control information that ADSM uses to manage report files on optical storage devices.

We recommend that you backup the databases at least once a week, and more often, if you load reports every day. If you are using DB2 to manage the OnDemand database, you can use the arsdb command to backup the database. If you are using SQL Server to manage the OnDemand database, we recommend that you use the Database Maintenance Plan Wizard to configure

and schedule database backups. For ADSM, you can schedule the ADSM database backup using the ADSM administrative client.

Backup the OnDemand database

For DB2, OnDemand provides the arsdb command so that you can create a backup image of the OnDemand database. When you use the arsdb command to backup the database and you maintain log files on disk, OnDemand automatically removes log files from the archived log file directory. When you use ADSM to maintain DB2 files, the ADSM policy domain determines when archived log files are removed from the storage pool.

The arsdb command supports two types of database backups, an online backup and an offline backup:

- An online backup can be taken when other applications or processes are connected to the database. That is, other applications and processes can continue to read or modify data while the backup is in progress. An online backup can be scheduled by the operating system.
- During an offline backup, only the backup task is connected to the database. Before starting an offline backup, we recommend that you stop the OnDemand server programs and other related processes.

If your production schedule allows, we strongly encourage you to create an offline backup of the database on a regular schedule, perhaps once a week. Taking an offline backup on a regular schedule can reduce the time required to rebuild the database if you need to do so.

"Chapter 8. Backup and recovery" on page 51 provides more information about backup and recovery strategies and options. Refer to "Chapter 25. arsdb command reference" on page 173 for information about how to use the arsdb command to create a backup image of the OnDemand database.

Backup the ADSM database

After you archive data in OnDemand, we recommend that you backup the ADSM database. A backup image can be used to recover the database, in the event a hardware failure occurs or the database becomes corrupted.

The *Installation and Configuration Guide* shows how to define a backup device to ADSM and create a full backup of the ADSM database. See the *ADSM Administrator's Guide* for details about protecting the ADSM database.

Protecting the document cache

The document cache is the primary, short-term storage location for reports. If you do not copy reports to archive media when you store them in OnDemand, you need to consider how you can recover the reports in the event you need to do so (for example, if a cache storage device fails).

The document cache can be protected by maintaining it on RAID storage subsystems. RAID storage can provide excellent availability, allowing users to access reports even if a disk or controller fails. The *Introduction and Planning Guide* provides more information about using RAID storage with OnDemand. However, RAID storage is not fail safe. There may be situations when, because of multiple disk or controller failures, users cannot access reports. We encourage you to use ADSM to maintain a copy of reports on archive media.

Checking messages and verifying totals

The arsload command writes output messages to the OnDemand system log at the end of each processing run. You can open the System Log folder and review the messages, such as the run date, the name of the input file, indexing information, and the number of rows loaded into the database. For example:

```
arsload: Processing file >br1010a<
arsload: Fri Mar 29 12:37:23 MST 1996 Indexing started,
          394419060 bytes to process
arsload: Fri Mar 29 12:41:33 MST 1996 Indexing completed
arsload: Fri Mar 29 12:41:35 MST 1996 Load Started,
          394419060 bytes to load
  Resource br1010a.res matches the resource 2
  OnDemand Load ID = >6850-25-0-15FAA-9577-9577<
  Loaded 130000 rows into the database
arsload: Fri Mar 29 13:05:14 MST 1996 Load Completed
arsload: Processing successful for file >br1010a<
```

You can verify the number of records that OnDemand added to the database:

- When processing a report and creating indexed groups of pages, the number of pages in the report divided by the number of pages in an indexed group of pages should equal the number of rows added to the database. For example, if the report contains 150,010 pages and there are 100 pages in an indexed group of pages, OnDemand should have loaded 1,501 rows into the database.
- For a report that contains logical items, such as statements and policies, the number of rows loaded into the database should equal the number of indexed items in the report. For example, if the report contains 1,000 statements, OnDemand should have loaded 1,000 rows into the database.

The arsload command creates one or more Load IDs while processing the report. The OnDemand Load ID represents the storage objects that the arsload command placed in the application group during the load. The Load ID can be used to identify a specific report load. You can use the arsadmin command and specify the Load ID to maintain storage objects, for example, to copy storage objects from optical to cache storage and to delete storage objects from OnDemand. Table 1 lists the fields in a Load ID.

Table 1. OnDemand Load ID

Load ID Field	**Meaning**
6850	Application group identifier
25	Primary node identifier
0	Secondary node identifier
15FAA	The load identifier within the application group
9577	The earliest date in the report
9577	The latest date in the report

Chapter 2. Loading image files

Overview

OnDemand provides support for archiving and retrieving letters and other types of correspondence. The most straight forward way to archive this type of information is to transform a hardcopy letter into an image file with a scanner and image capture software. In addition to scanning the letter and saving it as an image file, you must create index data for the letter. You can then use OnDemand programs to store the index information in the database and the file on a storage volume. Users can then query, retrieve, and view or print copies of the letter using one of the OnDemand client programs.

This chapter outlines how to use a scanner and image capture software to create the image files. Before capturing and archiving files, we explain how to define an application group to store the letters, including the database fields, an application, that describes the format of the data and other processing characteristics, and a folder, that users open to search for and retrieve the files.

OnDemand provides a generic index file format to specify index data for image files. The generic index file takes the place of running a program such as ACIF, which can extract index data and perform other sophisticated indexing functions. The generic index file contains the field names and values and identifies the image files to load into OnDemand. You can create up to 32 index or filter fields for each file that you archive in OnDemand, providing many ways for users to query and retrieve the letters. The number of index or filter fields that you define depends on user retrieval requirements. In our example, we defined index fields for the date on the letter, the name of the person that sent the letter, the company name, the subject of the letter, and the date that we scanned the letter into the system.

Defining the application group

When you archive files in OnDemand, you store index data into an application group table and load the files into storage objects associated with the application group. This topic provides information about specific application group properties and settings that we made when we defined an application group to archive letters. Use the OnDemand administrator interface to define an application group.

Database organization

We accepted the default Multiple Loads per Database Table database organization and table size of 2.5 million rows. Each time that we store a letter in the database, OnDemand adds one index record to the application group table.

Expiration type

We set the expiration type to Load. This means that OnDemand deletes a report file load at a time from the application group. Depending on how we load data into the application group, one or more letters may be deleted at a time. That is, if we process a generic index file that contains index data for several letters, OnDemand deletes all of the letters from the application group on the same date.

Permissions

We set the Access permission at the application group level. This means that all users defined to OnDemand can access data stored in the application group.

Field definition

We defined the following database fields, which contain the index data that enable users to locate letters based on different criteria, such as the date of the letter, the name of the person that sent the letter, and the subject of the letter.

sdate The date we scanned the letter into the system. Defined as a date field.

ldate The date on the letter. Defined as a date field.

name The person that sent the letter. Defined as a string field that contains variable length data.

company
The person's company. Defined as a string field that contains variable length data.

subject
The subject of the letter. Defined as a string field that contains variable length data.

OnDemand stores index data in the fields when we load a letter into the application group. We specify the name of the generic index file that contains the index field names and values when we run the load command.

Defining the application

You must define an application for each different type of file that you plan to archive in OnDemand. This topic provides information about specific application properties and settings that we made when we defined an application for archiving letters. Use the OnDemand administrator interface to define an application.

Application group

We assigned the application to the application group that we created in "Defining the application group" on page 19 to store the letters and the index data.

Data format

We selected TIFF as the data type. When we save the scanned image of a letter, we must save the file as a TIFF file.

Indexer

We selected Generic as the indexer. You must index image files using the generic index file format supported by OnDemand.

Data compression

We specified None, so that OnDemand does not compress the image file. We did this because we plan to save the image files as compressed TIFF files. If you save scanned files as uncompressed TIFF files, you should enable compression, so that OnDemand can compress the files for better storage efficiency and retrieval performance.

Defining the folder

You must define a folder so that users can search for and retrieve the files that you archive in OnDemand. This topic provides information about specific folder properties and settings that we made when we defined a folder to provide access to letters. Use the OnDemand administrator interface to define a folder.

Application group

We selected the application group that we created in "Defining the application group" on page 19 to store the letters and the index data. When users open the folder, they can access the letters stored in the application group.

Permissions

We set the Access permission at the folder level. That means that all users defined to OnDemand can open the folder.

Folder fields

We defined the following fields, which allow end-users to locate letters based on different criteria, such as the date of the letter, the name of the person that sent the letter, and the subject of the letter.

Scan Date
The date we scanned the letter into the system.

Letter Date
The date on the letter.

From The person that sent the letter.

Company
The person's company.

Subject
The subject of the letter.

We mapped each of the fields to its corresponding application group field.

Acquiring the images

We scanned each letter using an IBM 3119 PageScanner. We previewed and saved the image files using the IBM OS/2 Image Capture Facility. Each letter was saved to a separate file as a compressed TIFF file.

Settings:

- Paper Size 8.5 inches
- Bi-level image
- 300 DPI
- Compressed TIFF

If you plan to let OnDemand compress the data, save the files as uncompressed TIFF files.

Accessing the image files

The OnDemand data indexing and loading programs run on the server. Therefore, the files that you want to archive in OnDemand must reside on the server or you must provide access to the files.

You can transfer the files to the server using one of a number of methods. We used *ftp* to copy the files (binary) from the PC that performed the scanning operation to the server where we ran the load command.

You can access files that reside on another system with NFS. For example, we could have mounted the directory that contained the image files on the server, to access the files required by the load program.

Creating index data

You must create index data for files before storing them in OnDemand. The index data for image files must use the generic index file format provided by OnDemand. Refer to the *Indexing Reference* for details about the generic index file.

Use a standard text editor to create the index file. Figure 3 on page 24 shows a sample generic index file.

```
FIELD NAMES BEGIN:
sdate
ldate
name
company
subject
FIELD NAMES END:
12/18/95
09/01/95
Mr. Earl Hawkins
Soft Products
optical storage devices
letter1.tif
0
0
12/18/95
09/01/95
Hans G. Piker
IBM Corporation
optical storage devices
letter2.tif
0
0
12/18/95
09/16/95
Laurie Unicola
Dove Properties
account balance due
letter3.tif
0
0
12/18/95
10/01/95
George VanLocal
Express American
airline fares
letter4.tif
0
0
```

Figure 3. Generic Index File

We saved the index file as *letters.ind* on the server in the directory where we plan to run the load program.

Configuring arsload

The arsload command is the program that OnDemand uses to process input files. The arsload command determines if the input file needs to be indexed, and if so, calls the required indexing program. The arsload command then calls programs that process the index data, load index rows into the database,

optionally compress the file, segment the file into storage objects, and load the storage objects on storage volumes. Please refer to “Chapter 28. arsload command reference” on page 205 for a complete description of the arsload command, parameters that you can specify, and information about the files that the arsload command can process.

You schedule and run the arsload command each time that you want to archive a file or set of files in OnDemand. You can either invoke the arsload command from the command line or run the arsload command as a daemon (UNIX servers) or service (Windows NT servers) that periodically checks specific file systems for data. To support a low volume scanning operation, we chose to run the arsload command from the command line, specifying the required options. For example:

```
arsload -u admin -p "" -n -g Letters letters
```

The parameters for the arsload command include:

-u admin
: The name of an OnDemand user that can add documents to the application group. This is typically a user with administrator authority for the application group.

-p *""* The password for the OnDemand user.

-n Do not delete the input files.

-g Letters
: The name of the application group to load.

letters The filename of the input files.

In our example, the arsload command expects to find the *letters.ind* file in the current directory. This file contains index information that we created in “Creating index data” on page 23, which is in the OnDemand generic index format. We copied the image files from the PC to the server as binary files with a filetype of *tif*.

Processing the index data and the image files

The arsload command processes the generic index file, loading the index data into the application group and the image files on storage objects that the storage manager loads on storage volumes.

Processing index data

The arsload command processes the generic index file before loading the data in the database. The arsload command reads the definitions and parameters that we specified when we created the application and application group and performs the following processing:

- Reads the application group definitions about the database fields.
- Reads the application definitions and performs preprocessing, such as mapping index fields to database fields. This step is required if the field names in the index file are not the same as the database field names.
- After preprocessing the index file, the arsload command creates one row for every image file referenced in the index file. We created a set of index values for each file.
- Reads the application definitions and performs postprocessing, such as dropping duplicate index records. You can specify postprocessing parameters when you define the application.
- Passes the index rows to the database manager. The rows contain the index values and other fields generated by OnDemand. An index row contains:
  - One column for each index and filter field defined for the application group.
  - One or more columns of OnDemand control information.

Processing the image files

The arsload command processes each image file referenced in the generic index file. The arsload command reads the definitions and parameters that we specified when we created the application group and application and performs the following processing:

- Reads the application definitions to determine the compression method for the files.
- Reads the application group storage management definitions.
- Segments the image files and stores the segments into storage objects. OnDemand uses a 10 MB storage object (the default value) to provide better storage efficiency and performance. OnDemand assigns unique file names to the storage objects, sequentially numbering the objects within an application group.

Verifying processing

The arsload command writes output messages to the system log at the end of each processing run. You can open the System Log folder and review the messages, such as the run date, the name of the input file, indexing information, and the number of rows loaded into the database. For example:

```
arsload: Processing file >letters<
arsload: Mon Dec 18 13:04:37 MST 1995 Indexing started,
          116017 bytes to process
  OnDemand Load ID = >6927-0-0-1FAA-0-0<
  Loaded 4 rows into the database
arsload: Mon Dec 18 13:05:14 MST 1995 Loading Completed
arsload: Processing successful for file >letters<
```

You can verify the number of records that OnDemand added to the database. In the example, the number of rows loaded into the database should equal the number of index records in the generic index file.

The OnDemand Load ID represents the storage objects that the arsload command placed in the application group during the report file load. The Load ID can be used to identify a report file load. You can use the arsadmin command and specify the Load ID to maintain storage objects, for example, to copy storage objects from optical to cache storage and to delete storage objects from OnDemand.

Chapter 3. Loading user-defined data

Overview

OnDemand supports storing and retrieving almost any type of data. By default, OnDemand supports AFP, Line, PDF, GIF, JFIF (JPEG), PCX, and TIFF data. However, OnDemand is not limited to maintaining these types of data. OnDemand provides a user-defined data type to support any other type of data that you want to store in the system. For example, you can define an OnDemand application for Lotus WordPro files so that when a user retrieves one of the files from the system, OnDemand automatically starts Lotus WordPro to open the file.

To store user-defined data on the system, you must create index data for the input files and you must register the file type of the input file with OnDemand. The file type determines the program used to open a file when a user retrieves it from the system. The file type must also be registered with the client operating system. If your OnDemand system supports client programs that run under different operating systems, you must register the same file type on all of the client operating systems.

OnDemand provides a generic index file format you can use to specify index data for user-defined data. The generic index file takes the place of running a program such as ACIF, which can extract index data from the input and perform other sophisticated indexing functions. The generic index file contains the field names and values and identifies the files to load. You can create up to 32 index or filter fields for each input file, providing many ways for users to query and retrieve files. The number of index or filter fields that you define depends on user retrieval requirements. For example, you may want to define index fields for the date, author, and subject of a WordPro file.

About the example

We've provided an example that shows how to define Lotus WordPro files to OnDemand. We'll explain how to define an application group to maintain the files, an application to index, load, and view the files, and a folder that users can open to search for and retrieve the files. The example shows most of the key tasks required to define the application group, application, and folder. For details and to see a comprehensive example of defining a report to OnDemand, please see *Getting Started with the Administrator Interface*.

Defining the application group

When you store input data on the system, OnDemand loads index data into an application group table and copies the input data to storage volumes. The application group definition contains information that OnDemand uses to maintain the data, such as when to delete files and on what storage media to copy files. This topic provides information about specific application group properties and settings that we made when we defined an application group to maintain user-defined data. Use the OnDemand administrator interface to define an application group.

Database organization

On the General page, accept the default Multiple Loads per Database Table database organization and table size of 2.5 million rows. Each time that we load a Lotus WordPro file into the system, OnDemand adds one index record to the application group table.

Expiration type

On the Storage Management page, set the expiration type to Load. This means that OnDemand deletes all of the data related to a load process at the same time. Depending on how we load data into the application group, one or more files may be deleted at a time. For example, if a load process stores 100 files on the system, OnDemand deletes all 100 files (and the index data) from the application group on the same date.

Permissions

On the Permissions page, set the Access permission at the group level. Only users that belong to a specific OnDemand group can search for and retrieve the Lotus WordPro files.

Field definition

Define the database fields on the Field Definition page. After loading data into the system, the database fields contain index data that allow users to locate files based on different criteria, such as the date, author, and subject of the file.

fdate The date associated with the file. For example, the date the file was created or the date the file was published. Defined as a date field.

author The author of the file. Defined as a string field that contains variable length data.

subject
The subject or purpose of the file. Defined as a string field that contains variable length data.

OnDemand creates one index record and stores index data in the database fields when we load a file into the application group. We must supply the index data in a Generic index file. The Generic index file identifies the input file or files to load and contains the index field names and values.

Defining the application

You must define an application for each different report or source of data that you plan to store in OnDemand. The application definition contains information used by the OnDemand indexing, loading, and viewing programs. This topic provides information about specific application properties and settings that we made when we defined an application to maintain Lotus WordPro files. Use the OnDemand administrator interface to define an application.

Application Group

On the General page, assign the application to the application group created in "Defining the application group" on page 30. This is the application group where we will store the Lotus WordPro files when we load them into the system.

Data Format

On the View Information page, select User Defined from the Data Type list. Because Lotus WordPro files are not one of the default OnDemand data types, we must define them as a user-defined data type.

File Extension

When we select User Defined, we must enter a value in the File Extension field. The File Extension determines the program used to open a user-defined file when it is retrieved from the system. In our example, we enter LPW, for Lotus WordPro.

The File Extension must also be registered to the operating system. For example, in Windows NT, start Explorer. From the View menu, select Options. The File Types page shows the file types that are currently registered with the system. Verify that the file type that you specified in the File Extension field is registered with the system. If it is not, you must define a new type. Refer to the online help for details. If your OnDemand system supports client programs that run under different operating systems, you must register the same file type on all of the client operating systems.

Indexer

Changing to the Index Information page, select Generic from the Indexer list. We must use the Generic indexer to supply index data for all data types except AFP, Line, and PDF.

Data compression

On the Load Information page, accept the default Data Compression of OD77, because we want OnDemand to compress input files before storing them on the server. There are no resources with user-defined input data.

Defining the folder

Users open a folder to search for and retrieve documents from OnDemand. A folder provides access to documents stored in one or more application groups. This topic provides information about specific folder properties and settings that we made when we defined a folder to provide access to Lotus WordPro files. Use the OnDemand administrator interface to define a folder.

Application group

On the General page, select the application group created in “Defining the application group” on page 30. Based on the folder and application group permissions, when users open the folder, they can access the WordPro files stored in the application group.

Permissions

On the Permissions page, set the Access permission at the group level. Only users that belong to a specific OnDemand group can open the folder.

Folder fields

On the Field Definition page, define the folder fields. The client lists the fields, letting users enter search criteria to locate the files they need. Define the following fields:

File Date
The date associated with the file. For example, the date the file was created or the date the file was published.

Author
The person that created the file.

Subject
The subject or purpose of the file.

Field Mapping

On the Field Mapping page, map each folder field to its corresponding application group field. OnDemand uses the field mapping to build the query string sent to the server.

Accessing the input files

The OnDemand data indexing and loading programs run on the server. The files that you want to store in OnDemand must reside on the server or you must provide access to the files:

- You can transfer the files to the server using one of a number of methods. For example, you can use FTP to copy the files (binary) from a PC where the files were created to the server where you plan to run the load process.
- You can provide access to files that reside on another system with NFS. For example, you can mount the directory that contains the WordPro files on the server where you plan to run the load process.

Creating the index data

You must create index data for input files before storing them into the system. OnDemand provides the Generic indexer to process the index data for user-defined files. The Generic indexer reads the index data from a generic index file. Refer to the *Indexing Reference* for details about the Generic indexer and the generic index file.

Use a standard text editor to create the generic index file. Figure 4 on page 34 shows a sample generic index file, used to index four input files. Each index record in the file contains values for the three database fields and identifies the name of the input file.

```
FIELD NAMES BEGIN:
fdate
author
subject
FIELD NAMES END:
12/18/95
Jessica Hawkins
optical storage devices
letter1.lpw
0
0
12/18/95
Paul Garvey
optical storage devices
letter2.lpw
0
0
12/18/95
Randy Perkins
account balance due
letter3.lpw
0
0
12/18/95
George July
airline fares
letter4.lpw
0
0
```

Figure 4. Generic Index File

Save the index file as *lpw.ind* in a directory on the server.

Configuring arsload

The arsload command is the program that OnDemand uses to process input files. The arsload command determines if the input file needs to be indexed, and if so, calls the indexing program (identified in the application). The arsload command then calls programs that process the index data, load index records into the database, optionally compress input data, segment input data into storage objects, and load the storage objects on storage volumes. Please refer to “Chapter 28. arsload command reference” on page 205 for a complete description of the arsload command, parameters that you can specify, and information about the files that the arsload command can process.

You schedule and run the arsload command each time that you want to store data on the server. You can either invoke the arsload command from the command line or run the arsload command as a daemon (UNIX servers) or

service (Windows NT servers) that periodically checks specific file systems for data. To store a few WordPro files at a time, you can run the arsload command from the command line, specifying the required options. For example:

```
arsload -u admin -p "" -n -g 'Lotus WordPro documents' lpw
```

The parameters for the arsload command include:

-u admin
: The name of an OnDemand user that can add documents to the application group. This is typically a user with administrator authority for the application group.

-p ″″
: The password for the OnDemand user.

-n
: Do not delete the input files.

-g 'Lotus WordPro documents'
: The name of the application group to load.

lpw
: The filename of the generic index file.

In our example, the arsload command expects to find the *lpw.ind* file in the current directory. This is the generic index file created in "Creating the index data" on page 33. Because the generic index file does not include full path information for the input files, they also must reside in the current directory.

Processing the input data

The arsload command processes the generic index file, loading the index data into the database and the input files on storage volumes.

Processing index data

The arsload command processes the generic index file before loading the data in the database. The arsload command reads the definitions and parameters that we specified when we created the application and application group and performs the following processing:

- Reads the application group definitions about the database fields.
- Reads the application definitions and performs preprocessing, such as mapping index fields to database fields. This step is required if the field names in the index file are not the same as the database field names.
- After preprocessing the index file, the arsload command creates one record for each WordPro file referenced in the generic index file. We created a set of index values for each file.

- Reads the application definitions and performs postprocessing, such as dropping duplicate index records. You can specify postprocessing parameters when you define the application.
- Passes the index records to the database manager. The records contain the index values and other fields generated by OnDemand. An index record contains:
  - One column for each index and filter field defined for the application group.
  - One or more columns of OnDemand control information.

Processing the input files

The arsload command processes each WordPro file referenced in the generic index file. The arsload command reads the definitions and parameters that we specified when we created the application group and application and performs the following processing:

- Reads the application definitions to determine the compression method for the files.
- Reads the application group storage management definitions.
- Segments the WordPro files and stores the segments into storage objects. OnDemand uses a 10 MB storage object (the default value) to provide better storage efficiency and performance. OnDemand assigns unique file names to the storage objects, sequentially numbering the objects within an application group.

Verifying processing

The arsload command writes output messages to the system log at the end of each processing run. You can open the System Log folder and review the messages, such as the run date, the name of the input file, indexing information, and the number of rows loaded into the database. For example:

```
arsload: Processing file >lpw<
arsload: Mon Dec 18 14:14:47 MST 1995 Indexing started,
         5116017 bytes to process
  OnDemand Load ID = >6927-0-0-4FAA-0-0<
  Loaded 4 rows into the database
arsload: Mon Dec 18 14:25:11 MST 1995 Loading Completed
arsload: Processing successful for file >lpw<
```

You can verify the number of records that OnDemand added to the database. In the example, the number of rows loaded into the database should equal the number of index records in the generic index file.

The OnDemand Load ID represents the storage objects that the arsload command placed in the application group during a load process. The Load ID can be used to identify a specific load process. You can use the arsadmin

command and specify the Load ID to maintain storage objects, for example, to copy storage objects from optical to cache storage and to delete storage objects from OnDemand.

Chapter 4. Restarting a load process

The arsload command terminates when an unrecoverable error occurs during index, database, or storage manager processing. Termination processing includes setting a return code and writing error messages to the system logging facility. Open the System Log folder and view the message log for the load process to find return codes and messages.

If the arsload command fails during index processing, correct the problem and then restart the load from the beginning.

If an error occurs during processing of the database or the storage manager:

- Determine the problem and resolve the problem.
- If a Load ID is listed in the messages written to the system log, use the arsadmin command to unload the data. Refer to “Chapter 5. Deleting a report” on page 41 for instructions about how to delete a report from OnDemand.
- Restart the load from the beginning.

Chapter 5. Deleting a report

You can use the *unload* function of the arsadmin command to delete a Load ID from one or more storage locations. A Load ID represents the storage objects that the arsload command placed in an application group during a load process.

When you use the unload function, you specify the name of the application group, the Load ID, and optionally, a mask. Obtain the Load ID from the message log generated when you loaded the report. If there is more than one Load ID listed in the message log, use the last Load ID listed. The mask determines the storage locations where the storage manager unloads the storage objects. That is, the cache, primary node, or secondary node. By default, the unload function deletes storage objects from all three locations. During unload processing, the arsadmin command deletes rows of index data from the database and updates the database with segment information.

The following shows how to run the arsadmin command on a UNIX server (from the prompt) to delete a report from all storage locations in the Credit application group:

```
arsadmin unload -g 'Credit'-L 1220-1-0-8FAA-5538-5673
```

To run the arsadmin command in Windows NT:

1. Start the OnDemand Command Line.
2. Enter the command and parameters at the prompt. For example:

   ```
   arsadmin unload -g 'Credit' -L 1220-1-0-8FAA-5538-5673
   ```

Open the System Log folder to view messages generated by the arsadmin command.

See “Chapter 22. arsadmin command reference” on page 153 for a complete description of the arsadmin command, parameters that you can specify, and information about how to maintain storage objects in OnDemand.

Chapter 6. Download exits and processing

Overview

When processing files transmitted by Download, unless you specify otherwise, the data loading program uses the filename to identify the application group to load. If the filename does not contain the correct information, the load will fail. If the application group to load contains more than one application, you must identify the application to load. Otherwise, the load will fail.

OnDemand provides ways for you to identify the correct application group and application to load. For example, you can use the -A and -G parameters of the arsload command to specify the parts of the downloaded filename that identify the application group and application. However, if the filename cannot be used to identify the application group and application, you must use some other method to determine this information before you can load the data.

Using Download

You can use Download to transmit reports from MVS and OS/390 systems to OnDemand servers. When you use Download as part of an automated data loading process, you may need to write a user-exit program to process files before they can be used by the OnDemand data loading program. You must write a user-exit program if the filename of the file transmitted to the server cannot be used to identify the application group to load.

For example, suppose you use a report distribution system to place the output of application programs on the spool dataset, where Download selects them for transmission to an OnDemand server. After a report has been processed by the report distribution system, the output filename is no longer associated with the application program that created the input. That means the data loading program cannot use the filename (or the -A and -G parameters) to determine the application group and application to load. You must find some other way to identify the application group and application.

Download supports a user exit (APSUX15) that allows you to provide additional job information to OnDemand. Download includes the additional job information in the data stream that is transmitted from the spool dataset to the server. The *PSF for MVS: MVS Download Guide* and *PSF for OS/390:*

Download for OS/390 provide details about Download, the user exit, and the additional job information that can be included in the data transmitted to the server.

On the server, the arsjesd command provides the `-x` flag so that you can invoke a user-defined program to process the additional job information after successfully storing a transmitted file on the server. “Chapter 27. arsjesd command reference” on page 201 provides details about the arsjesd command.

Invoking a data download exit program

When Download selects output data from the spool dataset for transmission to an OnDemand server, it invokes the APSUX15 user exit program. The user exit program concatenates a string of additional job information to the print parameters that Download transmits to the server. Upon completion, the user exit program passes the location of the string and the string length to Download, which transmits the output dataset with associated JCL and the additional job information to the server.

The arsjesd command receives the dataset into a filesystem on the server. If the arsjesd command was started with the `-x` flag, it invokes the specified user-defined program. The program specified with the `-x` flag can be any user-defined program.

For example, you could write a program that parses the additional job information transmitted by Download and the APSUX15 user exit program. The user-defined program could extract the value of the `writer` parameter to identify the application to load. Using this value, the user-defined program could then query the database to determine the name of the application group that contains the application. The user-defined program could then call the arsload command with the -a parameter to identify the application to load and the -g parameter to identify the application group to load. (The user-defined program could also rename the input file and call the arsload command with the -A parameter to specify the part of the new filename that identifies the application and the -G parameter to specify the part of the new filename that identifies the application group.)

By using the Download user exit program, the `-x` flag with the arsjesd command, and a user-defined program, you can make sure that each file transmitted by Download is processed and loaded into the correct application and application group.

Please contact the IBM support center for more information about Download, data download exits, and creating user-defined programs to process reports before loading them.

Chapter 7. Maintaining migrated index data

Migration is the process by which OnDemand moves index data from the database to archive media. This process optimizes database storage space while allowing you to maintain index data for a very long time. You typically migrate index data after users no longer need to access the reports, but for legal or other business requirements, you still need to maintain the data for some number of years. If a user queries index data that has been migrated, an administrator must import a copy of the migrated table into the database. After maintaining the imported table in the database for a specified number of days, OnDemand expires it from the database.

This chapter provides information about importing index data into the database, including how to configure your system to migrate index data from the database to archive media, what happens when a user queries for migrated data, how to import the required index table, and what happens after you import a table into the database.

We assume that an experienced OnDemand administrator will use the information provided in this chapter. If you have questions about any of the following topics or if you would like help configuring your system to support migrating and importing of index data, please contact the IBM support center.

Configuring the system

Logging and routing messages

OnDemand provides the system logging facility so that administrators can monitor and manage the system. When you install and configure OnDemand, you initialize the system logging facility. The system logging facility is critical to the operation of the system.

When OnDemand processes a query for application group data requiring a table of index data that has been migrated to archive media, it writes an error message in the system log and sends the message to */dev/console* (UNIX) or the Event log (Windows NT) and to the system log user exit.

You can use a facility that examines messages and routes them to an administrator or a program, such as NetView. You can also configure the system log user exit to determine what action to take when a user queries migrated data.

Defining a system log user exit

Besides recording an "application group segment not available" error message in the system log, OnDemand sends the message to the system log user exit. If you have defined a system log user exit program, you can determine the action taken when OnDemand sends the message to the exit.

The program can take whatever action you deem necessary when the message occurs. For example, you may want to notify an administrator that a request for a migrated index table has occurred. Please refer to the *Installation and Configuration Guide* for information about programming a response to messages sent to the system log user exit.

Configuring ADSM

Before OnDemand can migrate index data, you must configure ADSM to maintain the data on archive media. You must define the library where ADSM will maintain the index data, define a device class for the library, and configure a policy domain, including a storage pool. You must make sure that ADSM maintains data stored in the domain indefinitely. If you need to maintain a backup copy of the index data, we recommend that you implement a copy storage pool in ADSM. The *Installation and Configuration Guide* provides details about defining a library to ADSM and implementing a copy storage pool.

Defining storage sets

You must assign the System Migration application group to a storage set that writes data to archive media. The storage set must identify a primary storage node. Use the OnDemand administrator interface to define storage sets and storage nodes and assign an application group to a storage set.

The primary storage node must identify a client node in ADSM. The client node must be registered in a domain that writes the migration data to archive media and maintains the data indefinitely. If you need to maintain a backup copy of the index data, we recommend that you configure a copy storage pool in ADSM. Refer to the *Installation and Configuration Guide* for information about ADSM domains, client nodes, and copy storage pools.

Configuring application groups

When you define an application group, you provide a set of storage management values that determine how long OnDemand maintains data stored in the application group and when OnDemand takes certain actions. For example:

- Life of Data and Indexes

Determines the length of time that OnDemand maintains index data and files stored in the application group.

- Migration of Indexes

  Determines the number of days before OnDemand moves index data from the database to archive media.

  Migrate index data after users no longer need to access the reports. Only in exceptional situations should users need to access data in migrated tables. If a user needs index data that has been migrated to archive media, the process of importing the table into the database requires administrator intervention and additional space in the database (and temporary storage) for imported index tables, and usually results in a significant delay in completing the query.

- Keep Imported Migrated Indexes

  Determines how long OnDemand maintains imported index data in the database before it is scheduled for deletion.

If you need to maintain index data on archive media, you must configure Migration of Indexes. If you want OnDemand to maintain imported index data in the database for longer than 30 days, you must provide a value for Keep Imported Migrated Indexes. OnDemand deletes imported index data after it resides in the database for the number of days specified for Keep Imported Migrated Indexes or the Life of Data and Indexes, whichever occurs first.

Use the OnDemand administrator interface to define application groups, including configuring storage management options to maintain index data.

What happens when a user queries migrated data

Message to the user

When OnDemand needs to access migrated index data to complete a query, it sends a message to the client program. The message states that the data required to complete the query is not available and the user should contact an administrator.

Message to the system log

When OnDemand needs to access migrated index data to complete a query, it writes a message to the system logging facility. You can open the System Log folder to display messages written to the system logging facility.

Message to the system

When OnDemand needs to access migrated index data to complete a query, it sends a message to */dev/console* (UNIX) or the Event log (Windows NT). You can use a facility that examines messages and routes them to an administrator or a program, such as NetView.

Message to the system log user exit

When OnDemand needs to access migrated index data to complete a query, it sends a message to the system log user exit. If you do not use some other facility that routes messages to an administrator or another program (such as NetView), we strongly encourage you to implement a user-defined program to process the message. The function of the program can vary, however at a minimum, we recommend that the program notifies a system administrator when a query for migrated index data occurs.

Importing the index data

Verify temporary work space

Importing migrated index data into the database requires temporary work space. Before you import the index data, you should verify that sufficient free space is available in the file system that OnDemand uses for temporary work file storage. The ARS_TMP environment variable (UNIX) determines where OnDemand stores temporary work files. The ARS_TMP environment variable is set in the ars.cfg file. For a Windows NT server, use the OnDemand configurator to define temporary work space file systems to OnDemand.

Verify database storage space

Importing migrated index data into the database requires additional database storage. Before you import the index data, you should verify that sufficient free space is available in the database directories.

Verify database log file space

Importing migrated index data into the database requires DB2 UDB log file storage. Before you import the index data, you should verify that sufficient free space is available in the archive log file directory. The ARS_ARCHIVE_LOGPATH environment variable (UNIX) determines the location of the archive log files. The ARS_ARCHIVE_LOGPATH environment variable is set in the ars.cfg file. For a Windows NT server, use the OnDemand configurator to define archive log file space to OnDemand.

Run the arsadmin command

OnDemand provides the arsadmin command to import a table of index data into the database. The following shows how to run the arsadmin command on a UNIX server (from the prompt) to copy a migrated index table from archive media into the database. The name of the application group and the index table to import can be obtained from the message written to the system logging facility:

```
arsadmin import -g 'Credit'-u admin -p "" SL27
```

To run the arsadmin command in Windows NT:

1. Start the OnDemand Command Line.
2. Enter the command and runtime parameters at the prompt. For example:

```
arsadmin import -g 'Credit' -u admin -p"" SL27
```

After the arsadmin command runs to completion, open the System Log folder to see messages generated by the import command.

Refer to "Chapter 22. arsadmin command reference" on page 153 for a description of the arsadmin command and details about parameters and options.

After you import index data

Querying data

After you import the required index data into the database, you should notify the user to re-enter the query.

Expiring imported migrated indexes

OnDemand flags an imported index table for deletion after it resides in the database for the number of days specified in the length of time to keep imported indexes field. After reaching this value, the next time that the arsmaint command runs, the data is deleted.

You typically configure the arsmaint command to run automatically on a regular schedule, using the cron facility (UNIX) or the OnDemand configurator (Windows NT). The *Installation and Configuration Guide* provides details.

If you need to run the arsmaint command on an exception basis, you can start it from the command line. "Chapter 29. arsmaint command reference" on page 213 provides details.

Configuring index migration

If you find that queries for migrated index data occur often, we recommend that you update the application group and increase the length of time that OnDemand maintains index data in the database. This should reduce the number of queries that need migrated index data.

Keeping imported migrated indexes

OnDemand deletes imported index data after it resides in the database for the number of days specified in Keep Imported Migrated Indexes or Life of Data and Indexes, whichever occurs first.

The default value for Keep Imported Migrated Indexes is 30 days. If you want OnDemand to maintain imported index data in the database for longer than 30 days, you must update the application group and change Keep Imported Migrated Indexes.

Chapter 8. Backup and recovery

Overview

This chapter describes backup and recovery for OnDemand and provides recommendations about methods and procedures that administrators can use to ensure the following critical OnDemand components can be recovered, if needed:

- OnDemand software
- OnDemand server information, created or modified during installation, configuration, and server operation
- The OnDemand database
- The ADSM database and other files critical to the operation of ADSM
- Archived reports and resources

Server software

If a media failure or some other unforeseen event occurs, you may be required to restore OnDemand and related software files, including the relational database management product you installed and ADSM. You can use the original product CD-ROMs to accomplish this task.

It is important that you store the original product CD-ROMs in a safe location. We recommend that you register OnDemand in your organization's business recovery plan and store the original product CD-ROMs with other vital information systems records.

Server information

When you installed and configured OnDemand, you specified information that customized OnDemand to operate in your environment. In UNIX, this information is stored in various control files. In Windows NT, this information is stored in the Registry. We recommend that you backup the control files or Registry immediately after you have verified the installation of OnDemand. In addition, if you periodically make changes to the OnDemand server information, we recommend that you backup the control files or Registry on a regular basis, perhaps once a day.

If you schedule regular backups of file systems, make sure that you backup the file systems where you installed the operating system, database manager, ADSM, and OnDemand.

You can use several different operating system commands to make backup copies of files and file systems and schedule the backups. For example, in UNIX, you can schedule the tar and mksysb commands with the cron facility. In Windows NT, you can use the NTBACKUP, REGBACK, and RDISK commands to backup the Registry and schedule the backup with the AT command. Refer to the documentation provided with the operating system for more information.

The OnDemand database

Note: The information in this section is specific to DB2. If you use a different database manager product, please refer to information provided with the product for details about database backup and recovery.

DB2 UDB table spaces

DB2 UDB table space support provides enhanced flexibility and improved performance for your application group data. For example, after you store a report in OnDemand, you can create a backup image of the table that changed during the load process, rather than creating a backup image of the entire database. You can also create an incremental backup image of the database, which contains only those tables that changed since the last backup image. Because the backup image only contains the changes made to the database, the backup process typically runs much faster than a full backup.

OnDemand creates one table space for each segment of application group data. After OnDemand closes the segment and you back up the table space, you do not need to back up the table space again, unless it is recovered or restored.

When you use the incremental table space backup capability, we recommend that you backup the OnDemand database after each report file load. If your schedule does not permit you to run the backup command after each report file load, we recommend that you backup the database once a day (assuming that you load multiple reports each day). While incremental backup images can be used to recover the database, we recommend that you periodically create a full backup image of the database. A full backup image of the database is the quickest way to recover the database in the event that you need to do so. However, if your OnDemand database is very large and cannot be backed up in a reasonable amount of time or requires many storage volumes to back up, you may find that maintaining full backup images of the database is not possible.

The *Installation and Configuration Guide* provides details about how to configure the system to support table spaces.

Database backup

OnDemand provides the arsdb command so that you can create backup images of the OnDemand database. The arsdb command supports incremental table space backups and full database backups, while other users are connected to the system or after stopping the OnDemand server programs and other related processes:

- An online backup can be taken when other applications or processes are connected to the database. That is, other applications and processes can continue to read or modify data while the backup is in process. An online backup can be scheduled with the cron facility (UNIX servers) or OnDemand configurator program (Windows NT servers).
- During an offline backup, only the backup task is connected to the database. Before starting an offline backup, we recommend that you stop the OnDemand server programs and other related processes to make sure that no users can connect to the database.

When you back up the database with the arsdb command, OnDemand removes the log files from the archived log file directory, releasing the space taken by files that are no longer needed. However, if you use ADSM to manage DB2 log files, the ADSM policy domain determines when archived log files are removed.

If your production schedule allows, we strongly encourage you to create offline backups on a regular schedule, perhaps once a week. Taking weekly offline backups can reduce the time required to rebuild table spaces or the database if you need to do so. We recommend that you write offline backup images to removable media or ADSM-managed storage. Keep backup images in a safe place until the next time that you create an offline backup image of the table space or database. If your schedule does not allow enough time to take offline backups (that is, your system must always remain available to users), you should take online backups on a regular schedule.

“Chapter 25. arsdb command reference” on page 173 shows examples of scheduling the database backup command and provides details about the parameters and options that support database backups.

See the *DB2 UDB Administration Guide* for details about backing up a database.

Using ADSM to maintain backup images

You can use ADSM to maintain DB2 UDB backup image files. This eliminates the need for you to maintain DB2 UDB backup image files on disk. ADSM can maintain the incremental table space backups and full database backups that you create with the arsdb command.

Before you can use ADSM to maintain DB2 UDB backup image files, you must define ADSM storage objects to maintain the DB2 files. The storage objects identify the type of media and storage devices that ADSM stores files on, determine the length of time that ADSM maintains files, and specify the number of backup copies that ADSM maintains. Before you schedule a backup command, you must make sure that ADSM storage volumes exist with sufficient free space to hold the backup image.

The *Installation and Configuration Guide* provides details about how to configure ADSM to maintain DB2 backup image files. See "Chapter 25. arsdb command reference" on page 173 for details about how to use the arsdb command to backup table spaces to ADSM-managed storage.

Backup to a tape device

If you plan to backup the database to a tape device you may need to configure the Buffer Size Limit in DB2. If you plan to backup the database to a tape device and specfiy a variable block size, you must configure the DB2 buffer size to a value that is less than or equal to the maximum block size limit for the backup device. For maximum performance, you should set the buffer size to the maximum block size for the backup device. See the DB2 UDB Version 5.2 documentation for details. Contact the IBM support center if you have questions.

Creating a full offline backup of the database

An offline backup of the database must be run at a time when no users are accessing the system and there are no other applications connected to the database. We recommend that you stop the OnDemand server programs and other related processes before starting the offline backup. After completing the backup command, you can restart the OnDemand server programs and other related processes.

If you plan to create the backup image on removeable media, such as tape, you must place a blank, formatted tape in the tape device before you start the backup command. If you are writing the backup image to a manual tape device, the backup image must fit on one tape volume.

The arsdb command writes processing messages to the system logging facility. After the backup completes, you can open the System Log folder to review messages generated by the backup command.

On UNIX servers, run the arsdb command from the command line to create a full offline backup of the database on tape. For example:

```
arsdb -v -y /dev/rmt0
```

To run the arsdb command on Windows NT servers:

1. Start the OnDemand Command Line.
2. Enter the command and runtime parameters at the prompt. For example:

```
arsdb -v -y \\.\Tape0
```

"Chapter 25. arsdb command reference" on page 173 provides details about the arsdb command, parameters, and options. The *DB2 UDB Administration Guide* provides details about backing up a database.

Database logging

DB2 UDB uses transaction logging to record changes to the OnDemand database. The information in the log file is used to recover from corruption of data in the database. Logging ensures that no data is lost. By combining the information in the log files with a backup copy of the database, the OnDemand database can be recovered to any point in time.

The OnDemand database and the DB2 UDB log files should reside on different physical volumes. The database backup image should be written to removable media. Unless multiple disk and tape volumes are damaged or lost **at the same time**, there is no possibility of losing the information contained in the OnDemand database.

The *DB2 UDB Administration Guide* provides details about database logs.

Using ADSM to maintain archived log files

You can use ADSM to maintain DB2 UDB archived log files. This eliminates the need for you to maintain the log files on disk.

Before you can use ADSM to maintain DB2 UDB log files, you must define ADSM storage objects to maintain the DB2 files. The storage objects identify the type of media and storage devices that ADSM stores files on, determine the length of time that ADSM maintains files, and specify the number of backup copies that ADSM maintains. Before DB2 UDB creates archived log files, you must make sure that ADSM storage volumes exist with sufficient free space to hold the files.

The *Installation and Configuration Guide* provides details about how to configure the system to use ADSM to maintain DB2 log files.

Database recovery

This topic provides an overview of the different recovery methods that can be used in the event there is a problem involving the database.

Note: We recommend that you contact OnDemand technical support to discuss the strategies available to you when there are problems with the database and help implement a backup and recovery plan that is best suited to your business and operating environment. The *DB2 UDB Administration Guide* provides details about database backup and recovery.

Typically you will need to recover the OnDemand database because of media and storage problems, power interruptions, and application failures. When a problem occurs that damages or corrupts the database in some way, you must rebuild the database. The rebuilding of the database is called recovery. There are two types of database recovery:

- The first type recovers from failures that occur while update transactions are taking place. For example, a system failure occurs while update transactions are taking place. The database is left in an inconsistent and unusable state and must be moved to a consistent and usable state before you can permit users to access the system.

  The transaction logs help correct this type of failure by allowing the transactions received before the failure to either be reapplied to the database or to be *rolled-out*. Rolling-out transactions is a way to return the database to the state it was in before the transaction that caused the failure.

  This type of recovery is done with the DB2 UDB RESTART DATABASE command. If you want this type of recovery to occur in every case of a failure, you can use the automatic restart enable database configuration parameter. The default for this configuration parameter is that the RESTART DATABASE routine will be started every time it is needed. Once enabled, you do not need to do anything to have this command done at the time of a failure.
- The second type of recovery deals with corruption of the OnDemand database and is usually caused by media failure. For example, one of the magnetic storage volumes that belongs to the database volume group becomes damaged and unusable. To recover from this type of failure, an administrator must intervene to recover the database.

  The combination of the DB2 UDB log files and a full backup copy of the database can be used to recreate the OnDemand database to any particular point in time. First, the latest full backup image of the database rebuilds the database to a point in time. Then, a roll-forward recovery restores all of the

units of work that occurred since the backup image was created. This allows you to restore the database to a state identical to the time of the failure.

The OnDemand database and the DB2 UDB log files should reside on different physical volumes. The database backup image should be written to removable media. Unless multiple disk and tape volumes are damaged or lost **at the same time**, there is no possibility of losing the information contained in the OnDemand database.

Factors affecting recovery

To decide which database recovery method to use, you must consider the following key factors:

- How near to the time of failure you will need to recover the database?

  When you restore a full backup copy of the database, the database is only as current as the time that the last backup was made.

  To restore the database to the time of a failure, you must use the log files to reapply changes that were made to the database since the backup copy was created. You can reapply the changes to the end of the logs or to a point in time. A point in time recovery may be useful if an application corrupts the database and you do not want to reapply its changes.

- How much time is spent associated with recovery?

  Your recovery plan should allow for regularly scheduled backups, since backing up the database requires time and system resource.

  You can do a backup while the database is either online or offline. If it is online, users can access the system and other processes can connect to the database and read and modify data while the backup task is running. If the backup is performed offline, only the backup task can be connected to the database. Users cannot access the system and other processes cannot connect to the database while the offline backup task is running.

- How much storage space can be allocated for backup copies and archived log files?

  To restore the database, you must allocate enough free disk space to hold the backup copy of the database and the restored database. To roll-forward transactions requires space to hold the backup copy of the database, the restored database, and all of the archived log files created between backup copies of the database.

- Table space level or full database level backup.

  With a table space backup, you can specify one or more tables spaces to be backed up, rather than the entire database. You can then restore selected table spaces to a state identical to the time the backup was made. However, those table spaces not selected at the time of the backup will not be in the same state as those that were restored.

We strongly encourage you to speak with OnDemand technical support before you implement a backup and recovery plan that includes table space backup and recovery.

The *DB2 UDB Administration Guide* provides details about recovering a database.

The ADSM database

ADSM maintains a database that contains information about the devices and files it manages. Whenever the system maintains data on archive media, ADSM updates its database with the status of the storage it manages. When you define devices and register client nodes, ADSM updates the database. When ADSM maintains storage volumes, it updates the database with status information about files and storage volumes. The ADSM database is critical to proper operation of the system in storing objects on and retrieving objects from archive media.

Mirroring the ADSM database is strongly encouraged. When you mirror the database, ADSM replicates the database onto different physical storage. ADSM automatically keeps track of and refreshes both copies of the database. When you configure physical storage so that ADSM can mirror the database on different physical devices and adapters, you can provide protection for the database due to failure of a single device. With mirroring, ADSM can continue operation without interruption if a database volume fails by using a mirrored copy of the failed volume. Mirroring requires additional storage space for the mirrored volumes. The *ADSM Administrator's Guide* provides details about mirroring the ADSM database.

To protect the information in the database, and ensure that it can be restored if a disaster occurs, you must periodically create a backup copy of the database. You can recover the database to its most current state or to a specific point in time with the backup copy:

- A full backup image of the database should be taken after you install and configure ADSM with OnDemand. In addition, we recommend that you create a full backup of the database on a regular schedule, such as once a week. A full backup copy of the database should be written to removable media.
- An incremental backup image of the database records changes that occurred since the last backup of the database (full or incremental). You can create a maximum of 32 incremental backups between full backups of the database. If you write incremental backup images of the database to disk, make sure that the disk is on a different controller than any of the database or recovery log volumes.

There are several factors to consider when you decide the type and frequency of backups:

- A full backup takes longer to run than an incremental backup.
- Recovery time is faster with a full backup. Incremental backups increase the time it takes to recover the database because a full backup must be loaded first, followed by some or all of the incremental backups.
- A full backup is required under specific conditions. You must create a full backup after installing and customizing ADSM with OnDemand. You can run up to 32 incremental backups between each full backup.

We recommend that you back up the database after you load report files into OnDemand and following ADSM maintenance of storage volumes (expiration and reclamation). Under typical conditions, you should back up the database each day.

ADSM includes a central scheduling component that allows the automatic processing of administrative commands, such as database backup. Each administrative command is called an event. Each scheduled event is tracked by the server and recorded in the database. You set up an administrative command schedule by defining schedule parameters, such as the start day, date, and time, specifying the command to be executed, such as the BACKUP DATABASE command, and activating the schedule. The *ADSM Administrator's Guide* provides details about scheduling operations.

For UNIX servers, OnDemand provides the ars_adsm command to create a full backup of the database. You can submit the command using the *at* command or run the command automatically on a regular schedule using the cron facility. If you invoke the arsload command from a shell script, you can add a step that runs the ars_adsm command after the arsload command completes. The *Installation and Configuration Guide* describes how to configure the server to start the ars_adsm command and create a full backup image of the database automatically on a regular schedule. "Chapter 23. ars_adsm command reference" on page 165 provides details about the ars_adsm command, parameters, and options.

For Windows NT servers, you can backup the database from the ADSM Admin Command Line interface. You can also define a backup schedule to ADSM. Refer to the ADSM documentation for more information.

ADSM recovery log

The recovery log is critical to the operation of the ADSM server. If the recovery log is unusable, the entire server is unavailable. With the recovery log available, and a restored backup image of the database, you can recover the database to its most current state.

To ensure fast recovery time and high availability of the database, you should always mirror the recovery log. Mirroring the recovery log requires much less space than mirroring the database. If you do not mirror the recovery log, you should allocate the recovery log on a disk other than the one on which the database resides. The *ADSM Administrator's Guide* provides details about mirroring the ADSM recovery log.

When a database backup completes, ADSM deletes recovery log records preceding the backup, and releases recovery log storage for reuse. Taking frequent database backups reduces recovery log storage requirements and the time required to recover the database.

ADSM volume history file

Volume history information is vital for recovery of a lost or damaged database. The volume history file contains information that ADSM needs about the volumes to use for database backups. The volume history file contains information that you need to know about which storage pool volumes to audit after a recovery.

ADSM cannot obtain volume history information from the database during a restore of the database. Therefore, you should store at least one backup copy of the volume history file on a disk other than the one on which the database resides.

The *ADSM Administrator's Guide* provides details about creating volume history backup files.

ADSM device configuration file

When you define, update, or delete a device class, drive, or library, ADSM updates the database and makes an entry in the device configuration file. To restore the database, ADSM requires a definition for the device class from which backup data is to be read. This definition is in the device configuration file.

When the database is being restored, no definitions can be read from the database. Therefore, you should have at least one backup copy of the device configuration information on a disk other than the one on which the database resides.

The *ADSM Administrator's Guide* provides details about creating device configuration backup files.

ADSM database recovery

Recovering using mirrored copies of the database

If a database volume fails because of media failure and you have enabled mirroring, you can recover the database by using mirrored copies of the database. After fixing the failing device, you can allocate space for the new mirrored copy and define the volume to ADSM. After you define the volume to ADSM, the server synchronizes the volume with the database.

Recovering using backup copies of the database

ADSM provides commands to recover the database should a catastrophic failure occur. These commands restore the database from the latest available full backup copy, apply all incremental backups that apply, and use the recovery log to apply any changes made to the database since the last backup was created.

If you restore the database to its most current state, ADSM automatically synchronizes the database and storage volumes.

If you restore the database to a specific point in time, you must audit all storage pool volumes to check for and resolve any inconsistencies between database information and storage pool volumes. Depending on the number of storage pool volumes and the amount of activity that occurred after the database backup that you restored, the audit may require a significant amount of time.

To perform a database recovery, you should have the following information, preferably stored offsite:

- Back up volumes of the database
- Copy storage pool volumes
- Server options file
- Volume history file
- Device configuration file
- Output from ADSM commands that provide details of the database and recovery log setup

The *ADSM Administrator's Guide* provides details about recovering the ADSM database.

Archived reports and resources

OnDemand can store copies of reports and resources in the document cache and on archive media. Most customers copy reports to cache storage and archive media at the same time, when they load a report into the system.

The primary purpose of the document cache is short-term, high-speed storage and retrieval. The document cache consists of magnetic storage volumes maintained by OnDemand on one or more object servers.

The primary purpose of archive media is long-term storage and retrieval. Reports on archive media can also be used as backup copies in the event cache storage becomes corrupted or unavailable. Archive media consists of optical or tape storage volumes managed by ADSM.

OnDemand can retrieve a copy of a report from archive media after the report has been deleted from cache storage or if the copy on cache storage is unavailable. However, you must properly configure your system to support multiple copies of reports. You must install and configure ADSM, add archive media devices to your system, and configure OnDemand to use archive media. You configure OnDemand to use archive media by defining client nodes to ADSM, assigning application groups to the appropriate storage sets, and setting application group data migration and caching information.

Note: If you do not copy reports to archive media, we recommend that you use ADSM to protect the cache file systems. ADSM can provide a range of storage management services. You can use ADSM to make backup copies of cache file systems. If a cache storage device fails, you can use ADSM to restore all or part of the cache file system. However, without a copy of reports on archive media, users cannot retrieve reports until cache storage is restored.

The document cache

The document cache is the primary, short-term storage location for reports.

If you do not copy reports to archive media when you store them in OnDemand, you need to consider how you can recover the reports in the event you need to do so (for example, if a cache storage device fails).

The document cache can be protected by maintaining it on RAID storage subsystems. RAID storage can provide excellent availability, allowing users to access reports even if a disk or controller fails. The *Introduction and Planning Guide* provides information about using RAID storage with OnDemand. However, RAID storage is not fail safe. There may be situations when,

because of multiple disk or controller failures, users cannot access reports. We encourage you to use ADSM to maintain a copy of reports on archive media.

Reports on archive media

The primary storage node identifies the object server and ADSM client node where OnDemand stores the primary copy of a report on archive media. OnDemand retrieves the primary copy of the report from archive media after the report has been deleted from cache storage. Customers with critical, high availability requirements may need to create a backup or secondary copy of the report on archive media. The backup or secondary copy can be used if the primary copy becomes corrupted or unavailable.

There are two methods that you can use to create a backup or secondary copy of reports:

- Define a *copy storage pool* to ADSM. With this method, ADSM manages a backup copy of files that are stored in a primary storage pool independently and transparently to OnDemand. The backup copy is stored in a copy storage pool that can be used to restore the original files if they become damaged, lost, or unusable. The copy storage pool can be assigned to the same library as the primary storage pool. However, you would typically assign the copy storage pool to a different library. You can copy data from one or more primary storage pools to the same copy storage pool. Copy storage pools require additional space in the ADSM database. A copy storage pool must reside on the object server where the primary storage pool resides.

  ADSM includes a central scheduling component that allows the automatic processing of administrative commands, such as copying data from a primary storage pool to a copy storage pool. Each administrative command is called an event. Each scheduled event is tracked by the server and recorded in the database. You set up an administrative command schedule by defining schedule parameters, such as the start day, date, and time, specifying the command to be executed, such as the BACKUP STGPOOL command, and activating the schedule.

  The *ADSM Administrator's Guide* provides details about defining and managing a copy storage pool, storage pool backup and recovery, and scheduling operations.
- Assign the primary storage node to a secondary storage node. With this method, ADSM maintains a secondary copy of files that are stored in a primary storage node. The secondary copy can be used if the primary copy becomes damaged, lost, or unusable. However, administrator intervention is required before OnDemand can use the secondary copy. For this reason, we discourage the use of secondary storage nodes.

Part 2. Managing ADSM storage

This part of the book provides information about tasks required to maintain ADSM storage, such as preparing storage volumes, managing storage volumes in an automated library, managing media mount operations, automating operations, and registering client nodes.

Topic	Page
Overview of managing ADSM storage	67
Starting, stopping and halting the server	75
Using scratch and private volumes	77
Labeling storage volumes	79
Adding storage volumes to a library	81
Removing storage volumes from a library	83

Chapter 9. Overview

This part of the book provides a brief overview of tasks that ADSM administrators do to manage ADSM storage. It also points to the sections in the ADSM Administrator's Guide that present the details of those tasks and the concepts you need to understand to complete them. The *Installation and Configuration Guide* is another important source of information. When you installed and configured an OnDemand system with ADSM, you completed many of the tasks described here: configuring devices and defining them to ADSM, defining policies, preparing storage volumes, registering client nodes, and increasing the size of the database and recovery log.

The following topics provide more information:

- Using magnetic disk devices with ADSM
- Using removable media devices with ADSM
- Managing removable media operations
- Defining drives and libraries
- Defining device classes
- Managing storage pools
- Managing storage pool volumes
- Managing policies
- Managing client nodes
- Automating server operations
- Managing server operations
- Managing the database and recovery log
- Monitoring the server
- Protecting and recovering your data

Using magnetic disk devices

In an OnDemand system, the primary use of magnetic disk devices with ADSM is the storage of the database and recovery log. The ADSM database contains information needed for server operations and information about data that has been backed up, archived, and space-managed. The database contains pointers to the locations of all client files in the ADSM storage pools. Changes to the database are recorded in the recovery log in order to maintain a consistent database image. The recovery log contains information about updates that have not yet been committed to the database. If the database is unusable, the entire ADSM server is unavailable. If a database is lost and

cannot be recovered, the backup, archive, and space-managed data for that server is lost. Refer to “Managing the database and recovery log” on page 72 and “Protecting and recovering data” on page 73 for steps that you can take to protect your database.

Using removable media devices

In an OnDemand system, removable media devices can be used with ADSM for the following purposes:

- Storage of application group data, including migrated index data. Application group data is typically stored in optical libraries, but can also be stored in automated tape libraries.
- Storage of ADSM database backups. Database backups are typically stored on manually operated devices, such as an 8mm tape drive, but can also be stored in optical or automated tape libraries.
- Storage of DB2 archived log files and backup image files. DB2 files must be stored on rewriteable optical media (not WORM) or tape.

ADSM allows you to use and reuse removable media to store data. You must prepare removable media for initial use by ADSM. You also control how and when media are reused. The *Installation and Configuration Guide* shows examples of labeling removable media for initial use and checking storage volumes into a library. For detailed guidance and scenarios on configuring your removable media devices, refer to the ADSM Administrator’s Guide.

Managing removable media operations

ADSM allows you to use and reuse removable media to store data. You must prepare removable media for initial use by ADSM. You also control how and when media are reused.

Volumes must be mounted in response to mount requests from ADSM. For manual libraries, you can monitor the mount requests by using an administrative client in mount mode or console mode. Someone you designate as the operator must respond to the mount requests by putting in tape volumes as requested. For devices in automated libraries, ADSM interacts with the library to mount volumes, but sends messages when the library needs attention from an operator. ADSM also tracks the inventory of media in each automated library.

For automated libraries, ADSM works with the operating system and the library to accomplish volume mounts. Mount messages are not sent to an operator. However, information about problems with the library are still sent

to the mount message queue. You can see these messages on administrative clients that have been started with either the mount mode or console mode parameter. However, you cannot use the ADSM REPLY command to respond to these messages. You can get information about pending operator requests either by using the QUERY REQUEST command or by checking the mount message queue on an administrative client started in mount mode.

In many cases, an operator request has a time limit. If the requested action is not performed within the time limit, the operation times out and fails.

For most types of requests, such as volume mounts, the server detects when the operator performs the action. The operator does not usually need to respond to the ADSM server carrying out the requested activity. However, sometimes the server cannot detect the completion of the requested action. When the server requires a reply, the message that is displayed by the server requests that the operator reply when the activity has been completed. For example, a request to mount a scratch volume requires that the operator reply when a scratch volume has been placed in the drive. ADSM waits for a reply to prevent the use of the wrong volume.

For information about managing removable media operations, refer to the ADSM Administrator's Guide.

Defining drives and libraries

To use removable media devices with ADSM, you must define the libraries and drives to ADSM.

The *Installation and Configuration Guide* provides examples of defining drives and libraries. For detailed information about defining drives and libraries, refer to the ADSM Administrator's Guide.

Defining device classes

A device class represents a set of storage devices with similar availability, performance, and storage characteristics. You must define devices classes for the types of drives available to an ADSM server. You specify a device class when you define a storage pool, which is a named collection of volumes for storing data.

The *Installation and Configuration Guide* provides examples of defining device classes. Refer to the ADSM Administrator's Guide for detailed information about device classes.

Managing storage pools

OnDemand data is stored in groups of volumes called storage pools. The data on these primary storage pools can be backed up to copy storage pools for disaster recovery purposes. Because each storage pool is assigned to a device class, you can logically group your storage devices to meet your storage management needs.

The *Installation and Configuration Guide* provides examples of defining primary storage pools and copy storage pools. For more information about storage pools, refer to the ADSM Administrator's Guide.

Managing storage pool volumes

You manage storage volumes by defining, updating, and deleting volumes, and by monitoring the use of server storage. Monitoring volumes can reveal inconsistencies between information in the database and client node files in storage pools. You can also move files within and across storage pools to optimize the use of server storage.

For each automated library, ADSM tracks in its volume inventory for the library whether a volume has scratch or private status:

- A scratch volume is a labeled volume that is empty or contains no valid data, and can be used to satisfy any request to mount a scratch volume. To support OnDemand, you typically define scratch volumes to ADSM. ADSM uses scratch volumes as needed, and returns the volumes to scratch when they become empty (for example, when all data on the volume expires).
- A private volume is a volume that is in use or owned by an application, and may contain valid data. Volumes that you define to ADSM are private volumes. A private volume is used to satisfy only a request to mount that volume by name. When ADSM uses a scratch volume, it changes the volume's status to private by defining it. ADSM tracks whether defined volumes were originally scratch volumes. Volumes that were originally scratch volumes return to scratch status when they become empty.

In addition to preparing removable media for ADSM, you need to maintain a supply of scratch volumes and manage the volume inventory in an automated library. Managing a library may mean that you need to remove volumes from a library, return volumes to a library, and manage a full library. Other chapters in this part of the book provide examples of preparing storage volumes and adding storage volumes to and removing storage volumes from automated libraries. For details about these tasks, refer to the ADSM Administrator's Guide.

Managing policies

OnDemand documents, application group index data, and DB2 files can be backed up to the server. This process ensures that the information can be retrieved when needed. Recall of documents and DB2 files is transparent and automatic when a client retrieves a document or DB2 needs to retrieve a backup image file or archived log file to restore the database. Importing migrated index data requires administrator intervention.

You define policies based on your requirements for archiving, backing up, or migrating data. You do this by defining policy objects, which identify archive, backup, and migration criteria, and by scheduling client operations.

The *Installation and Configuration Guide* provides examples of defining policies to support archiving documents, backing up DB2 files, and migrating index data. Refer to the ADSM Administrator's Guide for more information about establishing and managing policies.

Managing client nodes

You register OnDemand primary storage nodes as client nodes in ADSM. You provide client/server authentication by requiring the use of passwords to ensure that the client and the server are authorized to communicate with each other. You can also set the length of passwords and determine when passwords expire.

You can define sets of client options for clients to use. For example, you typically define one set of client options for OnDemand application group data and another set of client options for DB2 files.

You can control access to the server by administrators. An organization may name a single administrator or may distribute the workload among a number of administrators and grant them different levels of authority.

The *Installation and Configuration Guide* provides examples of registering client nodes, defining client options for OnDemand primary storage nodes, and registering administrators. Refer to the ADSM Administrator's Guide for more information about managing clients.

Automating server operations

You can define schedules for the automatic processing of most administrative commands, such as backing up primary storage pool data to a copy storage pool and backing up the database.

The *Installation and Configuration Guide* provides an example of scheduling the backup of a primary storage pool to a copy storage pool. Refer to the ADSM Administrator's Guide for more information about scheduling ADSM commands and operations.

Managing server operations

You can manage server operations such as starting and stopping the server, maintaining and suspending client sessions with the server, and controlling server processes.

In an OnDemand system, after you initially set up the ADSM Server and Scheduler services to start automatically and define schedules for specific server operations (such as backing up the database and copying data from primary storage pools to copy storage pools), there is very little you need to do to manage the server operations on a day-to-day basis.

Refer to the ADSM Administrator's Guide for details about the day-to-day tasks involved in administering the server and about reports and information available to help you manage the server.

Managing the database and recovery log

The ADSM database contains information about OnDemand data in storage pools, registered client nodes, ADSM policies, and ADSM schedules. The server recovery log, which records changes made to the database, is used to restore the database to a consistent state and to maintain consistency across server start-up operations.

After your system is operational, you should monitor the database and recovery log to see if you should add space. You can reset the maximum utilization counters for the database and recovery log to monitor daily utilization. To set the maximum utilization percentage equal to the current utilization, you might want to reset the utilization statistics each day. The *Installation and Configuration Guide* shows how to increase the size of the database and the recovery log. Refer to the ADSM Administrator's Guide for information about monitoring the database and recovery log.

Monitoring the server

ADSM provides you with many sources of information about server and client status and activity, the state of the database, and resource usage. By monitoring this information, you can provide reliable services to users while making the best use of available resources.

You can use ADSM queries and SQL queries to get information about the server. You can also set up logging of information about ADSM clients and server events. Refer to the ADSM Administrator's Guide for more information about these tasks.

Protecting and recovering data

ADSM provides a number of ways to protect and recover your data from media failure or from the loss of the ADSM database or storage pools due to a disaster. These recovery methods are based on the following preventive measures:

- Mirroring, by which the server maintains one or more copies of the database or recovery log, allowing the system to continue when one of the mirrored disks fails. We recommend that you mirror at least one copy of the database and the recovery log to different physical storage volumes.
- Periodic backup of the database. We recommend that you schedule frequent backups of the database, after every load or system configuration change, or once a day. If you do not load reports every day or your system configuration does not change very often, you may be able to schedule backups less frequently, perhaps once a week.
- Periodic backup of the storage pools. To protect OnDemand data stored in ADSM, you may want to backup data in a primary storage pool to a copy storage pool. The *Installation and Configuration Guide* shows how to configure a copy storage pool.
- Recovery of damaged files.

Disaster Recovery Manager (DRM) is an optional feature of ADSM Version 3 that assists an administrator with preparing a disaster recovery plan. The disaster recovery plan can be used to guide an administrator through disaster recovery as well as for audit procedures to certify the recoverability of the ADSM server. DRM's disaster recovery methods are based on the following measures:

- Enabling Disaster Recovery Manager
- Creating a backup copy of server primary storage pools and database
- Sending server backup volumes offsite
- Moving reclaimed or expired volumes back onsite
- Create the ADSM server disaster recovery plan file
- Storing client machine information
- Defining and tracking client recovery media

For more information about protecting your data and for details about recovering from a disaster, see the ADSM Administrator's Guide.

Chapter 10. Starting, halting, and restarting the server

ADSM administrators can manage server operations. These operations include such tasks as starting and halting the server, adding or updating server options, defining devices and policies, managing removable media, and monitoring server information.

Starting the server

To start the server, complete the following steps:

1. Change to the ADSM server program directory.
2. Start the server:

   ```
   dsmserv
   ```

   When the server is started, ADSM displays information about product licensing, server options, the database and recovery log, storage pools, and progress messages and any errors encountered during server initialization.

Note: You can capture ADSM server console messages to a user log file with the ADSM dsmulog utility. Refer to the ADSM documentation for more information.

Starting the ADSM server command line interface

In this part of the book, most examples illustrate how to perform tasks by using the ADSM server command line interface. To start the ADSM server command line interface, type the following command:

```
dsmadmc
```

ADSM provides you with ways to monitor processes and messages:

- Use the console mode from an administrative client to monitor processes and messages:

  ```
  dsmadmc -consolemode
  ```

  While the system is running in console mode, you cannot enter any administrative commands from the client session. You can, however, start another administrative client session for entering commands.
- Specify the OUTFILE option to write all terminal output to a file. For example:

  ```
  dsmadmc -consolemode -outfile=adsm.out
  ```

- From the command line interface, query the activity log for status information and possible error messages:

  ```
  query actlog
  ```

Refer to the ADSM documentation for more information about managing client sessions.

Halting the server

When you halt the server, all processes are abruptly stopped and client sessions are canceled, even if they are not completed. Any in-progress transactions are rolled back when the server is restarted. When the server is halted, administrator activity is not possible. If possible, halt the server only after current administrative and client node sessions have completed or canceled. To shut down the server without severely impacting administrative and client node activity with the server, follow the instructions in the ADSM documentation.

To halt the server and shut down all server operations, enter `halt` at the ADSM Server command line interface.

Restarting the server

To start the server after it has been halted, follow the instructions in “Starting the server” on page 75.

When you restart the server after it has been halted, ADSM rolls back any operations that had been in process to ensure that the database remains in a consistent state.

Chapter 11. Using scratch and private volumes

A scratch volume is a labeled volume that is empty or contains no valid data, and can be used to satisfy any request to mount a scratch volume. A private volume is a volume that is in use or owned by an application, and may contain valid data. Volumes that you define to ADSM are private volumes. A private volume is used to satisfy only a request to mount that volume by name. For each storage pool, you must decide whether to use scratch volumes.

If you use scratch volumes, ADSM uses volumes as needed, and returns the volumes to scratch when they become empty (for example, when all of the data on the volume expires). If you do not use scratch volumes, you must define each volume you want ADSM to use. Volumes that you define to ADSM are private volumes, and do not return to scratch when they become empty.

For each automated library, ADSM tracks in its volume inventory for the library whether a volume has a scratch or private status. If you allow scratch volumes to be used for a storage pool, ADSM chooses a scratch volume from the scratch volumes that are checked in for the library.

When ADSM uses a scratch volume, ADSM changes the volume's status to private by defining it. ADSM tracks whether defined volumes were originally scratch volumes. Volumes that were originally scratch volumes return to scratch status when they become empty.

One of the benefits of using scratch volumes is that different storage pools that share the same automated library can dynamically acquire volumes from the library's pool of scratch volumes. The volumes need not be preallocated to the different storage pools.

Another benefit of using scratch volumes, even if only a single storage pool is associated with an automated library, is that you need not explicitly define all of the volumes for the storage pool using DEFINE VOLUME commands. Volumes are automatically added to and deleted from the storage pool by the server.

If a scratch volume is used for an ADSM database backup or export operation, ADSM changes the volume's status to private. The volume returns to the scratch pool only when an ADSM administrator determines that the volume's data is no longer needed, and uses the UPDATE LIBVOL command to change the status of the volume to scratch.

Chapter 12. Labeling storage volumes

All removable media must be labeled before it can be used by ADSM. When the server accesses a volume, it checks the volume name in the header to make sure that the correct volume is being used. Any tape storage volumes must be labeled before the server can use them.

For storage pools in automated libraries, use the CHECKIN LIBVOL command to check labeled volumes into a library.

Use the LABEL LIBVOL command with drives in an automated library to label and check in the volumes with one command. To use the LABEL LIBVOL command, there must be a drive that is not in use by another ADSM process. This includes volumes that are mounted but idle. If necessary, use the DISMOUNT VOLUME command to dismount the idle volume to make that drive available.

Overwriting volume labels

By default, the label command does not overwrite an existing label on a volume. However, if you want to overwrite existing volume labels, you can specify the OVERWRITE=YES parameter when you use the LABEL LIBVOL command.

Important: By overwriting a volume label, you destroy all of the data that resides on the volume. Use caution when overwriting volume labels to avoid destroying important data.

Identifying volume labels

Use the LABEL LIBVOL command to specify the volume for labeling. You can use the VOLRANGE parameter of the LABEL command for a large number of volumes. For automated libraries, you are prompted to mount the volume in the entry/exit slot of the library. If no entry/exit slot is available, mount the volume in an empty slot within the library. For manual libraries, you are prompted to load the volume directly into the library.

Labeling volumes one at a time

The LABEL LIBVOL command assumes that you will insert volumes into the library when prompted to do so. The label process then mounts each inserted volume into a drive and writes a label to it using a name that you enter at a prompt. This is the default mode of operation when you specify a library for use with the LABEL LIBVOL command.

If the library does not have an entry/exit port, you are prompted to remove the volume from a specified slot number. If the library has an entry/exit port, the command by default returns each labeled volume to the entry/exit port of the library.

To label one volume in the `odlib0` library, enter:

```
label libvol odlib0 od5000 search=no
```

Where `od5000` is the volume label.

Searching the library

The LABEL LIBVOL command searches all of the storage slots in the library for volumes and tries to label each one that it finds. You choose this mode when you specify the SEARCH parameter. After a volume is labeled, the volume is returned to its original location in the library.

For an automated SCSI library, you can simply open the library access door, place all of the new volumes in unused storage slots, close the door, and issue the LABEL LIBVOL command with the SEARCH=YES parameter.

To label all of the volumes in the `odlib0` library by searching the library and specifying a range of volume labels, enter:

```
label libvol odlib0 search=yes volrange=od5000
```

ADSM labels the first volume `od5000` and increments the number `5000` for the label of each additional volume you label with this command.

Chapter 13. Adding storage volumes

You inform the server that a new volume is available in an automated library by checking in the volume with the CHECKIN LIBVOL command. When a volume is checked in, the server adds the volume to its library volume inventory. You can also use the LABEL LIBVOL command to label and check in volumes in one operation.

Because the CHECKIN LIBVOL command involves device access, it may take a long time to complete.

When you check in a volume, you must supply the name of the library and the status of the volume (private or scratch).

To check in one or just a few volumes, you can specify the name of the volume with the command, and issue the command for each volume.

To check in a larger number of volumes, you can use the search capability of the CHECKIN command or you can use the VOLRANGE parameter of the CHECKIN command.

Checking in volumes one at a time

Use this option if you want to check in only a single volume that is not currently in the library. ADSM requests that the volume be placed in the entry/exit port of the library.

If the library does not have an entry/exit port, ADSM requests that the volume be loaded into a slot within the library. The mount request specifies the location with an element address. Element addresses are listed on the device configuration worksheets for the libraries supported by ADSM. See the *ADSM Administrator's Guide* or the documentation provided with the library.

To check in a private volume:

```
checkin libvol odlib0 od1092 status=private search=no
```

Where `odlib0` is the name of the library and `od1092` is the volume label.

To check in a scratch volume:

```
checkin libvol odlib0 od2001 status=scratch search=no
```

Where `odlib0` is the name of the library and `od2001` is the volume label.

Searching the library

Use this option if you want ADSM to automatically search the library for new volumes that have not already been added to the library volume inventory. Use this mode when you have a large number of volumes to check in, and you want to avoid issuing an explicit CHECKIN LIBVOL command for each volume.

With this option, you cannot specify a volume name because the server searches for new volumes in the library.

For an automated SCSI library, you can simply open the library access door, place all of the new volumes in unused storage slots, close the door, and issue the CHECKIN LIBVOL command with SEARCH=YES. For example:

```
checkin libvol odlib0 search=yes status=scratch
```

Where `odlib0` is the name of the library. ADSM will label the new volumes, check them into the library, and add them to the library volume inventory.

Allowing swapping of volumes

If no empty slots are available in the library when you are checking in volumes, the check-in fails unless you allow swapping. If you allow swapping and the library is full, ADSM selects a volume to eject before checking in the volume you requested.

ADSM selects the volume to eject by checking first for any available scratch volume, then for the least frequently mounted volume.

Chapter 14. Removing storage volumes

ADSM tracks the scratch and private volumes available in an automated library through a library volume inventory. ADSM maintains an inventory for each automated library. The library volume inventory is separate from the inventory of volumes for each storage pool. To add a volume to a library's volume inventory, you check in a volume to that library.

To make sure that ADSM's library volume inventory remains accurate, you must check out volumes when you need to physically remove them from a SCSI or 3494 library device. When you check out a volume that is being used by a storage pool, the volume remains in the storage pool. If ADSM requires the volume to be mounted while it is checked out, a message to mount the volume is displayed with a request to check in the volume. If the check in is not successful, ADSM marks the volume as unavailable.

To check whether ADSM's library volume inventory is consistent with the volumes that are physically in the library, you can audit the library. The inventory can become inaccurate if volumes are moved in and out of the library without informing the server (by using check-in and check-out commands).

Removing volumes from a library

You may need to remove a volume from a library because all of the volumes in the library are full, and you want to remove some that are not likely to be accessed in order to make room for new volumes that can be used to store more data.

To remove a volume from an automated library, use the CHECKOUT LIBVOL command. By default, the server mounts the volume being checked out and verifies the internal label. When the label is verified, the server removes the volume from the library volume inventory, and then moves it to the entry/exit port of the library. If the library does not have an entry/exit port, ADSM requests that the volume be removed from a slot within the library.

If you check out a volume that is defined in a storage pool, the server may attempt to access it later to read or write data. If this happens, the server requests that the volume be checked in.

Returning volumes to a library

When you check out a volume that is defined to a storage pool, to make the volume available again, you must do the following:

1. Check in the volume for the library, with private status. Use the CHECKIN LIBVOL command with the STATUS=PRIVATE parameter.
2. Update the volume's ACCESS value. You must change the access from unavailable to read/write or read-only. Use the UPDATE VOL command with the ACCESS parameter.

Managing a full library

As ADSM fills volumes in a storage pool, the number of volumes needed for the pool may exceed the physical capacity of the automated library. To make room for new volumes while keeping track of existing volumes, you can define an overflow location near the library for the storage pool. You then remove media to the overflow location as needed.

The following shows a typical sequence of steps to manage a full library:

1. Define or update the storage pool associated with the automated library, including the overflow location parameter. For example, you have a storage pool named odaix1 associated with an automated library. To update the storage pool to add an overflow location of room2948, enter:

   ```
   update stgpool odaix1 ovflocation=room2948
   ```

2. When the library becomes full, use the MOVE MEDIA command.

   ADSM records the location of the volumes that you move with the MOVE MEDIA command. The location of the volumes is the overflow location that you defined for the storage pool. For example, to move all full volumes in the specified storage pool out of the library, enter:

   ```
   move media * stgpool=odaix1
   ```

3. Check in new scratch volumes, if needed.
4. As requested through ADSM mount messages, check in volumes that ADSM needs for operations. The mount messages include the overflow location of the volumes.

To find the location of volumes in a storage pool that has an overflow location, you can use the QUERY MEDIA command. You can also use the QUERY MEDIA command to generate the commands required to check in all of the volumes into the library.

Offline storage of storage volumes

Refer to the documentation provided by the library manufacturer for instructions that describe how to handle physical storage volumes and remove them from the library.

Refer to your organization's media storage guide for instructions about documentation you may need to complete when you remove a storage volume from a library and where to store them for safekeeping.

Part 3. Operating UNIX Servers

This section of the book describes day-to-day, operational tasks that you may need to perform when supporting a UNIX server. These tasks include starting and stopping server programs and related processes and monitoring important information about the operation of the server and server resources, such as processes and system load.

Topic	Page
Starting, stopping, and restarting processes	89
Monitoring the server	95

Chapter 15. Starting and stopping processes

Starting OnDemand

When you power-on the workstation, the operating system can automatically start the OnDemand server program and other related OnDemand processes as part of system initialization. If you followed the instructions in the *Installation and Configuration Guide*, you configured the init facility with information required to start the database, ADSM, Download, the arsload command, and the OnDemand server program during system initialization.

During normal system operation, the operating system can start programs at regularly scheduled intervals. If you followed the instructions in the *Installation and Configuration Guide*, you configured the cron facility with information to start database and cache storage maintenance programs, the database index maintenance program, and programs that backup the OnDemand and ADSM databases automatically on a regular schedule.

Starting the database

Under normal operating conditions, you never have to start the database on your own. However, should you need to, use the arsdb command to start the database. For example:

AIX `/usr/lpp/ars/bin/arsdb -gkv`

HP-UX, Solaris
`/opt/ondemand/bin/arsdb -gkv`

See "Chapter 25. arsdb command reference" on page 173 for more information about the arsdb command, parameters, and options.

Starting ADSM

Under normal operating conditions, you never have to start ADSM on your own. However, should you need to, use the ars_adsm command to start ADSM. For example:

AIX `/usr/lpp/ars/bin/ars_adsm -sv`

HP-UX, Solaris
`/opt/ondemand/bin/ars_adsm -sv`

See "Chapter 23. ars_adsm command reference" on page 165 for more information about the ars_adsm command, parameters, and options.

Starting Download

Under normal operating conditions, you never have to start Download on your own. However, should you need to, use the arsjesd command to start Download. For example:

AIX
```
/usr/lpp/ars/bin/arsjesd -p 6001 -d /arsacif/acif1 -d
/arsacif/acif2 2>&1
```

HP-UX, Solaris
```
/opt/ondemand/bin/arsjesd -p 6001 -d /arsacif/acif1 -d
/arsacif/acif2 2>&1
```

The strings /arsacif/acif1 and /arsacif/acif2 identify filesystems on the server used by Download. The string 6001 identifies the TCP/IP port Download monitors for data transmitted by the MVS or OS/390 system. See "Chapter 27. arsjesd command reference" on page 201 for more information about using the arsjesd command with Download.

Starting related MVS and OS/390 programs

After you power on the workstation or reinitialize the operating system, complete the following steps to enable communications between any MVS or OS/390 systems connected to the OnDemand system:

1. Vary online communications controllers between MVS and OS/390 systems and the server.
2. Recycle TCP/IP.
3. If you use the OnDemand CICS/ESA client program, enable the MVS or OS/390 TCP/IP port for the client. Invoke the **CSKE** transaction to enable the port. You must specify the TCP/IP port number for the client, for example, 3000.
4. If you use Download, start a Download writer. You can use the **$sprtnnnn** command to start a writer. Replace the string **nnnn** with the name of a Download writer. See *PSF for MVS: MVS Download Guide* and *PSF for OS/390: Download for OS/390* for information about operating and using Download.

Starting the server program

Under normal operating conditions, you never have to start the server program on your own. However, should you need to, use the arssockd (or arsobjd) program to start the server program. For example:

AIX
```
/usr/lpp/ars/bin/arssockd
```

HP-UX, Solaris
```
/opt/ondemand/bin/arssockd
```

To start the server program on an object server, replace the string `arssockd` with `arsobjd`.

Starting arsload

If you configured the system to automatically start the arsload command, under normal operating conditions, you never have to start the arsload command on your own. However, should you need to, use the *at now* command to start the arsload command, without associating the process with the current logged in user. For example, on an AIX server, enter:

```
at now
/usr/lpp/ars/bin/ars_load -v
     -d /arsacif/acif1 -d /arsacif/acif2
     -c /arsacif/acif3
^D
```

In the example, `^D` means to press the Ctrl and D keys at the same time. The strings `/arsacif/acif1`, `/arsacif/acif2`, and `/arsacif/acif3` identify file systems used by the arsload command to index and load data. See "Chapter 28. arsload command reference" on page 205 for more information about the arsload command, parameters, and options.

Stopping OnDemand

If you need to power off or reboot the server, reinitialize the operating system, or stop OnDemand operation for any reason, complete the following steps to bring the OnDemand operating environment to an orderly conclusion. Stopping OnDemand in this way allows the key processes, such as DB2 UDB and ADSM, to terminate normally, clearing buffers, closing log files, and dismounting storage volumes.

Note: The following procedure can require several minutes to complete. If you need to stop the system immediately, halt the operating system with the *shutdown* command. Then power off the server.

1. Login as the *root* user.
2. If you use Download, make sure that no data download operations are in process. Then, list the PID number of the arsjesd command and cancel the process with the kill command.
3. Make sure that no data loading operations are in progress. Then, list the PID number of the arsload command and cancel the process with the kill command.

4. Make sure that no database or cache maintenance operations are in progress. Then, list the PID numbers of the arsdb command that maintains the database indexes and the arsmaint command and cancel them with the kill command.
5. On all object servers, list the PID number of the arsobjd command and cancel the process with the kill command.
6. On the library server, list the PID number of the arssockd command and cancel the process with the kill command. Cancel the process that has a PPID of 1 (one).
7. On the library server, cancel any child processes of the arssockd command with the kill command.
8. Stop DB2 UDB.

   Complete the following steps.

   - Load the DB2 UDB environment, if it is not already loaded. For example:

     ```
     . /home/archive/sqllib/db2profile
     ```

   - Start the DB2 UDB command line processor. Enter the following command at the prompt:

     ```
     db2
     ```

   - Connect to the OnDemand database. Enter the following command at the DB2 prompt:

     ```
     connect to archive
     ```

   - Close the OnDemand database connections. Enter the following command at the DB2 prompt:

     ```
     force application all
     ```

   - Stop the DB2 command line processor. Enter the following command at the DB2 prompt:

     ```
     terminate
     ```

   - Stop DB2 UDB. Enter the following command at the prompt:

     AIX
     ```
     /usr/lpp/ars/bin/arsdb -hv
     ```

     HP-UX, Solaris
     ```
     /opt/ondemand/bin/arsdb -hv
     ```

9. Start an ADSM administrative session.
   - Issue the QUERY SESSION command to list any open ADSM sessions.
   - Use the CANCEL SESSION command to stop any open ADSM sessions.
   - Issue the QUERY MOUNT command to list any mounted storage volumes.
   - Use the DISMOUNT VOL command to cause ADSM to remove any mounted storage volumes.

10. Stop the ADSM administrative session.
11. Stop ADSM. Type the following command at the prompt:

    AIX `/usr/lpp/ars/bin/ars_adsm -tv`

    HP-UX, Solaris
    `/opt/ondemand/bin/ars_adsm -tv`

    The example uses the ADSM administrative userid and password set in the ars_adsm file.
12. If you use Download, stop any active Download writers on MVS and OS/390 systems. You can use the **$pprtnnnn** command to stop a writer. Replace the string **nnnn** with the name of an active Download writer. See *PSF for MVS: MVS Download Guide* and *PSF for OS/390: Download for OS/390* for information about stopping Download.
13. If you use Download, stop the Download functional subsystem under JES. You can use the **cancel fssname** command to stop the functional subsystem. Replace the string **fssname** with the name of the functional subsystem for Download. See *PSF for MVS: MVS Download Guide* and *PSF for OS/390: Download for OS/390* for information about stopping Download.
14. If you use the OnDemand CICS/ESA client, disable the MVS or OS/390 TCP/IP port for the client. Invoke the **CSKD** transaction to disable the port.
15. If you use either Download or the CICS/ESA client, vary offline communications controllers between MVS and OS/390 systems and the server.
16. Stop the operating system. Issue the **shutdown** command.

You can now power off or reboot the server or reinitialize the operating system.

Stopping the OnDemand server program

Complete steps 5 through 7 on page 92 if you need to stop the OnDemand server program without stopping other related processes, such as DB2 UDB and ADSM.

Stopping ADSM

Note: Stopping OnDemand in a distributed system, client/server environment should be done in an orderly, structured manner. We recommend you follow the steps outlined in “Stopping OnDemand” on page 91 to shutdown the system so that the various components of OnDemand remain synchronized.

Complete steps 9 through 11 on page 93 if you need to stop ADSM without stopping other related processes, such as DB2 UDB and the OnDemand server program.

Stopping DB2 UDB

Note: Stopping OnDemand in a distributed system, client/server environment should be done in an orderly, structured manner. We recommend you follow the steps outlined in "Stopping OnDemand" on page 91 to shutdown the system so that the various components of OnDemand remain synchronized.

Complete step 8 on page 92 if you need to stop DB2 UDB without stopping other related processes, such as ADSM and the OnDemand server program.

Restarting OnDemand

If you followed the instructions in the *Installation and Configuration Guide*, the operating system automatically starts OnDemand processes when you power on the system. The init facility starts the OnDemand server program and other required processes, such as DB2 UDB and ADSM. The cron facility starts OnDemand maintenance programs. Refer to "Starting OnDemand" on page 89 for information about initialization processing, starting OnDemand processes, and enabling related MVS programs.

Whenever you restart the server after a hardware or software problem and the operating system did not complete a normal system shutdown, verify that DB2 UDB and ADSM are operating without corruption to the OnDemand database and the ADSM database. This is especially important if the arsload command was running when the system problem occurred. If OnDemand was loading data and the system failed, you may need to perform one or more of the following tasks before allowing users to access the system:

- Check the configuration of the OnDemand database and DB2 UDB. Verify the database is in a consistent state with no backup or rollforward pending.
- Recover the OnDemand database if required. Refer to "Database recovery" on page 56 for details.
- Audit the ADSM database. Check the database for consistency and correct any problems that are reported. Refer to "ADSM database recovery" on page 61 for details.
- Recover any load files that were being processed when the system failed. Refer to "Chapter 4. Restarting a load process" on page 39 for details.

Chapter 16. Monitoring the server

You probably want to monitor OnDemand resources, such as processes, memory, disk space, and storage volumes, on a regular basis. You can run the commands described in this chapter and save the information reported to track resource utilization and to establish a performance baseline.

We recommend that you run these commands during key, peak processing periods and periods of little or no load. Armed with these statistics, you can quickly determine how the system is performing compared to the baseline. The performance baseline along with, for example, monthly *snapshots*, can help you measure transaction growth, predict the impact of adding applications and users, and determine when you need to add resources, such as storage volumes.

Even if you don't track these items periodically, OnDemand may issue messages that suggest potential problems with the system or users may complain of sluggish response to search and retrieval requests.

Regardless, the commands described in this chapter can help you spot potential problems and bottlenecks in the system and give you information to begin corrective action.

Errors and alerts

OnDemand assigns a severity to each message generated by server programs. OnDemand sends messages assigned a severity of *alert* or *error* to the system console device */dev/console*.

OnDemand provides a system logging facility and a message catalog to help you identify and resolve any alerts and errors that you receive. You can open the System Log folder to display messages recorded in the system logging facility. Refer to "OnDemand system logging facility" on page 96 for more information. You can search the message catalog to obtain details about messages.

For help with specific error messages, refer to the following publications for a listing of error messages and associated articles:

- *Messages Guide and Reference* for your operating system
- *DB2 UDB Messages Reference*
- *ADSM Messages*

OnDemand system logging facility

OnDemand provides a logging facility to help the system administrator track OnDemand activity and monitor the system. OnDemand records messages generated by the various OnDemand programs, when you permit logging for system events, user events, and application group events. You can use one of the OnDemand client programs to search for and filter messages by time stamp, severity, message number, and user name.

Note: Before starting OnDemand for the first time, you must initialize the system logging facility. See the *Installation and Customization Guide* for your server for information about initializing the system logging facility.

Searching for and viewing messages

Log on to OnDemand with an OnDemand client program and open the System Log folder.

You can specify a time stamp to search for and retrieve messages. You can specify other search criteria, for example:

- Log Id. Each time an end-user logs on the server, OnDemand assigns a transaction number to that instance of the client program. All requests issued by that instance of the client program includes the same Log Id.
- Userid. The OnDemand userid.
- Severity. OnDemand assigns a severity to each message: Alert, Error, Warning, Info, and Debug.
- View. Depending on the type of message in the log, you may be able to view related records or other data associated with the message from the System Log folder:
  - OnDemand records messages in the log that were issued by programs, such as the arsload command. You can display the messages created during a load by selecting an item that represents the arsload command and choosing the **View all Selected** command.
  - While most messages do not support related records or other data that can be displayed from the System Log folder, you can write a user exit program to process the messages and report information about events. For example, you could write a user exit program that reports the amount of time required to retrieve large objects from an application group. OnDemand provides a system log user exit so that you can process messages and take the appropriate action. See the *Installation and Configuration Guide* for your server for more information about the system log and the system log user exit.

  - Other messages in the system log do not provide additional records or other data associated with the event, for example, the Logoff event.
- Msg Num. The message number issued by OnDemand.
- Message. Enter the text of the message that OnDemand uses to restrict a search. For example, if you type **Login**, OnDemand searches for and displays the messages issued by the Logon to a Server command.

When you choose the **Search** command, OnDemand retrieves the messages from the database that match the search criteria you specified.

Monitoring users

OnDemand servers generate system log messages to help you track the number of users logged on to the server. OnDemand writes message number 201 to the system log every 30 minutes. This message contains the current number of users logged on to the server. OnDemand writes message number 202 to the system log every time the number of concurrent users exceeds the previous maximum number of concurrent users.

Reviewing a CD-ROM creation manifest

You can use the system logging facility to review a manifest created by the OnDemand ad-hoc CD-ROM mastering software. A CD-ROM creation manifest contains a cover page with control information about the CD-ROM and one or more pages that list the items retrieved from the database and written on the CD-ROM.

To review a CD-ROM creation manifest:

1. Log on to OnDemand with an OnDemand client program and open the System Log folder.
2. Type **CD** in the Message entry field.

   Note: If you know the CD-ROM volume number of the manifest that you want to review, enter the number in the Message entry field. For example, you can enter **%171** in the Message entry field to locate the manifest for volume number AOD00000171.
3. Specify a date. For example:
   - If you want to display a manifest for a particular date, select the **Equal** operator for the Time Stamp search field and enter the date in the Time Stamp entry field.
   - If you want to display the manifests created during a range of dates or you do not know the exact creation date, select the **Between** operator for the Time Stamp search field and enter a date range in the Time Stamp entry fields.

4. Choose **Search**. OnDemand displays the manifests that match the search criteria that you specified.
5. Review a manifest by selecting an item from the list and choosing the **View All Selected** command.
6. Use the Next Page and Previous Page commands to scroll through the manifest.

System log user exit

When you enable logging for system, user, or application group events, OnDemand sends messages to the arslog program. The arslog program provided by IBM does not perform any functions. However, you can replace the arslog program with one that does specific functions, such as check the severity of messages and issues alerts when appropriate.

You can configure OnDemand to provide information to the arslog program by setting the appropriate system, user, and application group logging parameters with the administrator interface.

See the *Installation and Configuration Guide* for your server for more information about the system log user exit and the arslog program.

Monitoring processes

You can monitor and display status information about OnDemand, DB2 UDB, and ADSM processes. To monitor processes, use the *ps* command. The *ps* command is an operating system command that returns process status information about active processes to standard output. Key fields in the *ps* command output include UID (user), PID (process), C and PRI (CPU use and execution priority), STIME (the time the process started), TIME (accumulated CPU time), and CMD (the command the process is running).

The following examples show how to use the *ps* and *grep* commands to monitor key processes on the OnDemand server. For more information about the *ps* and *grep* commands, type **man ps** or **man grep** at the prompt or refer to your operating system documentation.

Displaying OnDemand process information

OnDemand programs and commands that may be active on the server include the following:

arsdb The OnDemand command used to maintain the database. Refer to the “Chapter 25. arsdb command reference” on page 173 for information about starting the arsdb command to perform database maintenance.

arssockd
The main OnDemand program that runs on the library server. If OnDemand client programs cannot connect to the server, make sure that this process is active. Refer to “Starting the server program” on page 90 for instructions about how to restart the arssockd command.

There may be more than one arssockd process active, depending on the value of the ARS_NUM_DBSRVR parameter in the ars.cfg file and the number of instances running on the workstation or node. See the *Installation and Configuration Guide* for information about the ARS_NUM_DBSRVR parameter and running instances.

arsobjd
The main OnDemand program that runs on an object server. If OnDemand client programs experience problems retrieving reports from the server, make sure that this process is active. Refer to “Starting the server program” on page 90 for instructions about how to restart the arsobjd command.

There may be more than one arsobjd process active, if you are running more than one instance on the workstation or node. See the *Installation and Configuration Guide* for information about running instances.

arsload
The data loading program. To automatically store data in application groups, make sure that this process is started. Refer to “Starting arsload” on page 91 for instructions about how to restart the arsload command. You may see two other commands associated with this process. The arsacif and arsadmin commands run with the parent process ID of the arsload process.

arsacif The data indexing program invoked by the arsload command.

arsadmin
The database and storage manager program invoked by the arsload command.

arsjesd
The server component of Download. To automatically receive datasets from Download running on MVS and OS/390 systems, make sure that one of these processes is started for each system or port on a system from which you plan to receive data. Refer to “Starting Download” on page 90 for instructions about how to restart the arsjesd command.

arsmaint
The process that maintains cache storage and the OnDemand database.

Use the following command to display process status:

```
ps -ef ¦ grep ars
```

The output of the command issued on the library server should be similar to the following example:

```
UID  PID    PPID   C PRI  STIME     TTY  TIME CMD
root  6022 28991   0  60  13:30:13    -  0:00 arssockd: db (USERID)
root  6022 28991   0  60  13:30:13    -  0:00 arssockd: db (USERID - idle)
root  6018 28991   0  60  13:30:13    -  0:00 arssockd: db (idle)
root  6014 28991   0  60  13:30:13    -  0:00 arssockd: (accepting)
```

Displaying DB2 UDB processes

DB2 programs that may be active on the library server include the following:

db2agent
Handles all SQL processing for application requests. Several occurrences of this process may be active.

db2dlock
The data lock manager looks for and resolves deadlocks on the database.

db2gds
Responsible for spawning most DB2 daemons.

db2ipccm
Handles connection requests from local clients.

db2licd
The license daemon.

db2loggr
This process manages the database log files.

db2pclnr
Asynchronously writes dirty pages when the CPU would otherwise be idle. Several occurrences of this process may be active.

db2pfchr
Allows for more efficient processing with read-ahead, big-block and parallel I/O. Several occurrences of this process may be active.

db2resyn
Handles recovery of indoubt transactions.

db2sysc
This process handles DB2 startup and shutdown.

db2wdog
The “watch dog” monitors the other processes and cleans up resources after abnormal terminations.

Use the following command to display process status:

```
ps -ef ¦ grep db2
```

The output of the command should be similar to the following example:

```
    UID  PID    PPID   C STIME     TTY  TIME CMD
    root 21557     1   0 Apr 12      -  0:00 db2wdog
 archive 21814 21557   0 Apr 12      -  0:30 db2sysc
 archive 22071 21814   0 Apr 12      -  0:00 db2dlock
   db2as 15596 15596   0 Apr 12      -  0:00 db2sync
   db2as 15854 15596   0 Apr 12      -  0:00 db2gds
   db2as 16112 15596   0 Apr 12      -  0:00 db2ipccm
   db2as 16370 15596   0 Apr 12      -  0:00 Scheduler
   db2as 16628 15596   0 Apr 12      -  0:00 db2tcpcm
   db2as 16886 15596   0 Apr 12      -  0:00 db2tcpdm
 archive 24642 22328   0 Apr 12      -  0:04 db2loggr
 archive 31611 22071   0 Apr 19      -  0:05 db2gds
    root 22343     1   0  08:09      -  0:00 db2licd
 archive 34694 21601   0  13:13      -  1:09 db2agent (ARCHIVE)
 archive 34695 21601   0  13:13      -  1:09 db2agent (idle)
 archive 34949     1   0  13:13      -  1:20 db2agent (ARCHIVE)
 archive 22071 42414   0 Apr 12      -  0:00 db2ipccm
 archive 24642 22328   0 Apr 12      -  0:04 db2resyn
 archive 31665 42414   0 Apr 19      -  0:05 db2pfchr
 archive 31666 41414   0 Apr 19      -  0:05 db2agent (ARCHIVE)
```

Displaying ADSM processes

The following processes are associated with ADSM:

dsmserv
The ADSM server. Several occurrences of this process may be active.

dsmlabel
The ADSM command used to label storage volumes. This process must not be active at the same time that the dsmserv process is active.

Use the following command to display process status:

```
ps -ef ¦ grep dsm
```

The output of the command should be similar to the following example:

```
UID  PID    PPID   C STIME     TTY   TIME CMD
root 13626 24106   0 Apr 15      -   0:00 -dsmadmc=archive
root 14126 24106   0 Apr 15      -   0:00 dsmserv quiet
root 26413     1   0 Apr 15      - 115:23 dsmserv quiet
```

Monitoring system load

The uptime command displays an estimate of the system load. Type the following command at the prompt:

```
uptime
```

Information reported by the uptime command includes the number of users on the system and load averages for the latest minute, 5 minutes, and 15 minutes. For example:

```
11:40AM  up   4:39,  2 users,  load average: 0.17, 0.10, 0.09
```

In the example, the system load in the last minute is 17 percent, over the last 5 minutes is 10 percent, and over the last 15 minutes is 9 percent.

High load averages usually mean that the system is being used heavily and probably means users are experiencing slow response time. For more information about the uptime command, type **man uptime** at the prompt or refer to your operating system documentation.

Monitoring memory usage

To monitor memory use and paging activity, use the vmstat command along with the ps command. Type the following command at the prompt:

```
vmstat
```

The most important pieces of information reported by the vmstat command are the number of processes running and the number of page-outs and swaps. For example:

```
procs     memory             page              faults        cpu
----- ----------- ------------------------ ------------ -----------
 r  b   avm   fre  re  pi  po  fr   sr  cy  in   sy  cs us sy id wa
 0  0  5143 26542   0   0   0   0    0   0 135  179  78  1  5 90  5
```

After running the vmstat command, use the ps command to display the processes that are generating the memory usage displayed by the vmstat command.

For more information about the vmstat command, type **man vmstat** at the prompt or refer to your operating system documentation.

Monitoring system paging space

To monitor system paging space, use the *lsps* command. Type the following command at the prompt:

```
lsps -a
```

The *lsps* command reports the size of the page space and the percent used. For example:

```
Page Space  Physical Volume   Volume Group    Size   %Used  Active  Auto  Type
hd6         hdisk0           rootvg          64MB      17     yes   yes    lv
```

For more information about the lsps command, type **man lsps** at the prompt or refer to your operating system documentation.

Monitoring disks and file systems

The *df* command reports capacity numbers for disks and file systems, including cache file systems. Type the following command at the prompt:

```
df
```

The command shows the percentage each file system is used and how much space remains in each file system. For example:

```
Filesystem     Total KB      free %used   iused %iused Mounted on
/dev/hd4          24576     11112   54%     939    15% /
/dev/hd9var       53248     49964    6%     153     1% /var
/dev/hd2         548864     10000   98%   29610    21% /usr
/dev/hd3          12288      9876   19%      46     1% /tmp
/dev/lv07          4096      3004   26%      66     6% /home
/dev/lv00       1880064    423532   77%      42     0% /arsacif/acif1
/dev/lv01       1875968    484860   74%     111     0% /arsacif/acif2
/dev/lv02       1019904    551420   45%     106     0% /arsacif/acif3
/dev/lv03       1019904    960896    5%      20     0% /arsdb
/dev/lv04       1019904    960896    5%      20     0% /arsdb_primarylog
/dev/lv05       1019904    960896    5%      20     0% /arsdb_archivelog
/dev/cache1      323584    313272    3%      16     0% /arscache/cache1
/dev/cache2      323584    313272    3%      16     0% /arscache/cache2
/dev/cache3      323584    313272    3%      16     0% /arscache/cache3
/dev/cache4      323584    313272    3%      16     0% /arscache/cache4
/dev/cache5      323584    313272    3%      16     0% /arscache/cache5
/cdrom/aix       380928     49220   87%       -      -  /usr/lpp/info
```

Monitor the file systems that support the OnDemand database on a regular basis. Before the file systems become fully utilized, you must increase the size of the file systems or define additional table space file systems. See the *Installation and Configuration Guide* for information about how to define additional table space file systems.

Monitor the file systems that support the document cache on a regular basis. Before the cache file systems become fully utilized, you must increase the size of the file systems or define additional cache storage file systems. See the *Installation and Configuration Guide* for information about how to define additional cache storage file systems.

For more information about the df command, type **man df** at the prompt or refer to your operating system documentation.

Monitoring I/O activity

The *iostat* command reports input/output statistics for disks and shows you the time physical disks are active in relation to their average transfer rate. You can use the information reported by the iostat command to better balance input/output load between physical disks. Type the following command at the prompt:

```
iostat
```

For a report interval and each physical disk on the system, the iostat command reports the percentage of time the disk was active (tm_act), the amount of data read or written to the drive (Kbps), transfers per second to the physical disk (tps), and the total amount of data read and written (Kb_read and Kb_wrtn). For example:

```
Disks:        % tm_act     Kbps      tps    msps   Kb_read   Kb_wrtn
hdisk0           0.0       0.1       0.0             5993      9452
hdisk1           0.0       0.3       0.0            31193      6812
hdisk2           1.9       2.1       0.8            13153    232508
hdisk3           0.0       0.0       0.0                0         0
hdisk4           0.0       0.0       0.0             4293      4936
```

You can display continuous reports using the following format. In the example, the iostat command displays reports every 30 seconds for the logical disk `hdisk2`:

```
iostat -d hdisk2 30
```

For more information about the iostat command, type **man iostat** at the prompt or refer to your operating system documentation.

Monitoring storage volumes

You can use the following commands to display reports about ADSM storage volume utilization.

- Space report.

  The space report provides general information about the storage volumes you defined, including the status, capacity, and utilization of each storage volume. For example:

  ```
  query volume format=detailed
  ```

- Database report.

  The database report displays the volumes assigned to the database. For example:

  ```
  query db format=detailed
  ```

When these reports show the ADSM server is approaching the capacity of the storage volumes, you must label and check new storage volumes into the library.

For more information about monitoring storage volumes, refer to the *ADSM Administrator's Guide*

Monitoring the network

The netstat command monitors TCP/IP network activity on a server. This command is useful for monitoring and gathering information, such as the number of clients that are connected to the server, whether the network connection is active or waiting, and how much data is currently being transferred between the server and clients. For example:

```
netstat -f inet
```

The netstat command displays a report similar to the following example:

```
Active Internet connections
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp        0  15972  pikes.bpc.boulde.1445  csr0018.bpc.boulde.1039  ESTABLISHED
tcp        0   4039  pikes.bpc.boulde.1445  csr0013.bpc.boulde.1049  ESTABLISHED
tcp        0      0  pikes.bpc.boulde.1501  elric.bpc.boulde.1635  ESTABLISHED
tcp        0      0  pikes.bpc.boulde.1445  csr0001.bpc.boulde.1026  ESTABLISHED
tcp        0      0  pikes.bpc.boulde.1026  pikes.bpc.boulde.4242  ESTABLISHED
tcp        0      0  pikes.bpc.boulde.4242  *.*                    LISTEN
```

The Local Address field contains the TCP/IP port address for the server. Clients connected to the server can be identified using the port address. In this example, the TCP/IP port address is 1445.

For more information about the netstat command, type **man netstat** at the prompt or refer to your operating system documentation.

Part 4. Operating Windows NT Servers

This section of the book describes day-to-day, operational tasks that you may need to perform when supporting an OnDemand Windows NT system.

Topic	Page
Administrative tools and procedures	109
Readying the system for use	113
Managing servers	117

Chapter 17. Administrative tools and procedures

There are many tools to help you administer an OnDemand system. For example:

- OnDemand provides two administrative programs and several utilities to help you administer your system.
- The relational database management product you installed on the library server comes with administrative programs and utilities.
- If you installed ADSM to maintain data on archive media, ADSM provides a set of tools to help you define and maintain the storage it manages.

Other tools are provided with the Windows NT operating system.

Administering servers with OnDemand tools

OnDemand configurator

The OnDemand configurator is installed on an Windows NT server when you install the OnDemand software. Use the configurator to configure the database manager and storage manager, maintain OnDemand services, and schedule maintenance programs. You can manage servers locally or remotely with the configurator. To use the configurator, start OnDemand Configurator from the OnDemand for WinNT program group. The *ADSM for Windows NT Version 3 Administrator's Guide* describes how to use the configurator to initially set up your OnDemand system. Refer to the online help for information about commands and options in the configurator.

Administrative client

The administrative client can be used to maintain users, groups, application groups, applications, folders, storage sets, and printers. The administrative client runs under the Windows NT, Windows 95, and Windows 98 operating systems. The *Installation Guide for OnDemand Clients* describes how to install the administrative client. To use the administrative client, start OnDemand Administrator from the OnDemand program group. See *Getting Started with the Administrative Client* and the online help for information about using the administrative client.

OnDemand also provides several administrative utilities or server commands. You can use server commands to load and maintain data, add users to the system, and perform other administrative functions. You can run server commands from the OnDemand command line interface. You can also use the

OnDemand configurator to setup and run server commands. The server commands are documented in “Part 5. Command Reference” on page 133.

Administering databases with DB2 tools

You can perform database administration tasks locally or remotely using the DB2 administration tools. The DB2 Control Center allows you to graphically perform administrative tasks such as configuring, backing up and recovering data, managing file systems, scheduling jobs, and managing media. You can also manage database manager environments (known as DB2 instances), for example, you can set database manager configuration values that affect performance. You can choose to have the Control Center installed on OS/2, Windows NT, or Windows 95 workstations. Refer to the DB2 UDB Verson 5 Quick Beginnings publication for information about getting started with the Control Center. Refer to the *DB2 Universal Database Version 5 Administration Guide* for details about maintaining DB2 databases and instances.

Administering SQL Server

Microsoft SQL Server Version 6.5 comes with administrative tools and utilities to help you manage the system. The SQL Enterprise Manager is a graphical user interface that allows you to perform system administration tasks such as configuring the server, managing databases and database objects, scheduling events, and setting up alerts. You can choose to have SQL Enterprise Manager installed on Windows NT or Windows 95 workstations. Refer to the documentation provided with SQL Server for more information about SQL Enterprise Manager and the other administrative tools and utilities provided with SQL Server.

The Database Maintenance Plan Wizard allows you to configure and schedule database maintenance. The maintenance actions can include optimizing the database, reorganizing data and index pages, and performing database and transaction log backups. Maintenance actions can be scheduled to occur on daily or weekly intervals.

Note: When you use SQL Server as your database manager, we recommend you use the Database Maintenance Plan Wizard to configure and schedule database maintenance rather than the tools and utilities provided with OnDemand.

Administering ADSM

ADSM provides tools to help you administer the storage it manages. Using the ADSM administrative client graphical user interface, you can perform the following tasks:

- Register additional administrators
- Register client nodes
- Create new policies
- Automate storage operations and administrative commands

You can define automated devices to ADSM using the ADSM administrative command line interface.

The *ADSM for Windows NT Version 3 Quick Start* publication can help you learn about these tools. The *ADSM for Windows NT Version 3 Administrator's Guide* and the *ADSM for Windows NT Version 3 Administrator's Reference* provide details about ADSM concepts, operations, and commands that you'll use to manage ADSM storage.

Administering Windows NT

Windows NT provides several administrative programs and utilities that can help you control access to the system, list information about the system and system resources, manage server storage, track the performance of the system, and debug system problems. The primary tools you'll use are:

- Disk Administrator. The Windows NT tool used to manage hard disks. Use Disk Administrator to create, format, and delete partitions, volume sets, and stripe sets. You can also use Disk Administrator to assign drive letters, labels, and CD-ROM drive letters. Start Disk Administrator from the Administrative Tools program group.
- User Manager. The Windows NT tool used to manage accounts on the local domain or on any workstation, member server, or other domain to which you have access. A user account contains information that defines a user to Windows NT. You can use User Manager to add, delete, or disable accounts, set security policies, and add accounts to groups. Windows NT accounts are different than and administered separately from OnDemand users. Start User Manager from the Administrative Tools program group.
- Server Manager. Monitors session activity and keeps track of all resources and which users on the network are accessing those resources. Server Manager displays statistics showing current usage levels on both servers and NT workstations. You can also use Server Manager to send messages to users and designate computers that will receive alerts. Start User Manager from the Administrative Tools program group.

- Event Viewer. Provides information about such events as errors, warnings, and the success or failure of tasks. An event is any potentially significant occurrence in the system or in an application. Some critical events, such as a full disk drive or an interrupted power supply, are noted in on-screen messages, as well as in the Event Log. Non-critical events are merely logged. Event logging starts automatically each time Windows NT is started. With an Event Log and Event Viewer, it is possible to troubleshoot various hardware and software problems and to monitor Windows NT security events. Start Event Viewer from the Administrative Tools program group.
- Task Manager. Shows a quick view of how each application, application component, or system process is using CPU and memory resources, as well as a summary of overall CPU and memory usage. To run Task Manager, right-click the taskbar, and then click Task Manager. For more information on using Task Manager, see Task Manager Help.
- Windows NT Diagnostics. Shows computer hardware and operating system data stored in the Registry. Windows NT Diagnostics displays the status of services and devices, displays the system resources in use, and lists environment variables and network-related, operating system (such as the version and build, BIOS, HAL, and CPU), display adapter and driver, and physical and virtual memory information. Start Windows NT Diagnostics from the Administrative Tools program group.
- Performance Monitor. Provides information to help you identify and troubleshoot possible problems and plan for additional hardware needs. Use Performance Monitor to monitor real-time and historical system performance, identify trends over time, identify bottlenecks, monitor the effects of system configuration changes, and determine system capacity. Start Performance Monitor from the Administrative Tools program group.

  Note: For comprehensive documentation on monitoring Windows NT performance, see the *Windows NT Server Resource Kit Version 4.0*.
- System Recovery. If there is a severe error (a Stop error), it is possible to configure the system response using the Recovery options on the Startup/Shutdown page of the System program in Control Panel. When a Stop error occurs, you can write an event to the system Event Log, send an administrative alert, write debugging information to a file, and automatically reboot the system. The Write Debugging Information option is important for troubleshooting and should always be enabled.

Chapter 18. Readying the system for use

This chapter identifies the tasks that you need to complete to set up your OnDemand for Windows NT system for initial and ongoing use:

- Install and configure the system
- Backup databases and other critical files and information
- Control access to the system
- Software licensing

Install and configure the system

An OnDemand system contains one or more object servers communicating with one library server. OnDemand clients connect to the system by logging on and accessing the library server. The *Installation and Configuration Guide for Windows NT Servers* contains detailed instructions on installing and configuring servers. The *Installation Guide for OnDemand Clients* contains instructions on installing OnDemand client software on a PC.

After you've set up your system for initial use, you can use the OnDemand configurator program and other tools and utilities to make changes to the server operating environment. For example, you can define new cache or database file systems to OnDemand and change the number of concurrent database connections allowed. You can also use the configurator program to schedule maintenance tasks, such as migration and expiration processing for the database and cache storage. After you set up maintenance tasks for initial use, you can use the OnDemand configurator program to make changes to the schedule and runtime options or configure additional occurrences of the tasks with different options and schedules.

Note: After you make changes to the system with the configurator, you must stop and restart the OnDemand services. See "Starting and stopping services" on page 117 for more information.

Backup databases and other information

After you set up your system for initial use, we recommend that you create a backup copy of the OnDemand database, ADSM database, and other files and information critical to the operation of the system. You should backup this data to removable media.

We also recommend that you schedule regular backups during ongoing use of the system. For example, after you load reports into the system, we recommend that you create a backup copy of the OnDemand and ADSM databases. When you define new devices and policies to ADSM, you should backup the ADSM database and save the ADSM configuration files. When you make changes to the hardware and software configuration, you should backup the Registry.

"Chapter 8. Backup and recovery" on page 51 contains more information about the server components that should be protected and methods and procedures that you can use to ensure that the system can be recovered if needed.

For DB2, you can configure and schedule the Database Backup task with the OnDemand configurator program. For SQL Server, we recommend you use the Database Maintenance Plan Wizard to configure and schedule database backups.

Control access to the system

Controlling access to OnDemand consists of creating users and groups, assigning passwords, and assigning folder and application group permissions. You create users and groups and assign folder and application group permissions with the administrative client. Start the OnDemand Administrator program from the OnDemand program group. Refer to the online help and *Getting Started with the Administrative Client* for information about using the administrative client.

Creating users and groups

Only users who have been identified to OnDemand can log on to the library server. Users with common access needs can be grouped to make it easier for you to maintain permissions. Use the administrative client to create users who need to access OnDemand and create groups for users with common needs. You can assign a user to a group when you create the user or when you create the group.

Assigning permissions

Access to information stored in OnDemand is protected by permissions. For example, when a user logs on the system, OnDemand verifies that the userid is valid, verifies the password entered by the user, and creates a list of the folders that the user is permitted to access. Additional permissions can be specified for an application group, so that you can control whether a user can

view, print, FAX, or copy documents stored in the application group. Refer to *Getting Started with the Administrative Client* for details about permissions.

Software licensing

This section contains information about client access to the server, counting concurrent users, and other software licensing issues.

OnDemand

OnDemand is a client/server product which has three license units: base, object server, and concurrent user. The base includes one library server, one object server, and one concurrent user. A user license is required for each additional concurrent user beyond the first user. An object server license is required for each machine beyond the first machine on which an object server will be run.

ADSM

When you install ADSM for Windows NT, your system is licensed for the base ADSM support, which provides the following:

- An unlimited number of administrative clients
- One backup-archive client
- Certain types of removable media devices

Depending on your system configuration, you may need to register ADSM licenses beyond the base support. For example, if you need to support many storage pools or client nodes, you probably need to license additional clients. If you need to define devices to ADSM that are not covered in the base device license support, you probably need to register a license for advanced device support. The *Installation and Configuration Guide for Windows NT Servers* provides information about registering ADSM licenses. Contact the IBM support center for current information about devices supported by ADSM.

DB2 UDB

DB2 UDB Enterprise Edition allows an unlimited number of connections to the database. Under normal OnDemand operations, there is very little you need to do to maintain DB2 software licenses.

Important: If you set the Number of Database Servers to a value other than 0 (zero) or 1 (one), you must update the DB2 license information. Use the Nodelock Administration tool to update the number of concurrent licenses to a value equal to or greater than the Number of Database Servers. The Number of Database Servers is a

property of the OnDemand instance. Use the OnDemand configurator program to maintain OnDemand instances.

The Registration program, located in the DB2 program group, can be used to access and update the IBM Software Customer Profile and register DB2 software with IBM. The customer profile contains identification and demographic information about the user of each IBM software product. The IBM software registration contains the information and the instructions on how to register DB2 software by modem, telephone, Internet, FAX, or postal service.

Windows NT Server

In order to connect to a computer running Windows NT Server, the appropriate client software must be installed and configured on the client. In addition, appropriate licensing for the server and the clients must be obtained to ensure that network access complies with the Microsoft licensing legal requirements. Certain applications, such as SQL Server, also require appropriate licensing for client access.

Each client that accesses Windows NT Server to use file or print services, Remote Access Services, or share applications such as Microsoft Office products requires a Client Access License. Each client that accesses Backoffice applications, such as SQL Server, requires a separate Client Access License. If you plan to install OnDemand on a workstation that provides no other operating system or application services to clients, you only need to maintain Client Access Licenses for SQL Server.

During installation, the setup program requires you to select a licensing mode and then accept the terms of the license agreement. There are two licensing modes, Per Server and Per Seat:

- With Per Server licensing, the license is assigned to a particular server. You specify a maximum number of simultaneous user connections for the server. To change Per Server licensing information after installation, use the Licensing program in the Control Panel.
- With Per Seat licensing, the license is assigned to a particular workstation. Each workstation that will be accessing the server requires a Client Access License. This method allows an unlimited number of clients to access the server. To change Per Seat licensing information after installation, use the License Manager program in the Administrative Tools program group.

For more information about licensing, see the *Microsoft Windows NT Server Concepts and Planning* publication.

Chapter 19. Managing servers

Managing the servers in your system consists of the following tasks:

- Starting and stopping services
- Monitoring the server
- Database administration
- Configuring and managing cache storage

Starting and stopping services

Table 2 lists the OnDemand and related services that may be installed on your system and the default properties of each service.

Table 2. OnDemand Services in Windows NT

Service	Status	Startup	Logon Account
OnDemand LibSrvr	Started	Automatic	odadmin
OnDemand Load Data		Manual	odadmin
OnDemand MVSD		Manual	odadmin
OnDemand ObjSrvr	Started	Automatic	odadmin
OnDemand Scheduler	Started	Automatic	odadmin
ADSM Server	Started	Automatic	System
ADSM Central Scheduler	Started	Automatic	System
DB2	Started	Automatic	System
DB2DAS00	Started	Automatic	odadmin
DB2-Governor		Manual	odadmin
DB2-Security Server		Manual	odadmin
MSSQLServer	Started	Automatic	odadmin
SQLExecutive		Manual	odadmin

To view and manage services, you can use the Services program in Control Panel or the Server Manager program in the Administrative Tools program group.

You can start, stop, pause, or continue any of the services by following these steps:

1. From the Control Panel program group, start the Services program.

2. In the Services dialog box, select the service in the Service list.
3. Click Start, Stop, Pause, or Continue.

If you need to modify startup parameters to a service, simply type them in the Startup Parameters field at the bottom of the Services dialog box before choosing the Start button.

Note: We recommend that you start and stop the OnDemand services from the OnDemand configurator program. When you stop an OnDemand service from the configurator, it lets you know if there are other services that depend on the server you want to stop. (And allows you to stop them.) When you start an OnDemand service from the configurator, it lets you know if there are other services that must be started before the service you want to start can be started. (And allows you to start them.)

When do you start and stop services?

If the services on your system are configured as shown in Table 2 on page 117, there is very little you need to do to maintain them. You can use Server Manager or the Services program to display current information about services, such as whether a particular service is started. If you suspect a problem with a service, look for messages in the system Event Log. Each message in the Event Log identifies the source service that generated the message.

If you need to reboot or power-off the server, the Shutdown command takes care of stopping all services in an orderly manner so that no data is lost. When you restart the server, Windows NT starts each service that is configured to start automatically.

If you need to stop OnDemand or prevent clients from accessing OnDemand, use the OnDemand configurator program to stop the OnDemand Lib/ObjSrvr, Load Data, and Scheduler services. For example, you may want to stop the OnDemand services before starting an offline backup of the database. You should also stop the OnDemand services before defining new devices or policies to ADSM, removing storage volumes from an automated library, adding or maintaining cache or database file systems, or performing program maintenance to OnDemand software.

If you make changes to the system using the configurator program, you must stop and restart the OnDemand services.

Monitoring servers

Errors and alerts

OnDemand errors and alerts

During normal processing, OnDemand tasks, administrative tools, and utilities generate messages. These messages are written to the system logging facility and sent to the system log user exit program. OnDemand assigns a severity to each message. Messages assigned a severity of alert or error are automatically sent to the Event Log.

OnDemand provides a system logging facility and a message catalog to help you identify and resolve any alerts and errors that you receive. You can open the System Log folder to list messages recorded in the system logging facility. Refer to "System logging facility" for more information. You can search the message catalog to obtain details about messages.

Application errors and alerts

For help with error messages issued by applications, refer to the following publications for a listing of error messages and associated articles:

- *ADSM Messages*
- *DB2 Messages Reference*
- *SQL Server Administrator's Companion*

Windows NT errors and alerts

Windows NT records events in three kinds of logs:

- A system log that contains events logged by the Windows NT Server system components.
- A security log that contains information about logon attempts.
- An application log that contains events logged by applications such as OnDemand, DB2, and ADSM.

Use the Event Viewer program to troubleshoot various hardware and software problems, monitor security events, and archive logs in various formats.

For more information about Windows NT Server events, see the Messages Database Help on the *Windows NT Server Resource Kit Version 4.0* compact disc.

System logging facility

OnDemand provides a logging facility to help the system administrator track OnDemand activity and monitor the system. OnDemand records messages generated by the various OnDemand programs when you enable logging for

system events, user events, and application group events. You can use one of the OnDemand client programs to search for and filter messages by time stamp, severity, message number, and user name.

Note: Before starting OnDemand for the first time, you must initialize the system logging facility. Refer to the *Installation and Customization Guide* for information about initializing the system logging facility.

Searching for and viewing messages

Log on to OnDemand with an OnDemand client program and open the System Log folder.

You can specify a time stamp to search for and retrieve messages. You can specify other search criteria, for example:

- Log Id. Each time a client logs on the server, OnDemand assigns a transaction number to that instance of the client program. All requests issued by that instance of the client program include the same Log Id.
- Userid. The OnDemand userid.
- Severity. OnDemand assigns a severity to each message: Alert, Error, Warning, Info, and Debug.
- View. Depending on the type of message in the log, you may be able to view related records or other data associated with the message from the System Log folder:
  - OnDemand records messages in the database that were issued by programs, such as the arsload command. You can display the messages created during a load by selecting an item that represents the arsload command and choosing the **View all Selected** command.
  - While most messages do not support related records or other data that can be displayed from the System Log folder, you can write a user exit program to process messages and record information about events. For example, you could write a user exit program that reports the time required to retrieve large objects from an application group. OnDemand provides a system log user exit so that you can process messages and take the appropriate action. See the *Installation and Configuration Guide* for more information.
  - Other messages in the system log do not provide additional records or other data associated with the event, for example, the Logoff event.
- Msg Num. The message number issued by OnDemand.
- Message. Enter the text of the message that OnDemand uses to restrict a search. For example, if you type **Login**, OnDemand searches for and displays the messages issued by the Logon to a Server command.

When you choose the **Search** command, OnDemand retrieves the messages from the database that match the search criteria you specified.

Monitoring users

OnDemand servers generate system log messages to help you track the number of users logged on to the server. OnDemand writes message number 201 to the system log every 30 minutes. This message contains the current number of users logged on to the server. OnDemand writes message number 202 to the system log every time the number of concurrent users exceeds the previous maximum number of concurrent users.

Monitoring performance

Note: For comprehensive documentation on monitoring Windows NT performance, see the *Windows NT Server Resource Kit Version 4.0*.

You should monitor server resources on a regular schedule. The following resources often have the most impact on server performance:

- Memory. The lack of adequate RAM results in excess paging. In general, the more RAM and cache, the better your system will perform.
- Processor. The type of processor and the number of processors affects the overall performance of the system. Windows NT Server supports symmetric multiprocessing so that if a system has multiple applications running concurrently, or applications that are multithreaded, the overall processor power is shared.
- Disk subsystem. Free space, fragmented files, and high transfer rates affect the performance of the database manager and the storage manager. Lack of adequate RAM, resulting in excess paging, can cause a disk subsystem to become very busy. In addition, the type and number of disk controllers affects the overall system responsiveness when responding to reads and writes to disk drives.
- Network subsystem. High transfer rates and otherwise overloaded networks reflect badly on client/server applications. Network I/O is the resource that most affects the performance of TCP/IP.

You can also monitor applications, such as DB2 and SQL Server.

You can choose to monitor snapshots or events. Snapshots allow you to capture point-in-time information at specified intervals. Events allow you to record information over the duration of an event, such as a connection to the database.

As part of the initial setup of your system, we recommend that you create a measurement baseline of information for use in analyzing system performance. This is important because to determine if a resource is being

overused, under used, or is at maximum capacity and performance, you need to identify what is normal. You can also use this information to set expectations of system performance for users.

Automating the collection of server data is essential to control the amount of data collected and to facilitate its analysis. You can use tools such as Performance Monitor and Network Monitor to automatically capture server data during specified periods and then store the data in a database.

Finally, you should establish and implement a plan for long-term record keeping and long-term trend analysis. If you store server performance data in a database, you can use tools such as a spreadsheet to analyze the data and generate charts and create reports.

Windows NT Performance Monitor is the tool most often used to monitor server performance. Performance Monitor performs data collection and analysis. Performance Monitor uses objects and counters to associate statistical information with monitored components. For OnDemand server analysis, we recommend that you collect information about the following objects:

- System
- Processor
- Memory
- Logical disk
- Physical disk (if using RAID)
- Server
- Cache
- Network adapter
- Database (both DB2 and SQL Server provide Performance Monitor objects and counters)

For more information about Performance Monitor, see *Microsoft Windows NT Server Version 4.0 Concepts and Planning*.

Database administration

The primary database administration tasks you need to perform are:

- Backing up the database
- Optimizing the database
- Monitoring the utilization of DB2 database file systems
- Managing DB2 archived log files
- Monitoring the utilization of the SQL Server database and transaction log
- Migrating data from the database to archive media

Completing each of these tasks on a regular schedule and taking the required action when necessary contributes to good system performance.

Backing up the database

To help protect the information that you store in OnDemand from device failures or other unforeseen hardware problems, we recommend that you backup the database on a regular schedule. OnDemand provides the arsdb command to assist you with this task. The arsdb command can be used to create full database or incremental table space backups, while other users are connected to the system or after stopping the OnDemand server services and related processes:

- To reduce time required to backup the database, you can create incremental table space backups
- To help reduce the time required to recover the database, we recommend that you periodically create full backup images of the database
- An online backup can be run when other users are connected to the database
- When you run an offline backup, the backup task is the only process connected to the database

The arsdb command can be run from the command line or configured to run on a regular schedule. You typically create the backup image on removable media, such as tape. If you plan to create the backup image on a manual tape device, the backup image must fit on one tape volume. See “Chapter 25. arsdb command reference” on page 173 for more information about the arsdb command.

Important: If you are using DB2 and you plan to back up the database to a tape device, you may need to configure the Buffer Size Limit in DB2. If you plan to backup the database to a tape device and specfiy a variable block size, you must configure the DB2 buffer size to a value that is less than or equal to the maximum block size limit for the backup device. For maximum performance, you should set the buffer size to the maximum block size for the backup device. See the DB2 UDB Version 5.2 documentation for details. Contact the IBM support center if you have questions.

Running an offline backup

To run the arsdb command:

1. Before you create an offline backup of the database, we recommend that you stop the OnDemand server services and related programs.
2. Place a blank, formatted tape in the tape device.
3. Start the OnDemand Command Line.

4. Enter the arsdb command and runtime parameters at the prompt. For example:

   ```
   arsdb -v -y \\.\Tape0
   ```

5. After the backup completes, you can restart the OnDemand server services and other related processes.

Scheduling an online backup

To schedule the arsdb command:

1. Determine the schedule on which you want the arsdb command to run. For example, every day, on Friday, at 10 p.m., and so forth.

   Important: Before the arsdb command starts, make sure there is a blank, formatted tape in the tape device.

   Note: To run scheduled tasks, the OnDemand Scheduler service must be started on the server.
2. Start the OnDemand configurator program.
3. Expand the library server where you want to run the arsdb command.
4. Select Scheduled Tasks.
5. Point to the Database Backup task and click the right mouse button.
6. From the pop-up menu, select Properties.
7. On the Task page, verify the Startup Path.
8. Select the Enabled check box.
9. On the Options page, configure the options you want to use when the backup task runs. See the online help for information on the data that you need to enter.
10. On the Schedule page, configure the schedule. See the online help for information on the data that you need to enter.
11. Click OK to activate the schedule.

Optimizing the database

We recommend that you run database optimization utilities at least once a week. If you do not run them on a regular schedule, your database performance will degrade. For example, the amount of time that it takes for a user to search for documents may increase substantially.

Note: Run database optimization utilities during a time when there is no activity on the server, including activity generated by other utilities and services. Before you run or schedule optimization utilities, see “Part 5. Command Reference” on page 133. The arsdb, arsload, and arsmaint commands should be run and scheduled with care to avoid problems,

such as locking database tables or resource contention that could cause jobs to run much longer than normal.

OnDemand provides two optimization utilities to reorganize the database:

- The arsdb command can be used to optimize the OnDemand system tables and their indexes. The arsdb command is labeled System Table Maintenance in the Scheduled Tasks list.
- The arsmaint command can be used to optimize application group tables and their indexes. The arsmaint command is labeled ApplGroup Data Maintenance in the Scheduled Tasks list.

Note: When you use SQL Server as your database manager, we recommend you use the Database Maintenance Plan Wizard to configure and schedule database maintenance rather than the tools and utilities provided with OnDemand.

Starting the optimization utilities

To start one of the optimization utilities on the server:

1. Start the OnDemand Configurator program.
2. Expand the library server where you want to run the utility.
3. Select Server Commands.
4. Point to the maintenance task that you want to run and click the right mouse button.
5. From the pop-up menu, select Properties.
6. Verify the runtime parameters in the Startup Parameters field. Refer to the Command Reference for information on the data that you need to enter.
7. Click OK.
8. Click Start. The utility will run to completion. Open the System Log folder to view messages generated by the utility.

Scheduling the optimization utilities

You can schedule the optimization utilities to run automatically at a time you set. To schedule one of the optimization utilities:

1. Start the OnDemand Configurator program.
2. Expand the library server where you want to run the utility.
3. Select Scheduled Tasks.
4. Point to the maintenance task that you want to schedule and click the right mouse button.
5. From the pop-up menu, select Properties.
6. On the Task page, verify the Startup Path.

7. Select the Enabled check box.
8. On the Options page, verify the options you want to use when the task runs. Refer to the online help for information on the data that you need to enter.
9. On the Schedule page, configure the schedule. Refer to the online help for information on how to configure the schedule.
10. Click OK to activate the schedule.

Note: To run scheduled tasks, the OnDemand Scheduler service must be started on the server.

Monitoring DB2 database file systems

You must monitor database file systems to ensure that sufficient space is available for the database as it grows. If the database manager runs out of disk space for any reason, it stops the database services. After that point, no OnDemand activity can take place until you resolve the problem.

We recommend that you set up Performance Monitor to monitor disk space and issue an alert when the available space drops below a certain threshold. At a minimum, the alert should send a message to an administrator. Depending on your requirements, the alert can cause other actions to be performed. For example, in some situations you may want to stop the OnDemand or database services to prevent running out of space on a file system. The alert could also invoke optimization utilities to migrate and expire data and attempt to reclaim space on a file system. Refer to the *Windows NT Server Resource Kit Version 4.0* for details about configuring Performance Monitor.

When you need to increase the space available for the database, you can extend existing database file systems or define new database file systems to OnDemand. Use Disk Administrator to extend existing database file systems. Use the OnDemand configurator to define new database file systems to OnDemand.

Managing DB2 log files

If you maintain archived log files on disk, OnDemand removes the log files from the archived log file directory when you backup the database with the arsdb command. This process releases the space taken by files that are no longer needed to recover the database. If you schedule database backups frequently, there is very little you need to do to maintain the archived log file directory. However, if you schedule database backups infrequently, you must make sure that the file system where you store the archived log files contains sufficient free space to hold all the log files created between database backups. DB2 issues an error message if it is unable to copy log files from the primary

log file directory to the archived log file directory. However, database processing continues. If sufficient space is not available in the archived log file directory, there is a possibility that files critical to a database backup could be lost. We recommend that you set up Performance Monitor to monitor disk space and issue an alert when the available space drops below a certain threshold. At a minimum, the alert should send a message to an administrator. Depending on your requirements, the alert can cause other actions to be performed. For example, in some situations you may want to stop the OnDemand or database services to prevent running out of space on a file system. Refer to the *Windows NT Server Resource Kit Version 4.0* for details about configuring Performance Monitor.

If you use ADSM to maintain DB2 archived log files, the ADSM policy determines when archived log files are removed. We recommend that you define the policy to maintain archived log files in ADSM storage indefinitely. However, doing so does not mean that you need to keep individual files indefinitely. We recommend that you periodically purge archived log files from ADSM storage when the files are no longer needed. For information about these tasks, see the ADSM Administrator's Guide.

Monitoring the SQL Server database and transaction log

You must monitor the storage assigned to the database to ensure that sufficient space is available for the database as it grows. If the database manager runs out of disk space for any reason, it stops the database services. After that point, no OnDemand activity can take place until you resolve the problem.

We recommend that you set up Performance Monitor to monitor disk space and issue an alert when the available space drops below a certain threshold. At a minimum, the alert should send a message to an administrator. Depending on your requirements, the alert can cause other actions to be performed. For example, in some situations you may want to stop the OnDemand or database services to prevent running out of space on a file system. The alert could also run optimization utilities that migrate and expire data and attempt to reclaim space on a file system. Refer to the *Windows NT Server Resource Kit Version 4.0* for details about configuring Performance Monitor.

When you need to increase the space available for the database, you can extend existing database devices or define new database devices to SQL Server. Use Disk Administrator to prepare the new storage volumes. Then use Enterprise Manager to extend or add devices.

You should backup the transaction log each time you backup the database. If you backup the database and not the transaction log, the transaction log is not

cleared out and you may run out of space. You may be able to clear out the transaction log with the TRUNCATE ONLY transaction log dump command option. However, if the transaction log is full, you cannot dump it with the TRUNCATE ONLY option. Also, when you dump the transaction log with the TRUNCATE ONLY option, you should immediately backup the database.

As with the database, we recommend that you set up Performance Monitor to monitor disk space and issue an alert when the available transaction log space drops below a certain threshold. You may want to configure the alert to run the DUMP DATABASE and DUMP TRANSACTION commands.

To reduce the risk of encountering full transaction logs, we recommend that you configure database and transaction log backups to run automatically on a regular schedule. We recommend that you use the Database Maintenance Plan Wizard to configure and schedule database maintenance.

Database migration

Migration is the process by which OnDemand moves index data from the database to archive media. This process optimizes database storage space while allowing you to maintain index data for as long as you require. You typically migrate index data after users no longer need to access the data, but for legal or other business requirements, you still need to maintain the data for some number of years.

Note: To perform migration, you must define a migration device and policy to ADSM and configure your application groups to migrate data. Refer to “Chapter 7. Maintaining migrated index data” on page 45 for more information.

OnDemand provides the arsmaint command to migrate data. We recommend that you run the migration utility at least once a month. If you do not run it on a regular schedule, you may need to allocate additional space for your database file systems.

Note: Run the migration utility during a time when there is no activity on the server, including activity generated by other utilities and services. The arsdb, arsload, and arsmaint commands should be run and scheduled with care to avoid problems, such as locking database tables or resource contention that could cause jobs to run much longer than normal. See “Part 5. Command Reference” on page 133 for details.

Starting the migration utility

To start the migration utility:

1. Start the OnDemand Configurator program.

2. Expand the library server where you want to run the utility.
3. Select Server Commands.
4. Point to Administrative Tasks and click the right mouse button.
5. From the pop-up menu, select Properties.
6. Enter runtime parameters in the Startup Parameters field. Refer to the Command Reference and online help for information on the data that you need to enter.
7. Click OK.
8. Click Start. The utility will run to completion. Open the System Log folder to view messages generated by the utility.

Scheduling the migration utility

You can schedule the migration utility to run automatically at a time you set. To schedule the migration utility:

1. Start the OnDemand Configurator program.
2. Expand the library server where you want to run the utility.
3. Select Scheduled Tasks.
4. Point to ApplGroup Data Maintenance and click the right mouse button.
5. From the pop-up menu, select Properties.
6. On the Task page, verify the Startup Path.
7. Select the Enabled check box.
8. On the Options page, verify the options you want to use when the task runs. Refer to the online help for information on the data that you need to enter.
9. On the Schedule page, configure the schedule. Refer to the online help for information on configuring the schedule.
10. Click OK to activate the schedule.

Note: To run scheduled tasks, the OnDemand Scheduler service must be started on the server.

Configuring and managing cache storage

You must monitor cache file systems to ensure that sufficient space is available to store new documents as well as hold existing documents. If the storage manager runs out of disk space for any reason, you will not be able to load any more data on the system, until you resolve the problem. Cache file systems exist on object servers. OnDemand caches data on the object server where the arsload command runs.

We recommend that you set up Performance Monitor to monitor disk space and issue an alert when the available space drops below a certain threshold. At a minimum, the alert should send a message to an administrator. Depending on your requirements, the alert can cause other actions to be performed. For example, in some situations you may want to stop the arsload command (if you have it running as a service) to prevent loads from failing. The alert could also invoke the arsmaint command to migrate and expire data and attempt to reclaim space on a file system. Refer to the *Windows NT Server Resource Kit Version 4.0* for details about configuring Performance Monitor.

When you need to increase the space available for the document cache, you can extend existing cache file systems or define new cache file systems to OnDemand. Use Disk Administrator to extend existing cache file systems. Use the OnDemand configurator to define new cache file systems to OnDemand.

The arsmaint command migrates documents from cache storage to archive media and expires documents from cache storage. The frequency that you run the arsmaint command and the application group storage management settings determine when documents are migrated and expired from cache storage. In general, we recommend that you schedule the arsmaint command to run every day. The arsmaint command uses values that you specify for the life of data and indexes, the length of time to cache data on magnetic, and the length of time before copying cache to archive media. Refer to *Getting Started with the Administrative Client* for information about defining application groups and storage management settings.

Starting the migration utility

To start the migration utility:

1. Start the OnDemand Configurator program.
2. Expand the library server where you want to run the utility.
3. Select Server Commands.
4. Point to ApplGroup Data Maintenance and click the right mouse button.
5. From the pop-up menu, select Properties.
6. Enter runtime parameters in the Startup Parameters field. Refer to the Command Reference and online help for information on the data that you need to enter.
7. Click OK.
8. Click Start. The utility will run to completion. Open the System Log folder to view messages generated by the utility.

Scheduling the migration utility

You can schedule the migration utility to run automatically at a time you set. To schedule the migration utility:

1. Start the OnDemand Configurator program.
2. Expand the library server where you want to run the migration utility.
3. Select Scheduled Tasks.
4. Select ApplGroup Data Maintenance and click the right mouse button.
5. From the pop-up menu, select Properties.
6. On the Task page, verify the Startup Path.
7. Select the Enabled check box.
8. On the Options page, verify the options you want to use when the task runs. Refer to the online help for information on the data that you need to enter.
9. On the Schedule page, configure the schedule. Refer to the online help for information on configuring the schedule.
10. Click OK to activate the schedule.

Note: To run scheduled tasks, the OnDemand Scheduler service must be started on the server.

Part 5. Command Reference

This section of the book contains reference information about the OnDemand administrative commands. Please read the “Overview”, beginning on page 135 for important information about interpreting command syntax.

Topic	Page
arsadm command reference	139
arsadmin command reference	153
ars_adsm command reference	165
arsdate command reference	169
arsdb command reference	173
arsdoc command reference	181
arsjesd command reference	201
arsload command reference	205
arsmaint command reference	213
arstblsp command reference	219
arsview command reference	223

Chapter 20. Overview

The commands contained in this reference include OnDemand server commands and commands that you can use to control ADSM and the database manager. The commands are presented in alphabetical order. Each command contains a description of its purpose and syntax (including descriptions of the parameters that can be used). Examples and general information about using the commands have also been included.

Examples of commands may appear in one of the following two formats:

```
/usr/lpp/ars/bin/arsmaint -cdmsv >> /tmp/arsmaint.log 2>&1

/usr/lpp/ars/bin/arsmaint -cdmsv
     >> /tmp/arsmaint.log 2>&1
```

However, when you type the command, you must enter the entire command syntax on a single line.

How to read a syntax diagram

A syntax diagram shows you how to specify a command so that the operating system can correctly interpret what you type.

Read a syntax diagram from left to right and from top to bottom, following the horizontal line (the main path). If the line ends with an arrowhead, the command syntax is continued and the next line starts with an arrowhead. Facing arrowheads mark the end of the command syntax.

When you type a command from the syntax, be sure to include punctuation, such as commas and equal signs.

Parameters are classified as keywords or variables:

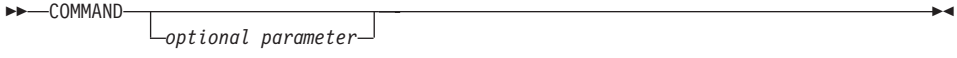
- Keywords represent constants and are shown (in syntax) in uppercase letters; however, at the command prompt, you can enter keywords in either uppercase or lowercase. An example of a keyword is a command name. On UNIX servers, command names are case sensitive. On Windows NT servers, command names are not case sensitive.
- Variables represent names or values you supply and are shown (in syntax) in lowercase letters; however, at the command prompt, you can enter variables in either uppercase or lowercase unless the syntax directions

explicitly state the case restrictions. An example of a variable is a file name. On UNIX servers, file names are case sensitive. On Windows NT servers, file names are not case sensitive.

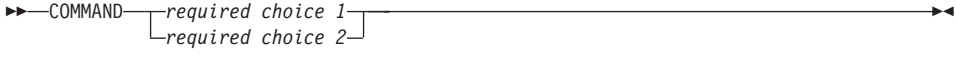
A parameter can be a combination of a keyword and a variable.

Required parameters are displayed on the main path.

Optional parameters are displayed below the main path.



A stack of parameters, with the first parameter displayed on the main path, shows that you must choose one of the parameters.



A stack of parameters, with the first parameter displayed below the main path, shows that you can choose one of the parameters.

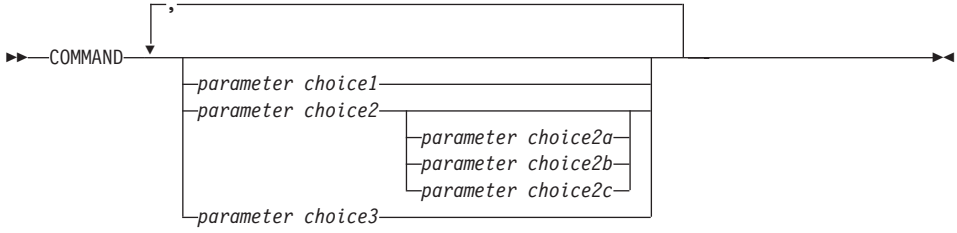
An arrow returning to the left, above the path, shows that items can be repeated following these conventions:

- If the repeat arrow contains a break, the item can be repeated in a list with the items separated by blank spaces.

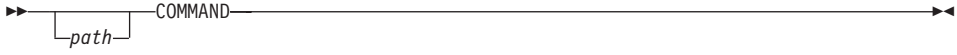
- If the repeat arrow contains a comma, the item can be repeated in a list with the items separated by commas.

You can repeat items from parameter stacks following the stack conventions for required and optional parameters described previously.

Some syntax diagrams contain parameter stacks within other parameter stacks. You can only repeat items from stacks according to the conventions described previously. That is, if an inner stack does not have a repeat arrow above it but an outer stack does, you can choose only one parameter from the inner stack and combine it with any parameter from the outer stack, and that combination can be repeated. For example, the following diagram shows that you could combine parameter choice2a with parameter choice2 and then you can repeat that combination again (choice2 plus choice2a).

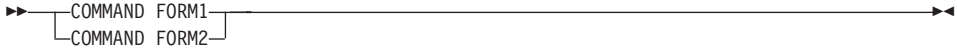


Some commands are preceded by an optional path parameter.



If you do not supply the path parameter, the system searches the current directory for the command. If the command is not in the current directory, the system continues to search for the command using the directories defined in the PATH environment variable.

Some commands in this section have several formats that accomplish the same task. These commands appear (in syntax) similar to the following:



The description of the command directs you to the correct format to use.

Chapter 21. arsadm command reference

Purpose

The arsadm command can be used to add, update, and delete users, groups, and printers.

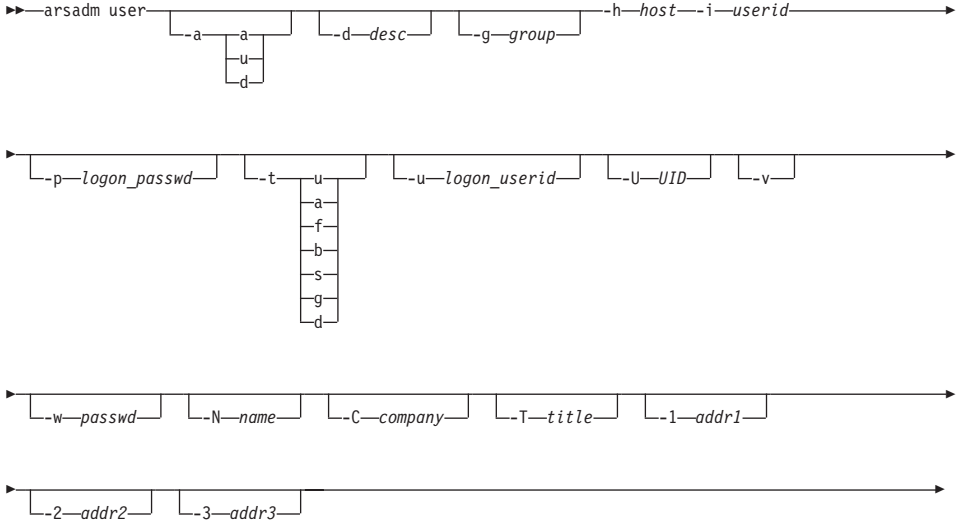
Syntax

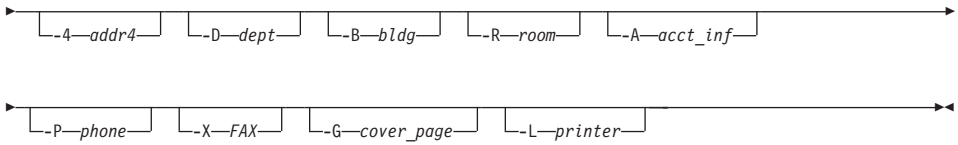
The arsadm command provides the following functions:

- USER
- GROUP
- PRINTER

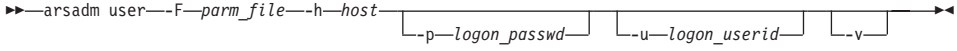
For each function, you can specify all of the required options on the command line or you can specify the name of a parameter file that contains the command options. We've listed the syntax of each function twice: first, when you do not use the parameter file option, then the syntax when you specify a parameter file (along with the syntax of the parameter file).

arsadm USER command function

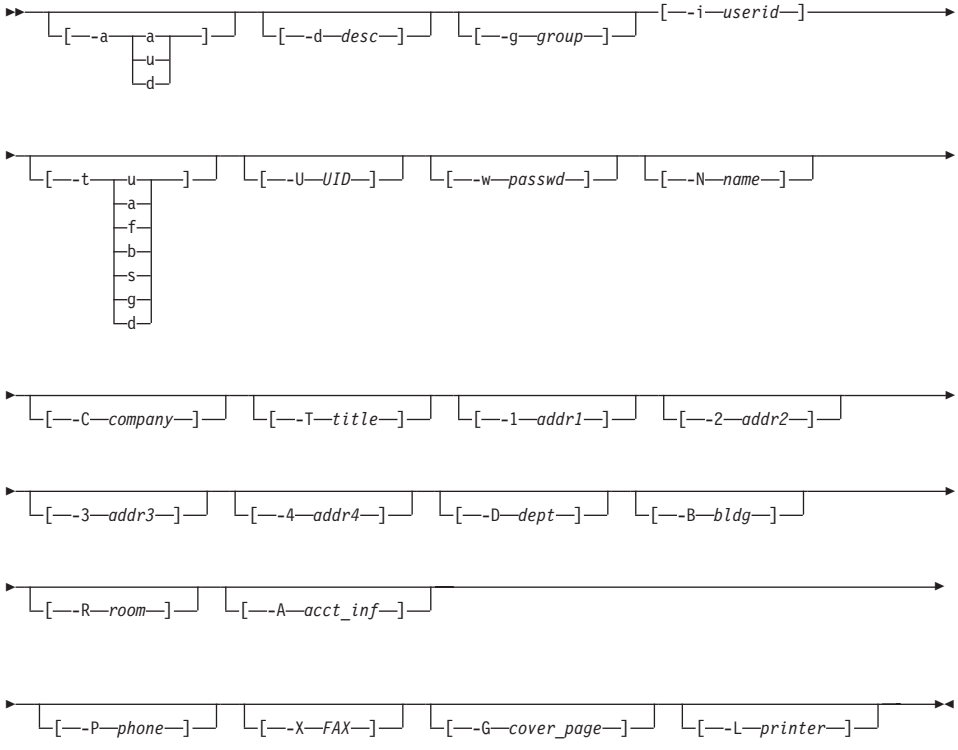




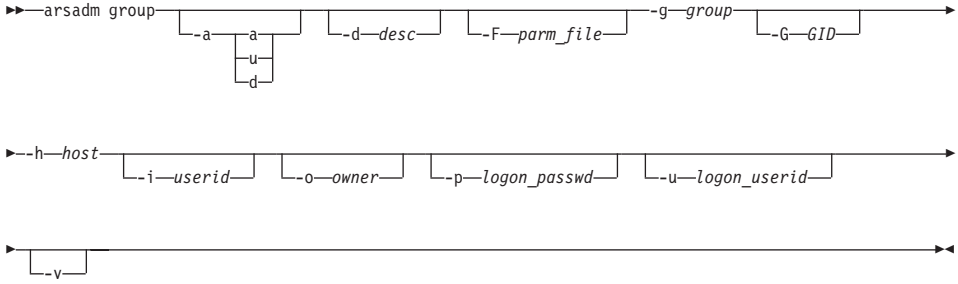
When you use the parameter file option, specify the -h, -u, -p, -v, and -F flags on the command line. If you do not specify the -u and -p flags on the command line, OnDemand prompts you to enter a userid and password that is valid for the server specified with the -h flag. All other flags and values required for the type of action that you want to perform must be specified in the parameter file. If you do not specify the -a flag, the default action is add a user. The following shows the syntax of the arsadm USER command function when you use the parameter file option:



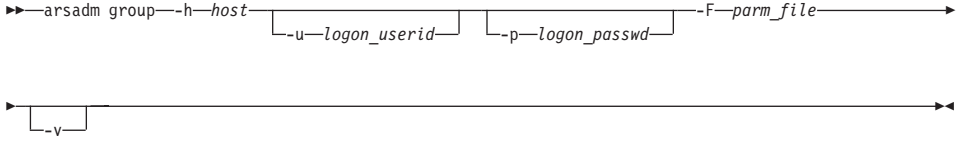
The syntax of the parameter file for the arsadm USER command function is:



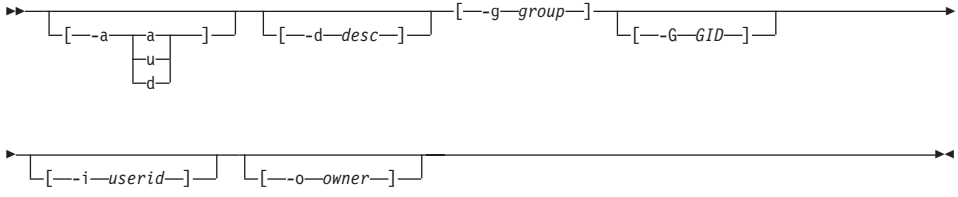
arsadm GROUP command function



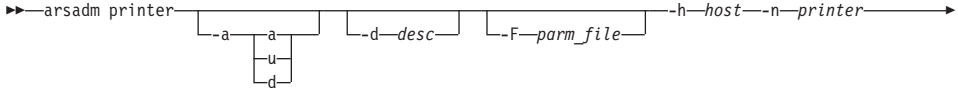
When you use the parameter file option, specify the -h, -u, -p, -v, and -F flags on the command line. If you do not specify the -u and -p flags on the command line, OnDemand prompts you to enter a userid and password that is valid for the server specified with the -h flag. All other flags and values required for the type of action that you want to perform must be specified in the parameter file. If you do not specify the -a flag, the default action is add a group. The following shows the syntax of the arsadm GROUP command function when you use the parameter file option:

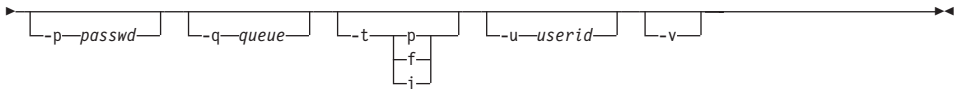


The syntax of the parameter file for the arsadm GROUP command function is:

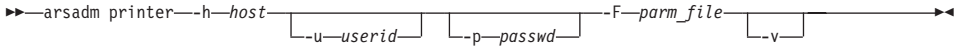


arsadm PRINTER command function

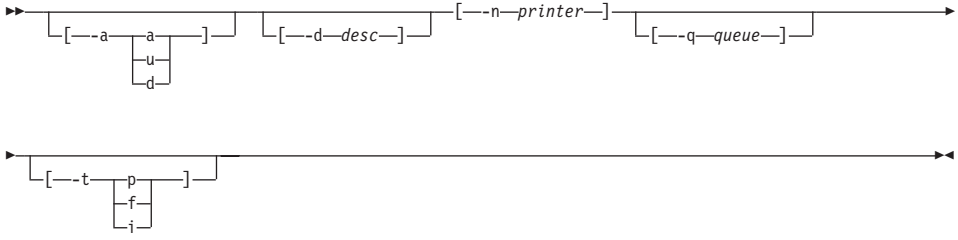




When you use the parameter file option, specify the -h, -u, -p, -v, and -F flags on the command line. If you do not specify the -u and -p flags on the command line, OnDemand prompts you to enter a userid and password that is valid for the server specified with the -h flag. All other flags and values required for the type of action that you want to perform must be specified in the parameter file. If you do not specify the -a flag, the default action is add. When you add a printer, you must specify the -q flag. If you do not specify the -t flag, the default printer type is printer. The following shows the syntax of the arsadm PRINTER command function when you use the parameter file option:



The syntax of the parameter file for the arsadm PRINTER command function is:



Description

The arsadm command is a multi-purpose program for maintaining users, groups, and printers. You can use the arsadm command to add, delete, and update users, groups, and printers. You can run the arsadm command from the command line or a user-defined program. The arsadm command provides the following functions:

user Use the user command function to add a user to OnDemand or update or delete an existing OnDemand user. You specify the type of action that you want to perform, the name of the OnDemand library server where you want to update the database, and various attributes

and values for the user that you are adding, updating, or deleting. If you do not specify the type of action, the default action is to add a user.

When you add a user:

- If you do not specify the UID, OnDemand assigns the first available UID in the database.
- If you do not specify the user type, OnDemand sets the user type to User.
- If you do not specify the name of a default printer, OnDemand sets the default printer to *NONE.
- If you do not specify a value for a user information field, OnDemand sets the field to blank.

You can specify the name of a parameter file that contains the flags and values for one or more users that you want to add, update, or delete. Each line in the file represents a different user that you want to add, update, or delete.

group Use the group command function to add a group to OnDemand or update or delete an existing OnDemand group. You specify the type of action that you want to perform, the name of the OnDemand library server where you want to update the database, and various attributes and values for the group that you are adding, updating, or deleting. If you do not specify the type of action, the default action is to add a group. You can specify the name of a parameter file that contains the flags and values for one or more groups that you want to add, update, or delete. Each line in the file represents a different group that you want to add, update, or delete.

printer

Use the printer command function to add a printer to OnDemand or update or delete an existing printer. You specify the type of action that you want to perform, the name of the library server where you want to update the database, the name of the printer, and, when you add a printer, the queue name. If you do not specify the type of action, the default action is Add. When you add a printer, you must specify the -q flag and name an AIX queue. If you do not specify the type of printer, the default printer type is Printer. You can specify the name of a parameter file that contains the flags and values for one or more printers that you want to add, update, or delete. Each line in the file represents a different printer that you want to add, update, or delete.

Parameters

-a action

The type of action that you want the arsadm command to perform. Specify **a** to add a user, group, or printer. Specify **u** to update a user, group, or printer. Specify **d** to delete a user, group, or printer.

If you do not specify the -a flag, the default action is add.

-d desc

You can enter up to 120 characters of descriptive information about the user, group, or printer. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-d** ″**This is a description**″.

-F parm_file

The name of a file that contains the flags and values for one or more users, groups, or printers that you are adding, updating, or deleting.

The flags and values in the file must be enclosed in brackets. Each line in the file typically represents a different user, group, or printer that you want to add, update, or delete. However, you can use the \ (backslash) character to continue parameters for an action to two or more lines. An action (one or more input lines) can contain a maximum of 32767 characters (bytes).

-g group

For the USER command function, use to assign the user to or remove the user from one or more groups. You can remove a user from a group only during the update action. Specify **-g group** or **-g +group** to assign the user to a group. Specify **-g -group** to remove the user from a group. If you do not specify a + (plus) or a - (minus) before the group name, the default action is to add the user to the group. You can add the user to one or more groups in a single action. For example, the following fragment:

```
-g "security +admin"
```

Causes OnDemand to add the user to the security and admin groups. If you are updating a user, you can add the user to or remove the user from one or more groups in a single action. For example, the following fragment:

```
-g "mgmt -security -admin"
```

Causes OnDemand to add the user to the mgmt group and remove the user from the security and admin groups. If you specify more than one group name, you must enclose the list of group names in double quote characters.

For the GROUP command function, the name of the group to add, delete, or update. This is a required parameter that can be specified on the command line or in the parameter file.

-G GID

For the GROUP command function, the GID (Group Identifier) to assign the group, when adding a group. If you do not specify this flag and value, OnDemand assigns the first available GID in the database. If you assign a GID, OnDemand verifies that the value that you specify is valid and is not already defined in the database.

-G cover_page

For the USER command function, the name of an AFP file that contains a user-defined FAX cover page. The file name can be from 1 to 8 characters in length.

-h host

The host name or IP address of the OnDemand library server where you want to add, delete, and update users, groups, and printers.

This is a required parameter and can only be specified on the command line.

-i userid

For the USER command function, the userid that you want to add, update, or delete. The userid can be from 1 to 20 characters in length. When you add a user, OnDemand verifies that the userid is not already defined in the database. Before updating or deleting a user, OnDemand verifies that the userid is defined in the database. This is a required parameter that can be specified on the command line or in the parameter file.

For the GROUP command function, use to assign one or more users to or remove one or more users from the group. You can remove a user from a group only during the update action. Specify **-i user** or **-i +user** to assign the user to the group. Specify **-i -user** to remove the user from the group. If you do not specify a + (plus) or a - (minus) before the user name, the default action is to add the user to the group. You can add one or more users to a group in a single action. For example, the following fragment:

```
-g csr -i "csr0001 csr0002"
```

Causes OnDemand to add both users to the csr group. If you are updating a group, you can add one or more users to or remove one or more users from the group in a single action. For example, the following fragment:

```
-g csr -i "-csr0001 -csr0002 csr0003 csr0004"
```

Causes OnDemand to remove two users from and add two users to the csr group. If you specify more than one userid, you must enclose the list of userids in double quote characters.

-n printer

For the PRINTER command function, the name of the printer. The printer name can be a maximum of 60 characters in length and can include imbedded blanks. Refer to *Getting Started with the Administrator Interface* for more information about naming printers.

-o owner

For the GROUP command function, identifies the group owner. The group owner can add users to and remove users from the group. To maintain a group's application group and folder permissions, the group owner must have administrator authority for the application groups and folders or be an application group / folder administrator or a system administrator. If you do not assign a group owner, only a system administrator can make changes to the group.

-p passwd

The password of the OnDemand userid that you specified with the **-u userid** flag. If there is no password assigned to the userid, use double quotes to show a null password. That is, specify **-p** *""*.

If you omit this flag, the arsadm command prompts you for the password when you run the command. If there is no password assigned to the userid, press the Enter key when prompted.

-q queue

For the PRINTER command function, the name of the AIX queue where OnDemand sends the print data stream. The queue name can be a maximum of 16 characters in length and must identify a valid AIX queue name. This flag is required when you add a printer.

-t type For the USER command function, the user type. If you omit this flag when you add a user, the default user type is User (u). If you omit this flag when you update a user, the user type is not changed. OnDemand supports the following user types:

u	User
a	Create application groups and applications
f	Create folders
b	Create application groups, applications, and folders
s	System administrator
g	User administrator
d	Application group and folder administrator

For the PRINTER command function, the printer type. If you omit this flag when you add a printer, the default printer type is Printer (p). If you omit this flag when you update a printer, the printer type is not changed. The valid printer types are:

p Printer. A print device.

f FAX. A FAX device.

i A special type of Printer. Lets users specify information placed on a separator page that is sent to the printer with the document.

-u userid

The OnDemand userid used to log on the server. The arsadm command verifies that the userid that you specify is valid for the server that you specify with the **-h host** flag. The arsadm command also verifies that the userid is authorized to perform the requested function. For example, only a System Administrator can add or delete groups.

If you omit this flag, the arsadm command prompts you for the userid when you run the command.

-U UID

When adding a user with the USER command function, the UID (User Identifier) assigned to the user. If you do not specify this flag and value, OnDemand assigns the first available UID in the database. If you assign a UID, OnDemand verifies that the value that you specify is valid and is not already defined in the database.

-v Enables verbose mode, to display all messages (informational and error). By default, the arsadm command displays error messages.

-w password

For the USER command function, the password for the userid that you want to add or update. The password can be from 1 to 20 characters in length. If you omit this flag when you add a user, the user is not assigned a password. To remove a user's password when you update a user, use double quotes to show a null password. That is, specify **-w** ″″.

-N name

For the USER command function, the user's name. You can enter from 1 to 60 characters of information. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-N** ″**J. Kelvin, X 5555**″.

-C company

For the USER command function, the name of the company. You can enter from 1 to 60 characters of information. If the information that

you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-C** ″**Acme Unlimited**″.

-T title

For the USER command function, the user's title or job responsibility. You can enter from 1 to 60 characters of information. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-T** ″**Special Operations**″.

-1 addr1

For the USER command function, the first of four address lines. You can enter 60 characters of information in an address line. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-1** ″**Jessica Kelvin**″.

The OnDemand CICS/ESA client program places the information that you enter in the first address line on the print separator page.

-2 addr2

For the USER command function, the second of four address lines. You can enter 60 characters of information in an address line. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-2** ″**Mail Drop G16**″.

-3 addr3

For the USER command function, the third of four address lines. You can enter 60 characters of information in an address line. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-3** ″**6124 Ruatan Place**″.

-4 addr4

For the USER command function, the fourth of four address lines. You can enter 60 characters of information in an address line. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-4** ″**Iowa City IA 77777-0001**″.

-D dept

For the USER command function, the user's department name or number. You can enter from 1 to 60 characters of information. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-D** ″**G16 Purchasing**″.

-B bldg

For the USER command function, the user's building or location. You

can enter from 1 to 60 characters of information. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-B** ″**Rusty Bucket**″.

-R room

For the USER command function, the user's room number. You can enter from 1 to 60 characters of information. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-R** ″**001 4**″.

-A acct_inf

For the USER command function, accounting or other related information. You can enter from 1 to 60 characters of information. If the information that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-A** ″**G16-0110 P95 A**″.

OnDemand includes the string that you specify in system log messages generated by the actions performed by the user. These messages are also passed to the system log user exit shell script.

-P phone

For the USER command function, the user's phone number. The phone number can be from 1 to 32 characters in length. If the phone number that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-P** ″**303 555 1212**″.

-X FAX

For the USER command function, the phone number of the user's FAX machine. The phone number can be from 1 to 32 characters in length. If the phone number that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-X** ″**303 555 1212**″.

-L printer

For the USER command function, the name of the default server printer for the user. The printer name can be from 1 to 60 characters in length. If the name that you enter contains the blank or other special characters, you must enclose the string in quotes. For example, **-L** ″**Dallas 3990**″.

Examples

1. The following example shows how to add a user:

```
arsadm user -h rockies -i csr0001 -N "Earl Roy" -v
```

2. The following example shows how to specify the name of a parameter file to add a user:

```
arsadm user -h rockies -F parmfile -v
```

The following parameter file contains the information that OnDemand uses to add the user:

```
[-i csr0001] [-N "Earl Roy"]
```

3. The following example shows how to add a System Administrator user and assign a default printer for the user:

   ```
   arsadm user -h rockies -i admin01 -t s -N "Earl Roy" -L P4028 -v
   ```

4. The following example shows how to specify the name of a parameter file to add several users:

   ```
   arsadm user -h rockies -F parmfile -v
   ```

   The following parameter file contains the information that OnDemand uses to add the users, including the name of the OnDemand group to assign the users:

   ```
   [-i csr0001] [-N "Earl Royce"] [-g "+csr"]
   [-i csr0002] [-N "Pearl Aman"] [-g "+csr"]
   [-i csr0003] [-N "June James"] [-g "+csr"]
   [-i csr0004] [-N "Grace Kelp"] [-g "+csr"]
   [-i csr0005] [-N "James June"] [-g "+csr"]
   ```

5. The following example shows how to update a user, changing the user type to User Administrator:

   ```
   arsadm user -a u -h rockies -i csr0001 -t g -v
   ```

6. The following example shows how to delete a user:

   ```
   arsadm user -a d -h rockies -i admin2 -v
   ```

7. The following example shows how to add a group:

   ```
   arsadm group -h rockies -g csr -i "csr0001 csr0002" -v
   ```

8. The following example shows how to specify the name of a parameter file to add a group:

   ```
   arsadm group -h rockies -F parmfile -v
   ```

   The following parameter file contains the information that OnDemand uses to add the group:

   ```
   [-g csr] [-i "csr0001 csr0002"]
   ```

9. The following example shows how to add a server print device:

   ```
   arsadm printer -h rockies -n "Customer Service InfoPrint 60" -q csprint1 -v
     -d "Customer Service default server printer; building 002; office I9-4"
   ```

10. The following example shows how to add a server FAX device:

    ```
    arsadm printer -h rockies -n "Customer Service FAX" -q csfax1 -t f -v
      -d "Customer Service default server FAX; building 002; office I9-4"
    ```

Notes

If a User Administrator attempts to add a System Administrator or an Application Group and Folder Administrator, the arsadm command adds the user as a (general) User. No messages are issued.

If a User Administrator attempts to update the user type of a user to a System Administrator or an Application Group and Folder Administrator, the arsadm command ignores the user type change. No messages are issued.

Files

/usr/lpp/ars/bin/arsadm
The AIX executable program.

/opt/ondemand/bin/arsadm
The HP-UX and Solaris executable program.

\Program Files\IBM\OnDemand for WinNT\bin\arsadm
The Windows NT executable program.

Chapter 22. arsadmin command reference

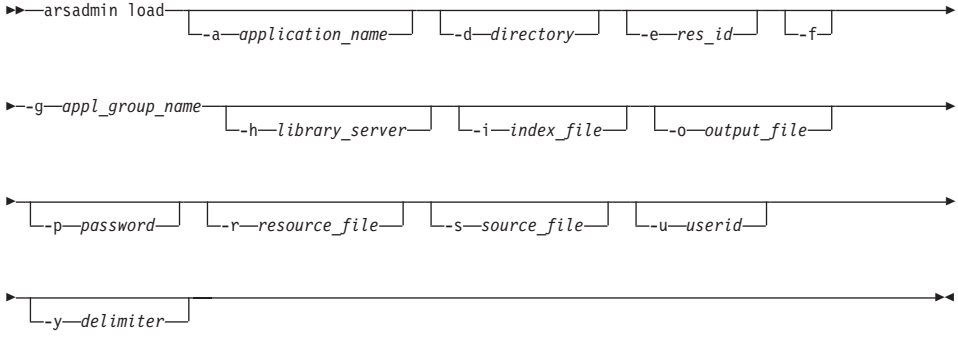
Purpose

The arsadmin command is the primary OnDemand data processing program. The arsload command calls the arsadmin command to process report, index, and resource files and create index data and storage objects. The import command function can be used to copy migrated index data from archive media to the database. The arsadmin command can also be invoked from the prompt by an experienced OnDemand administrator for problem determination and error recovery.

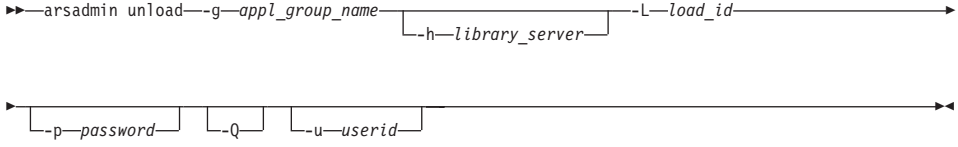
Syntax

The arsadmin command provides the following functions:

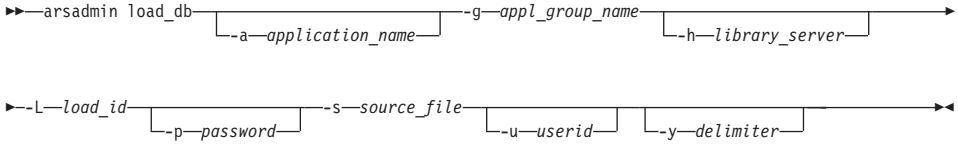
Load command function



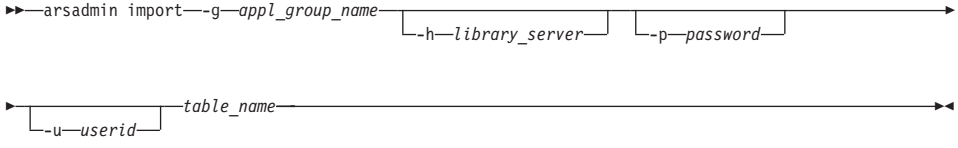
Unload command function



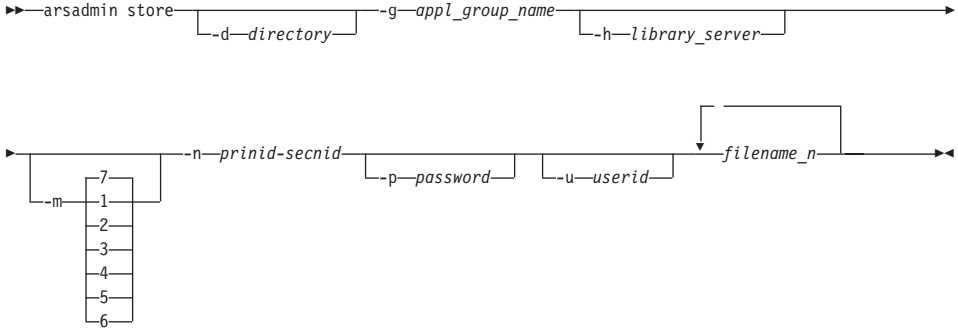
Load_db command function



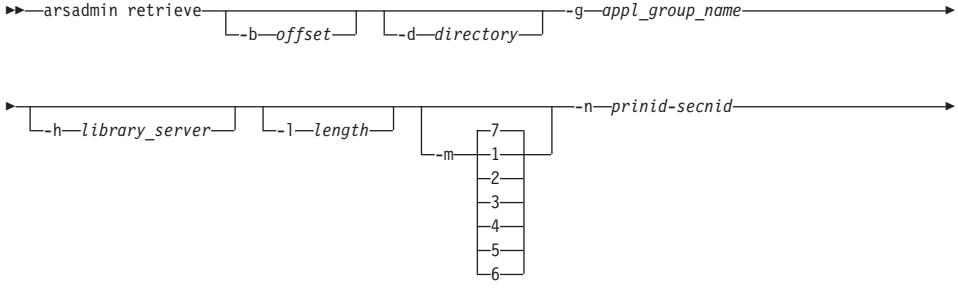
Import command function

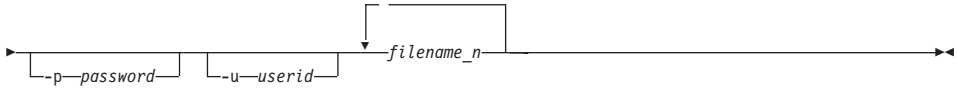


Store command function

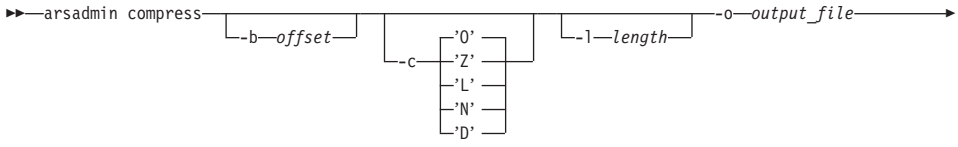


Retrieve command function

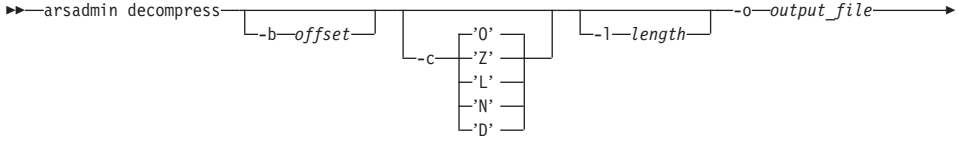




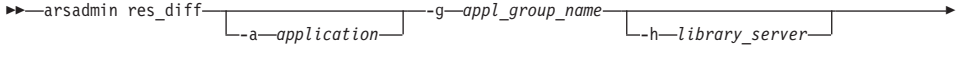
Compress command function

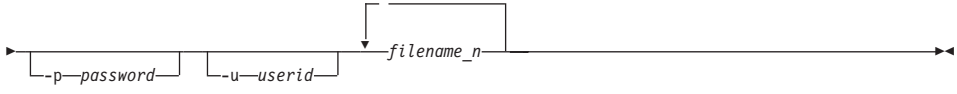


Decompress command function

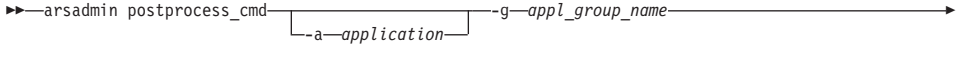


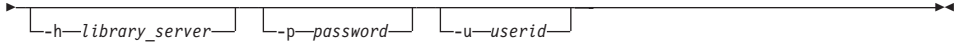
Res_diff command function



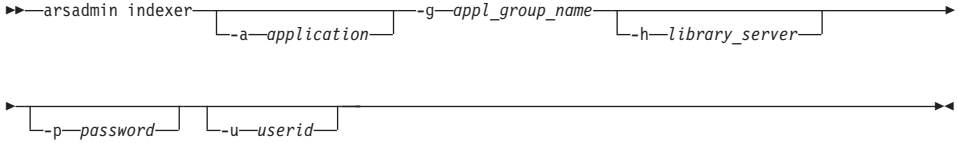


Postprocess_cmd command function





Indexer command function



Description

The arsadmin command processes the output files produced by the data indexing program (report files, index files, and optionally, resource group files) to load index data into the OnDemand database and store copies of the report files and resources in cache storage and on archive media.

The import command function can be used to import migrated index data from archive media to the database.

The arsadmin command can also be used by OnDemand administrators for testing, problem determination, and data recovery. An experienced OnDemand administrator can create, delete, and restore storage objects, retrieve individual documents and resources from storage volumes, and calculate data compression ratios.

The arsadmin command provides the following functions:

load The load command function creates the storage objects used by the OnDemand storage manager to load data in cache storage and on archive media and passes index data and parameters to the database manager to update the OnDemand database. Input data can be one or more of the following: a report file (**-s source_file**), an index file (**-i index_file**), or a resource group file (**-r resource_file**).

If you use ACIF to index a report file, you can concatenate the three types of input data into a single file. If you do so, the **-s**, **-i**, and **-r** parameters must specify the same file name.

unload

The unload command function deletes the storage objects and database rows associated with a Load ID from OnDemand. A Load ID can identify one or more storage objects. You would typically use this command function to delete the Load ID associated with a report file that you want to back out of OnDemand. You can also use this command function to delete a Load ID created during testing. The userid that you specify when you run the unload command function must have permission to delete documents from the application

group. The arsadmin command searches the system log for the specified Load ID. If the Load ID is not found in the system log, the unload fails, unless you specify the **-Q** parameter.

load_db

The load_db command function stores index data in the OnDemand database. If you specify postprocessing instructions for an application, the arsload command calls the load_db command function to store the index data in the database after postprocessing the index file created by the load command function. Index data created with the load command function can be saved to a file.

import

The import command function can be used to import migrated index data from archive media to the database. When OnDemand determines that a query requires index data that has been migrated to archive media, a message is written to the system log. An administrator must use the arsadmin command to import the required table into the database. After the index data has been imported, the user can re-enter the query. Importing migrated index data requires additional space in the database and log file storage.

store The store command function causes the OnDemand storage manager to store objects contained in a file into an application group. You can specify a mask that causes the storage manager to store the objects on one or more storage locations, that is, the primary storage node, secondary storage node, and cache storage. You would typically use this command function to copy storage objects from one location to another, for example, copy a storage object previously loaded in cache storage to archive media.

retrieve

The retrieve command function causes the OnDemand storage manager to retrieve a storage object from the specified location. You can specify a mask that tells the storage manager where to retrieve the storage object from, that is, the primary storage node, secondary storage node, or cache storage. After retrieving the storage object, you can use the decompress command function to extract a specific report file or resource from the storage object file.

compress

The compress command function can be used to compress report files and resources in preparation for creating storage objects. The compress command function can also be used to estimate how much storage space will be required to store a report file in OnDemand

To estimate the amount of storage required to store a report file that contains logical items, such as bills, invoices, or statements, compress a portion of the report file (with the **-l** parameter). Divide the size of

the output (compressed) file by the number of bytes of report file data that you compressed. The result is the compression ratio achieved.

To estimate the amount of storage required to store a report file that contains line data, such as a transaction log, extract a portion of the report file to a separate file and compress the segment of the report file. Divide the size of the output (compressed) file by the size of the segment of the report file. The result is the compression ratio achieved.

To determine the best compression method for a particular report, we recommend that you compress a sample of the data using each of the three compresson methods (LZW12, LZW16, and OD77) supplied by OnDemand. Compare the compressed file sizes and the time required to compress the data to determine the best compression method.

decompress

The decompress command function can be used to extract a specific report file or resource from a storage object file. You must first create the storage object file with the retrieve command function. The uncompressed file can be sent to a printer or displayed with the standalone viewer program.

res_diff

The res_diff command function can be used to find out if the resources contained in the specified resource group file already exist in OnDemand.

postprocess_cmd

The postprocess_cmd command function can be used to print the postprocessor statements for the specified application to stdout (UNIX servers) or the console (Windows NT servers). Postprocesser instructions can be specified when you create or update an application with the administrator interface. Refer to administrator online help for more information about how to specify postprocesser instructions for an application.

indexer

The indexer command function can be used to print the indexing parameters for the specified application to stdout (UNIX servers) or the console (Windows NT servers). Indexing parameters can be specified when you create or update an application with the administrator interface. Refer to the administrator interface online help for more information about how to specify indexing parameters for an application.

Parameters

-a application_name
The name of the OnDemand application. You must specify this parameter, if there is more than one application contained in the application group.

-b offset
For the retrieve command function, the offset (in bytes) into the specified storage object where the arsadmin command begins a partial storage object retrieval. The default is the first byte in the storage object.

For the compress command function, the offset (in bytes) from the beginning of the file where the arsadmin command begins data compression. The default is 0 (zero).

For the decompress command function, the offset (in bytes) from the beginning of the file where the arsadmin command begins data decompression. The default is 0 (zero).

-c type
For the compress and decompress command functions, the compression type for source files. Options are **'O'** for OD77 compression (the default), **'L'** for LZW12 compression, **'Z'** for LZW16 compression, **'N'** no compression, and **'D'** disable compression.

-d directory
For the load command function, if you specified the **-f** parameter to create storage object files, you can use this parameter to specify the directory where the *arsadmin* command writes the storage object files. The default is the directory where the arsadmin command was invoked.

For the store command function, the name of the directory that contains the storage object file to load. The default is the directory where the arsadmin command was invoked.

For the retrieve command function, the directory where the arsadmin command writes the file that contains the storage object retrieved from OnDemand. The default is the directory where the arsadmin command was invoked.

-e res_ID
For AFP files, this parameter and the **-r** parameter determine how OnDemand processes the resource group file. The default resource ID is 0.

- If you specify a resource ID and a resource file (**-r**), OnDemand loads the resource from the file and overrides the resource specified in the document with the resource ID you specified.

- If you specify a value of 0 (zero) and a resource file (**-r**), OnDemand checks the Server to determine if the resource specified in the document exists. If the resource does exist, OnDemand does not load the resource from the file. Otherwise, OnDemand loads the resource from the file.
- If you specify a value of 0 (zero) and do not specify a resource file (**-r**), OnDemand does not assign a resource ID to the document.
- If you specify a value other than 0 (zero) and do not specify a resource file (**-r**), OnDemand assigns the resource ID you specified to the document.

-f Causes the load command function to write storage objects to file, rather than to storage volumes. By default, OnDemand creates storage objects in memory and stores them in cache storage and on archive media. This parameter can be specified when you want OnDemand to process a report file and create storage objects, but not store the storage objects on storage volumes. You may want to use this option when you are testing the data loading process.

-g appl_group_name
The name of the application group where OnDemand processes the data.

-h library_server
The TCP/IP host name alias, fully-qualified host name, or IP address of the OnDemand library server.

You must specify this flag and the name of the OnDemand library server when you run the arsadmin command on an object server that resides on a different workstation than the library server.

-i index_file
The name of the input index file created by the data indexing program. Optionally, for AFP data, the name of a concatenated input file that also contains resource group and report data (also refer to the **-r** and **-s** parameters).

-l length
For the retrieve command function, the number of bytes that the arsadmin command retrieves from the specified storage object. The **-l** and **-b offset** parameters provide support for partial storage object retrieval.

For the compress command function, the number of bytes (from the offset) of data to compress. The default is zero bytes.

For the decompress command function, the number of bytes (from the offset) of data to decompress. The default is zero bytes.

-L load_id

A value that OnDemand generates to uniquely identify a report file load. A Load ID contains six parts. For example:

```
1220-1-0-8FAA-9339-9345
```

Where 1220 is the internal application group identifier, 1 is the primary storage node, 0 is the secondary storage node, 8FAA is a counter, 9339 is the first (internal) date in the report file, and 9345 is the last (internal) date in the report file. If you redirected the output of the arsload command or the load command function to a log file, check the log file for the Load ID created during load processing.

For the unload command function, the Load ID that you want to delete. When recovering from a system failure or some other problem that occurred during load processing, delete the last (or only) Load ID listed in the log file.

For the load_db command function, the Load ID that represents the report file load that generated the index file.

-m location

Determines the location where the storage manager processes the storage objects. Possible locations are cache storage, the primary storage node, and the secondary storage node. The default value is **7** (seven), which causes the storage manager to process the storage objects in all locations defined for the application group. Specify **1** (one) for cache storage, **2** (two) for the primary storage node, or **4** (four) for the secondary storage node. The values are additive, that is, if you specify a value of **3** (three), the storage manager processes the storage objects in cache storage and on the primary storage node.

-n prinid-secnid

The primary and secondary storage node identifiers for the application group. Separate the identifiers with the − (dash) character.

OnDemand stores the primary and secondary storage node identifiers in the database, when you load a report file into the application group. OnDemand includes the identifiers in the Load ID.

The values specified with the **-n** and **-g** parameters enable the arsadmin command to store data on or retrieve data from a specific OnDemand object server.

-o output_file

For the load command function, optionally create a file that contains the index information that the database manager uses to update the

OnDemand database. You can use this option to create the index data but not load the database, for example, if you want to test the index process or inspect the index data.

For the compress command function, the name of the output (compressed) file.

For the decompress command function, the name of the output (uncompressed) file.

-p password
The password for the OnDemand userid. If the user is not assigned a password, enter a null password (that is, `-p ""`). If you do not specify this parameter, the arsadmin command prompts you to enter the password when you run the command. If the user is not assigned a password, press Enter when prompted.

-Q For the unload command function, allows you to force the arsadmin command to unload the Load ID named with the **-L** parameter, even if the Load ID is not found in the system log. If the arsadmin command does not find the Load ID in the system log and you do not specify the **-Q** parameter, the unload fails.

-r resource_file
The name of the input resource group file. Created by the data indexing program. Optionally, for AFP data, the name of a concatenated input file that also contains index and report data (also refer to the **-i** and **-s** parameters).

-s source_file
For the load command function, the input report file or a concatenated file that contains the report file and index and resource group data (see the **-i** and **-r** parameters).

For the load_db command function, the name of the file that contains the index data to load.

For the compress command function, the name of the input, uncompressed file.

For the decompress command function, the name of the storage object file that contains the item to be decompressed.

-u userid
An OnDemand user with administrator authority for the application group. If you do not specify this parameter, the arsadmin command prompts you to enter the userid when you run the command.

For the unload command function, the userid that you specify must have permission to delete documents from the application group.

-y delimiter

The column delimiter. Optionally specify the character that separates the columns (fields) in the database rows contained in the index file passed to the database manager. The default delimiter is the "¦" (pipe symbol).

OnDemand uses the delimiter when you specify the **-o** flag to write the database input records to file.

filename_n

For the store command function, the names of one or more storage object files to load.

For the retrieve command function, the names of one or more storage objects to retrieve. Also the names of the files that the arsadmin command creates in the **-d** directory.

For the res_diff command function, the names of one or more resource group files to process.

table_name

For the import command function, the name of the migrated index table to import into the database.

Examples

1. The following example shows how to invoke the load command function to load report file data contained in the `credit.out` file and index data contained in the `credit.ind` file into the `Credit` application group.

   ```
   arsadmin load -g 'Credit' -i credit.ind -s credit.out -r credit.res
   ```

2. The following example shows how to invoke the load command function to load a concatenated input file that contains the report file, index data, and resources.

   ```
   arsadmin load -g 'Credit' -i credit.cat -s credit.cat -r credit.cat
   ```

3. The following example shows how to invoke the unload command function to delete a Load ID from OnDemand.

   ```
   arsadmin unload -g 'Credit' -L 1220-1-0-1FAA-9339-9345
   ```

4. The following example shows how to use the compress command function to estimate the storage required for a report file that contains logical items, such as bills, invoices, or statements. The arsadmin command compresses the first 200 KB of data in the input file using the default (OD77) compression method.

   ```
   arsadmin compress -s report_file -o output_file -l 200000
   ```

5. The following example shows how to use the compress command function to estimate the storage required for report files that contain transaction data, such as a general ledger. The input file is a portion of a larger report file.

```
arsadmin compress -s report_file_segment -o output_file
```

6. The following example shows how to use the import command function to import a migrated index table from archive media into the database. The name of the application group and the index table to import can be obtained from the message written to the system log.

```
arsadmin import -g 'Credit' SL27
```

Files

/usr/lpp/ars/bin/arsadmin
The AIX executable program.

/opt/ondemand/bin/arsadmin
The HP-UX and Solaris executable program.

\Program Files\IBM\OnDemand for WinNT\bin\arsadmin
The Windows NT executable program.

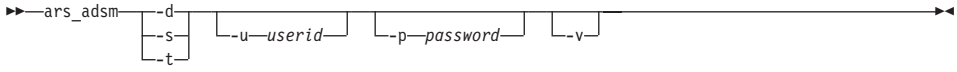
Chapter 23. ars_adsm command reference

Purpose

Provides an interface to ADSM for specific functions, such as starting and stopping the ADSM server and creating a backup image of the ADSM database.

Note: The ars_adsm command is available on UNIX servers.

Syntax



Description

The ars_adsm command provides a command line interface to ADSM for the following functions:

- Start the ADSM server as a background process.
- Stop the ADSM server.
- Create a backup image of the ADSM database.

When you perform ADSM administrative tasks, you must log on to the server with administrative userid and password. A set of default values for the userid and password are set in the ars_adsm file. You can use the values in the file or specify different values using the **-u** and **-p** command line parameters. The *Installation and Configuration Guide* describes how to change the ADSM administrative userid and password in the ars_adsm file.

Parameters

-d Create a backup image of the ADSM database. Writes the backup image to the backup device that you configured when you installed and configured ADSM. The *Installation and Configuration Guide* provides details about configuring the database backup device.

If you plan to create a backup image of the database on tape, you must mount a labeled, ADSM tape storage volume in the tape drive before the ars_adsm command starts.

-s Start the ADSM server as a background process.

-t Stop the ADSM server.

-u userid
An ADSM administrative userid.

-p password
The password for the ADSM administrative userid specified with the -u flag. If the user is not assigned a password, enter a null password. For example, `-p ""`.

-v Enables verbose mode, to display all messages (informational and error). By default, the ars_adsm command displays error messages.

Examples

1. To start the ADSM server, enter:

   ```
   ars_adsm -sv
   ```

2. To stop the ADSM server, enter:

   ```
   ars_adsm -tv
   ```

3. To create a backup image of the ADSM database, enter:

   ```
   ars_adsm -dv
   ```

   Note: If you backup the database to tape, you must place a formatted ADSM tape storage volume in the drive before you start the ars_adsm command.

Notes

OnDemand server programs can remain active while you create a backup copy of the ADSM database.

If you stop ADSM while the OnDemand server daemon is active, OnDemand issues an error message if a client attempts to retrieve an item that resides on a storage volume managed by ADSM.

Files

/usr/lpp/ars/bin/ars_adsm
The AIX executable program.

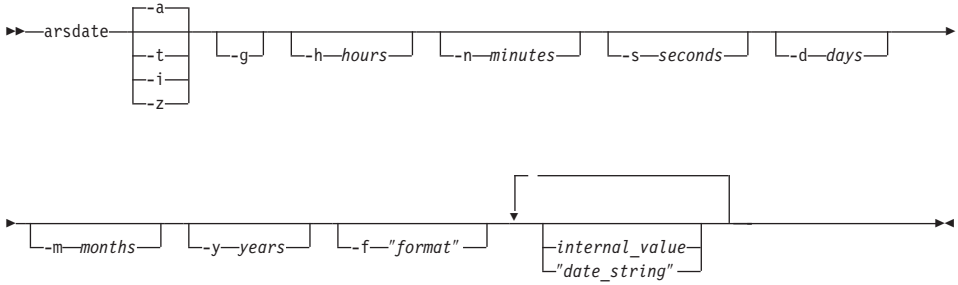
/opt/ondemand/bin/ars_adsm
The HP-UX and Solaris executable program.

Chapter 24. arsdate command reference

Purpose

Display the OnDemand internal date value for a given date string or display the date string for a given OnDemand internal date value.

Syntax



Description

The arsdate command can be used to display a date string using the internal date value that OnDemand stores in the database. You can also use the arsdate command to display the internal value for a given date string.

Parameters

-a Display the OnDemand internal date format for the specified date string.

-t Display the OnDemand internal time format for the specified time string.

-i Display the OnDemand internal date and time format for the specified date and time string.

-z Display the OnDemand internal date and time format for the specified date and time string using the time zone of the server.

-g Display the current date.

-d days
Add the specified number of days to the specified internal value.

-h hours
Add the specified number of hours to the specified internal value.

-m months
Add the specified number of months to the specified internal value.

-n minutes
Add the specified number of minutes to the specified internal value.

-s seconds
Add the specified number of seconds to the specified internal value.

-y years
Add the specified number of years to the specified internal value.

-f "format"
Determines the format of the date and time string that OnDemand displays.

The following table lists some of the formats supported by OnDemand. If the format that you require is not listed, you can create your own, using values, separators, and the percent character.

The format consists of a set of values (for example, **%m**) and separators (such as the blank character).

Table 3. Date and Time Format Specifications

Date Format Specifier	Date Format	Example
%m/%d/%y	mm/dd/yy	01/31/95
%d/%m/%y	dd/mm/yy	31/01/95
%f/%e/%y	m/d/yy	1/31/95
%e/%f/%y	d/m/yy	31/1/95
%m-%d-%y	mm-dd-yy	01-31-95
%d-%m-%y	dd-mm-yy	31-01-95
%m%d%y	mmddyy	013195
%m%d%Y	mmddyyyy	01311995
%y%m%d	yymmdd	950131
%Y%m%d	yyyymmdd	19950131
%b %e, %Y	Mth d, yyyy	Jan 31, 1995
%B %e, %Y	Month d, yyyy	January 31, 1995
%m/%d/%y %H:%M	mm/dd/yy hh:mm	01/31/95 10:50
%H:%M	hh:mm	10:50

Table 3. Date and Time Format Specifications (continued)

Date Format Specifier	Date Format	Example
%T	hh:mm:ss	10:50:59

internal_value
The internal date value from the OnDemand database. Enter one or more internal date values when you want to display formatted date strings.

″**date_string**″
The date string. Enter one or more date strings when you want to display internal date values.

Examples

1. To obtain the date in the default format for a given internal value, enter:

```
arsdate 9404
```

   The arsdate command displays:

```
9404 -> 09/30/95
```

2. To obtain the dates in the default format for a given set of internal values, enter:

```
arsdate 9404 9374 9343
```

   The arsdate command displays:

```
9404 -> 09/30/95
9374 -> 08/31/95
9343 -> 07/31/95
```

3. To obtain the date using a specific format for a given internal value, enter:

```
arsdate -f "%B %e, %Y" 9404
```

   The arsdate command displays:

```
9404 -> September 30, 1995
```

4. To obtain the current date, enter:

```
arsdate -g
```

   The arsdate command displays:

```
Today:  07/24/95
```

5. To obtain the internal date for July 24, 1995, enter:

```
arsdate -a 7/24/95
```

   The arsdate command displays:

```
7/24/95 -> 9336
```

6. To obtain the internal dates for a set of given dates, enter:

```
arsdate -a 7/31/95 8/31/95 9/30/95
```

The arsdate command displays:

```
7/31/95 -> 9343
8/31/95 -> 9374
9/30/95 -> 9404
```

Files

/usr/lpp/ars/bin/arsdate
The AIX executable program.

/opt/ondemand/bin/arsdate
The HP-UX and Solaris executable program.

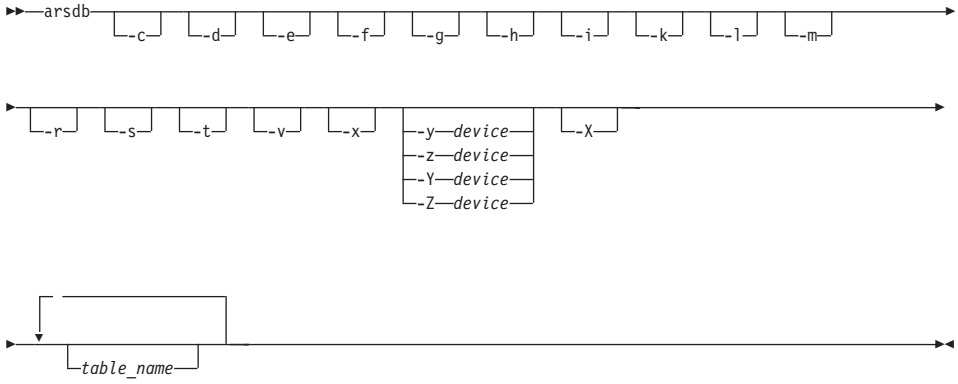
\Program Files\IBM\OnDemand for WinNT\bin\arsdate
The Windows NT executable program.

Chapter 25. arsdb command reference

Purpose

Provides an interface to the database manager for specific OnDemand database functions, such as creating the database, starting and stopping the database manager, creating a backup image of the database, and reorganizing OnDemand system tables. The OnDemand database resides on the library server.

Syntax



Description

The arsdb command provides an interface to the database manager for the following database functions:

- Create and initialize the OnDemand database.
- Start the database manager.
- Stop the database manager.
- Create backup images of the OnDemand database. You can use the arsdb command to create backup images of table spaces and the full database. When you run the arsdb command to create backup images of DB2 table spaces and databases, you can specify that you want ADSM to manage the backup images. However, you must first properly configure ADSM. See the *Installation and Configuration Guide* for details.

- Reorganize and optimize OnDemand system tables.
- Import and export tables and databases.

The arsdb command reads the ars.cfg file (UNIX servers) or the Registry (Windows NT servers) to obtain database configuration information. Before creating the database, verify, and if necessary, change the database configuration information. The *Installation and Configuration Guide* provides details.

Parameters

-c Create the (initial) OnDemand database structure.

The **-c** parameter implies the **-r**, **-s**, and **-t** parameters.

-d Drop tables from the database. The default is all tables. To drop a specific table, include the **tablename** parameter, where **tablename** is the name of the OnDemand system table.

-e Drop indexes from the database. The default is all indexes. To drop a specific index, include the **tablename** parameter, where **tablename** is the name of the OnDemand system table.

-f Ignore database manager failures.

-g Start the database manager.

-h Stop the database manager.

-i Import tables. The default is all OnDemand system tables. To import a specific table, include the **tablename** parameter, where **tablename** is the name of the OnDemand system table and the name of the file that contains the table to be imported. OnDemand reads the file from the directory where the arsdb command was invoked.

-k Activate the connection to the OnDemand database.

-l Import or export tables in the IXF file format instead of the DEL file format (the default).

The IXF file format is the PC version of the Integrated Exchange Format. A DEL file is a sequential ASCII file with row and column delimiters.

This flag is valid only with the **-i** and **-x** flags.

-m Perform maintenance on the OnDemand database, reorganizing the OnDemand system tables. This option refreshes the tables and optimizes access to information in the database. The system must be idle when you run the arsdb command with this option.

-r Create configuration indexes. The default is all indexes. To configure a

specific index, include the **tablename** parameter, where **tablename** is the name of the OnDemand system table.

-s Run database statistics. This parameter is used to optimize indexes and tables and make access to information as efficient as possible. The default is all OnDemand system tables. To run statistics for a specific table, use the *-s tablename* form of the command, where *tablename* is the name of the OnDemand system table.

-t Create configuration tables. The default is all OnDemand system tables. To configure a specific table, use the *-t tablename* form of the command, where *tablename* is the name of the OnDemand system table.

-v Enables verbose mode, to display all messages (informational and error). By default, the arsdb command displays error messages.

-x Export tables. The default is all OnDemand system tables. To export a specific table, use the *-x tablename* form of the command, where *tablename* is the name of the OnDemand system table and the name of the file OnDemand creates. OnDemand writes the file in the directory where the arsdb command was invoked.

-y device

Create an offline backup image of the OnDemand database on the specified device.

Before you start an offline backup of the database, you must make sure that no other users or processes are connected to the OnDemand database. We recommend that you run the offline backup at a time when you are certain no users will access OnDemand or stop the OnDemand server programs and other related processes before you start the offline backup.

If you specify a tape device, you must place a blank tape volume in the device before starting the arsdb command.

Note: When using the arsdb command to backup the database to a manual tape device, the backup image must fit on one tape volume. If the backup image requires more than one tape volume, you must use an automated tape library.

If the specified device is ADSM, it means that ADSM maintains the database backup image file. However, before you use the arsdb command to create backup images to ADSM-managed storage, you must properly configure ADSM. See the *Installation and Configuration Guide* for details. The *DB2 UDB Administration Guide* also contains important information about ADSM support and applicability for your system.

-z device

Create an online backup image of the OnDemand database on the specified device.

An online backup of the database can run while other users and processes are connected to the database. That is, other users and applications can read and modify the database while the backup process is running.

If you specify a tape device, you must place a blank tape volume in the device before starting the arsdb command.

Note: When using the arsdb command to backup the database to a manual tape device, the backup image must fit on one tape volume. If the backup image requires more than one tape volume, you must use an automated tape library.

If the specified device is ADSM, it means that ADSM maintains the database backup image file. However, before you use the arsdb command to create backup images to ADSM-managed storage, you must properly configure ADSM. See the *Installation and Configuration Guide* for details. The *DB2 UDB Administration Guide* also contains important information about ADSM support and applicability for your system.

-X Do not backup the USERSPACE1 table space. The USERSPACE1 table space is created by DB2 when you initialize the OnDemand database. It contains system tables and application group tables that have not been migrated to their own table spaces. This parameter is only valid with the -Y and -Z flags.

A typical use of this parameter is to test the table space backup process. For example, assume that you migrated a table of application group data to its own table space. All other tables of application group data remain in the USERSPACE1 table space. To create an online backup image of the migrated application group table, run the arsdb command with the –XZ parameters.

-Y device

Create an incremental, offline backup image of table spaces on the specified device. An incremental backup means that OnDemand backs up only those tables that have changed since the last time they were backed up. OnDemand keeps track of table space backups and can determine which table spaces have changed and need to be backed up.

Before you start an offline backup of the database, you must make sure that no other users or processes are connected to the OnDemand database. We recommend that you run the offline backup at a time

when you are certain no users will access OnDemand or stop the OnDemand server programs and related processes before you start the offline backup.

If you specify a tape device, you must place a blank tape volume in the device before starting the arsdb command.

If the specified device is ADSM, it means that ADSM maintains the backup image file. However, before you use the arsdb command to create backup images to ADSM-managed storage, you must properly configure ADSM. See the *Installation and Configuration Guide* for details. The *DB2 UDB Administration Guide* also contains important information about ADSM support and applicability for your system.

-Z device

Create an incremental, online backup image of table spaces on the specified device. An incremental backup means that OnDemand backs up only those tables that have changed since the last time they were backed up. OnDemand keeps track of table space backups and can determine which table spaces have changed and need to be backed up.

An online backup of the database can run while other users and processes are connected to the database. That is, other users and applications can read and modify the database while the backup process is running.

If you specify a tape device, you must place a blank tape volume in the device before starting the arsdb command.

If the specified device is ADSM, it means that ADSM maintains the backup image file. However, before you use the arsdb command to create backup images to ADSM-managed storage, you must properly configure ADSM. See the *Installation and Configuration Guide* for details.The *DB2 UDB Administration Guide* also contains important information about ADSM support and applicability for your system.

table_name

Optionally specify the name of one or more OnDemand system tables for the drop (-d and -e), import (-i), create configuration (-r and -t), run statistics (-s), and export (-x) parameters. If you do not specify the **tablename** parameter, OnDemand runs the command for all OnDemand system tables.

For the import (-i) parameter, OnDemand reads the imported tables from a file in the directory where arsdb was invoked. For the export (-x) parameter, OnDemand writes the exported tables to a file in the directory where arsdb was invoked.

Examples

1. The following shows how to start the database manager:

   ```
   arsdb -gkv
   ```

2. The following shows how to stop the database manager:

   ```
   arsdb -hv
   ```

3. The following shows how to reorganize all the OnDemand system tables in the database and run the database statistics command, which optimizes access to the tables:

   ```
   arsdb -msv
   ```

4. The following shows how to reorganize a specific OnDemand system table. In the example, the arsdb command reorganizes the indexes for the Application Group table:

   ```
   arsdb -mv arsag
   ```

5. The following shows how to export a specific OnDemand system table to an IXF format file:

   ```
   arsdb -xl arsag
   ```

6. The following shows how to create an offline backup image of the OnDemand database on tape:

   UNIX servers

   ```
   arsdb -v -y /dev/rmt0
   ```

   Windows NT servers

   ```
   arsdb -v -y \\.\Tape0
   ```

   Note: You must place a blank, formatted tape in the device before you start the arsdb command. If the device is a manual tape device, the backup image must fit on one tape volume.

7. The following shows how to create an online, incremental backup of table spaces in the OnDemand database. The backup image is written to storage managed by ADSM.

   ```
   arsdb -v -Z ADSM
   ```

8. The following shows how to create an offline, incremental backup of table spaces in the OnDemand database. The backup image is written to storage managed by ADSM.

   ```
   arsdb -v -Y ADSM
   ```

9. The following shows how to create an online, full backup of the OnDemand database. The backup image is written to storage managed by ADSM.

   ```
   arsdb -v -z ADSM
   ```

10. The following shows how to create an offline, full backup of the OnDemand database. The backup image is written to storage managed by ADSM.

```
arsdb -v -y ADSM
```

Notes

The arsdb command should be run only on the library server.

We recommend that you stop the OnDemand server programs and related processes before you run the arsdb command with the create (-c), import (-i), reorganize (-m) or offline database backup (-y, -Y) parameters. “Stopping OnDemand” on page 91 and “Starting and stopping services” on page 117 provide details about stopping OnDemand programs.

The OnDemand server daemon (or service) can remain active if you need to stop the database manager. However, OnDemand records an error message in the system log and the OnDemand client program issues an error message whenever the database manager is stopped and a user attempts to query the database.

The online database backup command (-z, -Z parameters) can be run while other applications are connected to the database.

We recommend that you do not run the arsload command at the same time that you run the arsdb command with the -m or -s options.

If you plan to back up the database to a tape device you may need to configure the Buffer Size Limit in DB2. If you plan to backup the database to a tape device and specfiy a variable block size, you must configure the DB2 buffer size to a value that is less than or equal to the maximum block size limit for the backup device. For maximum performance, you should set the buffer size to the maximum block size for the backup device. See the DB2 UDB Version 5.2 documentation for details. Contact the IBM support center if you have questions.

Files

/usr/lpp/ars/bin/arsdb
The AIX executable program.

/opt/ondemand/bin/arsdb
The HP-UX and Solaris executable program.

\Program Files\IBM\OnDemand for WinNT\bin\arsdb
The Windows NT executable program.

Chapter 26. arsdoc command reference

Purpose

The arsdoc command can be used to generate a list of items that match a query, retrieve documents, add, delete, and update documents, and print documents using the OnDemand server print facility.

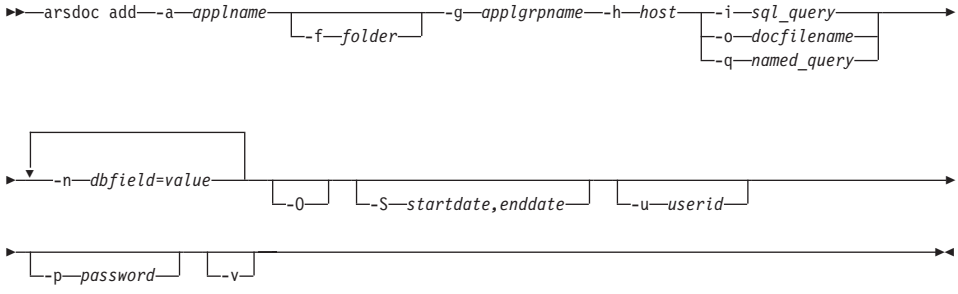
Syntax

The arsdoc command provides the following functions:

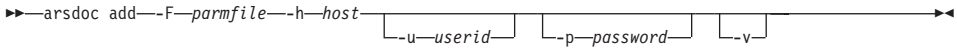
- ADD
- DELETE
- GET
- PRINT
- QUERY
- UPDATE

For each function, you can specify all of the required options on the command line or you can specify the name of a parameter file that contains the command options. We've listed the syntax of each function twice: first, when you do not use the parameter file option, then the syntax when you specify a parameter file (along with the syntax of the parameter file).

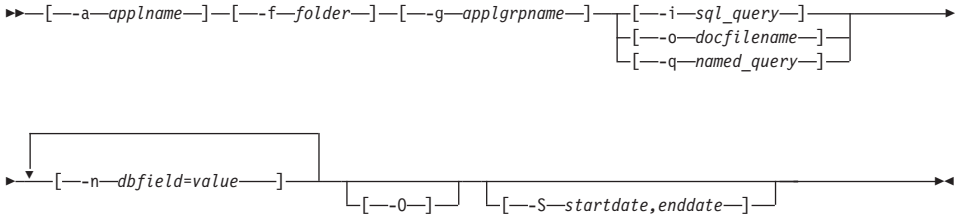
arsdoc ADD command function



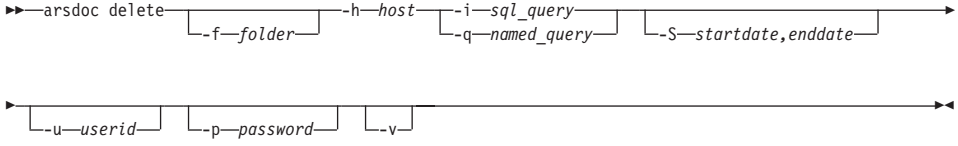
When you use the parameter file option, you must not specify the -a, -f, g, -i, -n, -o, -O, q, or -S flags on the command line. The following diagram shows the syntax of the arsdoc ADD command function when you use the parameter file option:



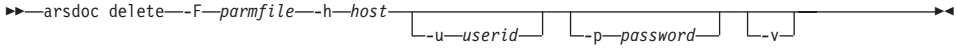
The syntax of the parameter file for the arsdoc ADD command function is:



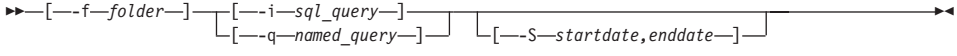
arsdoc DELETE command function



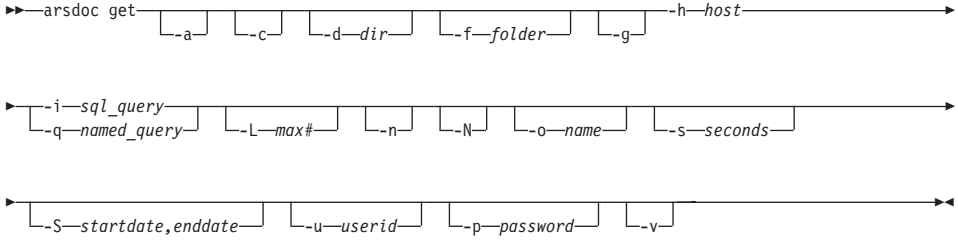
When you use the parameter file option, you must not specify the -f, -i, -q, or -S flags on the command line. The following diagram shows the syntax of the arsdoc DELETE command function when you use the parameter file option:



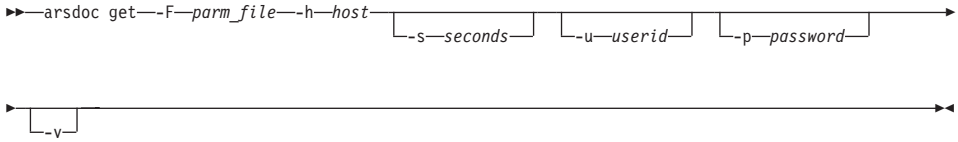
The syntax of the parameter file for the arsdoc DELETE command function is:



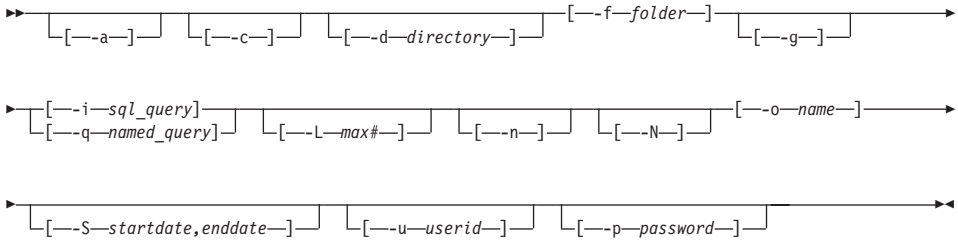
arsdoc GET command function



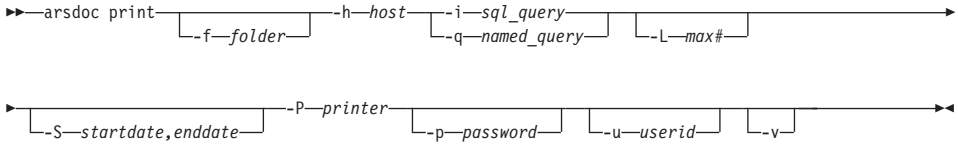
When you use the parameter file option, you must not specify the -a, -c, -d, -f, -g, -i, -L, -n, -N, -o, -q, or -S flags on the command line. The following diagram shows the syntax of the arsdoc GET command function when you use the parameter file option:



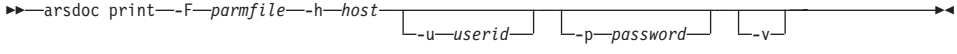
The syntax of the parameter file for the arsdoc GET command function is:



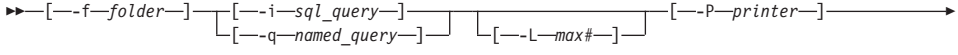
arsdoc PRINT command function

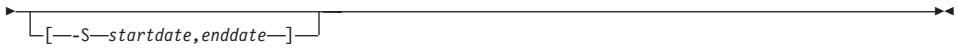


When you use the parameter file option, you must not specify the -f, -i, -L, P, -q, or -S flags on the command line. The following diagram shows the syntax of the arsdoc PRINT command function when you use the parameter file option:

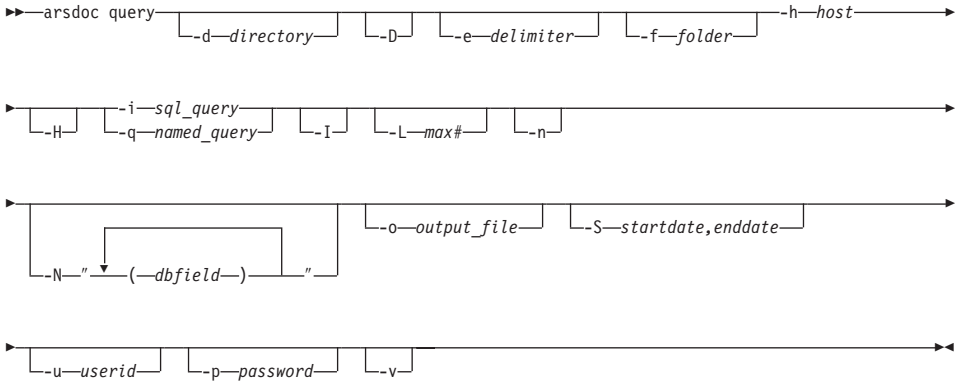


The syntax of the parameter file for the arsdoc PRINT command function is:

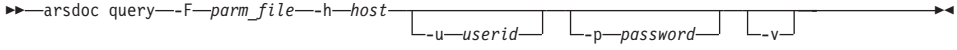




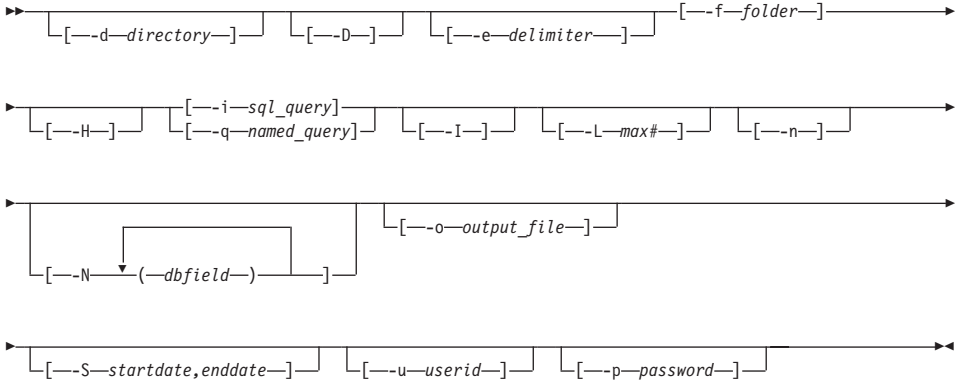
arsdoc QUERY command function



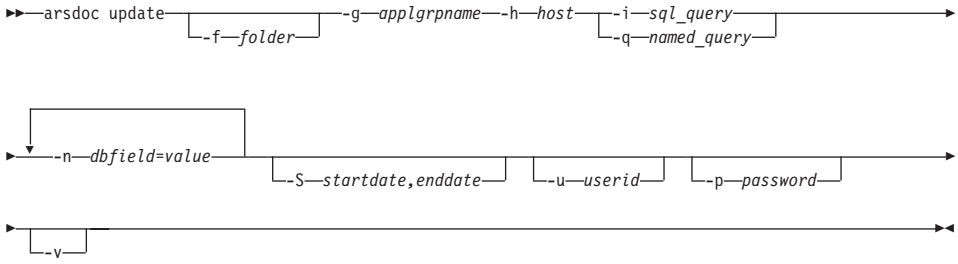
When you use the parameter file option, you must not specify the -d, -D, e, -f, -H, -i, -I, L, -n, -N, -o, -q, or -S flags on the command line. The following shows the syntax of the arsdoc QUERY command function when you use the parameter file option:



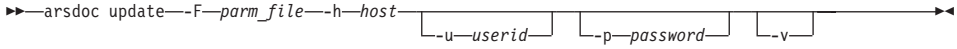
The syntax of the parameter file for the arsdoc QUERY command function is:



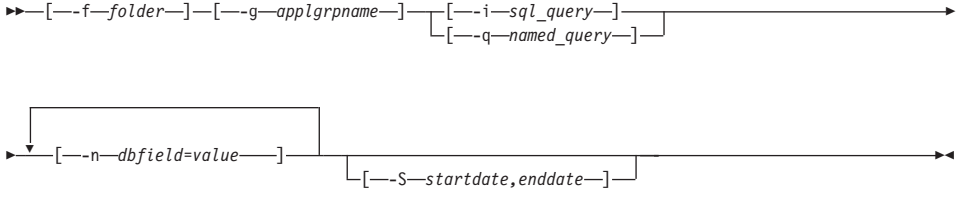
arsdoc UPDATE command function



When you use the parameter file option, you must not specify the -f, -g, i, -n, -q, or -S flags on the command line. The following shows the syntax of the arsdoc UPDATE command function when you use the parameter file option:



The syntax of the parameter file for the arsdoc UPDATE command function is:



Description

The arsdoc command is a multi-purpose document processing program. You can use the arsdoc command to query the library server and generate a list of items that match a query, retrieve documents from the system, add, delete, and update documents, and send documents to the server print facility. You can run the arsdoc command from the command line or a user-defined program. The arsdoc command provides the following functions:

add Use the add command function to store data in OnDemand, by specifying the folder, application group, application, and database fields and values. If you specify the **-O** flag, the database fields are optional, except for date and time fields. When you specify the **-O** flag, the arsdoc command stores a default value in any other database field you omit. The default value for string fields is an empty string. The default value for numeric fields is 0 (zero). Numeric fields include integer and decimal fields. When adding a document, you can specify

the data in one of three ways: (1) a document file, (2) an SQL query that contains clauses, database field names, index values, and operators, or (3) a public named query.

When you use the add command function to add a row for an existing document, the row must be added to the application group that contains the document. The application group must be specified with the **-g** flag and must be one of the application groups referenced by the folder named with the **-f** flag.

Important: The arsdoc add command will fail unless the application group Database Organization is Multiple Loads per Database Table and Expiration Type is Segment or Document.

delete Use the delete command function to delete documents from OnDemand. To identify the documents you want to delete, you must enter an SQL query that contains clauses, database field names, index values, and operators or a public named query. The delete command function deletes all documents that match the query.

get Use the get command function to retrieve documents and their resources from OnDemand. The get command function can also generate and save generic index data for the documents that match the query. You identify the name of an OnDemand library server, the name of the folder that you want to search, and specify one or more queries to run. To query the database, you can enter an SQL query that contains clauses, database field names, index values, and operators or you can enter the name of a public named query. By default, the arsdoc command sends a copy of the documents that match the SQL query to stdout (UNIX) or the console (Windows NT). You typically redirect the output to a printer or another process. You can also choose to write the output to a file.

print Use the print command function to send items to the OnDemand server print facility. You identify the name of an OnDemand library server, the name of the folder that you want to search, and specify the query to run. The items that match the query are sent to the server printer named with the -P flag. The server printer must be defined to OnDemand using the administrator interface. To query the database, you can enter an SQL query that contains clauses, database field names, index values, and operators or you can enter the name of a public named query. You can limit the number of documents sent to the printer by using the -L flag and specifying the maximum number of documents that should be retrieved, regardless of the number of documents that match the query. You can limit the number of database tables searched, and possibly increase the performance of a query, by using the -S flag and specifying a start date and an end

date. The print command function does not currently support sending items to the OnDemand server FAX facility.

query Use the query command function to generate a list of items that match a query. You identify the name of an OnDemand library server, the name of the folder that you want to search, and the query to run. To query the database, you can enter an SQL query that contains clauses, database field names, index values, and operators or you can enter the name of a public named query. By default, the arsdoc command sends the list of items that match the SQL query to stdout (UNIX) or the console (Windows NT). You typically redirect the output to a printer or another process. You can also choose to write the output to a file.

update

Use the update command function to update documents. You identify the application group to update and specify one or more application group fields and their values. To identify the documents you want to update, you must enter an SQL query that contains clauses, database field names, index values, and operators or a public named query. The update command function updates all rows that match the query.

Parameters

-a For the GET command function, when retrieving AFP documents, specify this parameter to include resources with the documents that are retrieved. If documents from the same application have different resource groups, separate output files are created for each resource group.

For the ADD command function, you must specify the name of the OnDemand application The application must belong to the application group named with the **-g** flag.

-c For the GET command function, use to concatenate all of the items that match the query in one output file. Name the output file with the **-o** flag. However, even if you do specify the **-c** flag, the arsdoc command creates separate output files when any of the following conditions occur:

- If more than one application group is referenced by the folder. The arsdoc command creates one output file for each application group that contains items that match the query.
- If more than one application is contained in an application group. The arsdoc command creates one output file for each application that contains items that match the query.

- If documents from the same application have different resource groups, the arsdoc command creates separate output files for each resource group.

For example, if a folder references two application groups, the following specification:

```
-o student -c
```

Can result in file names such as:

```
student.516
student.517
```

Where 516 and 517 are application group identifiers. One file is created for each application group. Each file contains all of the items that match the query for that particular application group.

-d directory
The name of the directory where arsdoc writes the output files. The directory must exist before the arsdoc command attempts to write the output files. In UNIX, directory names are case sensitive; in Windows NT, they are not.

-D For the QUERY command function, appends the document handle information to the end of each output record. The document handle information consists of the following ten values, in the order listed:

Document name
Offset
Length
Compressed object offset
Compressed object length
Annotation type
Compression type
Resource ID
Primary node ID
Secondary node ID

The values are separated by a delimiter. The default delimiter is the comma character. You can specify a different delimiter with the **-e** parameter.

You cannot specify the **-H**, **-n**, or **-N** parameters when you specify the **-D** parameter.

-e delimiter
For the QUERY command function, determines the character that

separates the database field values in the output. By default, OnDemand separates values in the output with a comma.

-f folder

The name of the OnDemand folder. The folder name must be specified exactly as it appears in OnDemand. The case of the folder name is significant. For example, to query the OnDemand system log folder, you must enter:

```
-f "System Log"
```

If you are using a parameter file, you must specify the **-f** flag in the parameter file. If you are not using a parameter file and you do not specify the **-f** flag, the arsdoc command prompts you for the folder name when you run the command.

Note: A folder can reference one or more application groups. If the folder references more than one application group, the arsdoc command searches each of the application groups. Because the arsdoc command generates a single SQL query to search the application groups, the attributes of the database fields must be the same for each application group. Attributes include the field name, type, and length. For example, suppose you define the following application groups and fields:

Application Group	Field Names
Student Bills	name, account, billDate
Student Grades	name, account, gradeDate
Student Transcripts	name, account, transcriptDate

You cannot query the application groups using the arsdoc command because the name of the date field is not the same for each application group. However, if you were to define the application groups and fields as follows:

Application Group	Field Names
Student Bills	name, account, studentDate
Student Grades	name, account, studentDate
Student Transcripts	name, account, studentDate

You can query the application groups using the arsdoc command because the names of the database fields are the same for each application group.

-F parmfile

Determines the name of the file that contains the actions to run and

other document and file parameters, values, and options. You typically specify this option when you want to perform multiple actions.

Flags and values specified in the parameter file must be enclosed in brackets.

You can use the \ (backslash) character to continue the parameters of an action to two or more lines. An action (one or more input lines) can contain a maximum of 32767 characters (bytes).

A parameter file can contain blank lines and comment lines. A comment line contains the # character in the first column.

-g For the GET command function, specify this parameter to generate generic index data for the items that match the query. Refer to the *Indexing Reference* for details about the format of the generic index file.

When you specify the **-g** flag, you must specify the **-c**, **-N**, and **-o** flags. However, you cannot specify database field names with the **-o** flag.

The arsdoc command uses the following convention to name output files generated with the **-g** flag:

-o.res_id.appl_group.appl.type

Where:

-o is the value specified with the -o flag.

res_id is the resource group identifier. A resource group identifier always starts at 1 (one) and is incremented by one for each version of the resource group.

appl_group is the name of the application group.

appl is the name of the application.

type is the file type:

- **out** identifies a document file.
- **ind** identifies a generic index file.
- **res** identifies a resource file.

In general, the number of files generated is dependent on the number of application groups in a folder, the number of applications in an application group, and the number of versions of resource groups in an application.

For the ADD and UPDATE command functions, specify the name of the OnDemand application group. The application group that you specify must be referenced by the folder named with the **-f** flag.

-h host

The host name or IP address of the OnDemand library server. This is a required parameter.

-H For the QUERY command function, specify this parameter to generate a header record in the output. The default header record contains the application group field names. By default, the field names are delimited with the comma character. You can specify a delimiter of your choice with the **-e** flag.

You can use the **-H** flag to generate output that contains only the application group database field names. To do so, specify the **-H** flag without specifying the **-i** or **-q** flags. (You also must not specify the **-L**, **-n** or **-N** flags.) The arsdoc command writes the database field names to the specified output file, or to stdout (UNIX servers) or the console (Windows NT servers), in the format used for the header record.

-i sql_query

A valid SQL query, that includes the names of one or more application group database fields, index values, and operators. OnDemand does not validate the string that you enter. Please refer to the SQL reference for your database manager product for an overview of SQL concepts and details about how to construct a query.

If you specify the **-i** flag, you must not specify the **-q** flag

For the DELETE or UPDATE command functions, if the SQL string results in multiple hits, multiple documents will be deleted or updated. For an update, all of the documents will be updated with the same values.

To construct a query with a database field of type date, you must use the OnDemand internal format of the date. That is, the number of days since January 1, 1970. You can use the arsdate command to display the internal format for a given date. The following shows how to use the arsdate command on an AIX server to obtain the internal date for July 21, 1995:

```
/usr/lpp/ars/bin/arsdate -a 7/21/95
```

The arsdate command displays:

```
7/21/95 -> 9333
```

Enter 9333 as the index value for the date database field.

-I For the QUERY command function, appends the Load ID to the end of each output record. The Load ID is separated from the database field values by a delimiter. The default delimiter is the comma

character. You can specify a different delimiter with the **-e** parameter. You cannot specify the **-H**, **-n**, or **-N** parameters when you specify the **-I** parameter.

To use the **-I** parameter, the user running the query must have permission to access the System Log application group and folder.

If the Load ID is not found in the system log, the string `Load ID could not be found` is appended to the end of the output record.

Note: The Load ID for a document is determined by searching the system log. Searching the system log can be very time consuming, depending on the number of records stored in the system log. The system log is searched for each document that matches the query.

-L max#

For the GET and PRINT command functions, determines the maximum number of items retrieved from OnDemand, regardless of the number of items that match the query. OnDemand retrieves items in the order that they were loaded in the database.

For the QUERY command function, determines the number of items included in the hit list, regardless of the number of items that match the query. OnDemand lists the items in the order that they were loaded into the database.

-n For the GET command function, specify this parameter to retrieve items one at a time from the database. By default, the arsdoc command uses bulk retrieval query processing for high-speed retrieval of items from the database.

For the QUERY command function, specifies that the items in the output file contain line numbers. If you specify this option, OnDemand sequentially numbers each line in the output file, beginning with 1 (one).

For the ADD and UPDATE command functions, specify application group database field names and their values using the form **-n***dbfield*=*value*. You can specify one or more field names and their values (by specifying the **-n** parameter one or more times). When adding a document, you must specify all of the application group fields unless you specify the **-O** flag. When updating a document, you can specify one or more fields and their values. For a date field, the value must be specified using the folder Display Format.

-N For the GET command function, when querying a folder that references more than one application group or a folder that references an application group that contains more than one application, specify this parameter to add the resource identifier, application group name,

and application name to the output file name. When you specify the **-N** flag, you must specify the **-c** flag. If you specify the **-g** flag to generate generic index data, you must specify the **-N** flag.

If the folder references more than one application group or an application group contains more than one application and you do not specify the **-N** flag, the arsdoc command adds the application group or application identifier to the output file name. For example, the following specification:

```
-o student -c
```

Can result in output file names such as:

```
student.516
student.517
```

Where 516 and 517 are application group identifiers. However, when you specify the **-N** flag, the arsdoc command uses the resource identifier, application group name, and application name to name the output file. For example, the following specification:

```
-o student -c -N
```

Can result in output file names such as:

```
student.1.BILLS.1995
student.1.BILLS.1996
```

Where 1 is the resource identifier, BILLS is the application group name, and 1995 and 1996 are application names.

The number of index files created is dependent on the number of application groups in a folder, the number of applications in an application group, and the number of resource groups in an application.

For the QUERY command function, determines the application group fields that the arsdoc command writes to the output file and the field names that appear in the header record. By default, the arsdoc command writes all fields to the output file. You can specify one or more application group field names using the form **-N(***dbfield***)...(***dbfield***)**. Each field name that you specify must be delimited with parenthesis. When you run a query from the command line, you must delimit the entire string in double quote characters. For example, **-N"(***dbfield1***)(***dbfield2***)"**.

-o name

For the GET command function, causes the arsdoc command to write

documents to one or more files and identifies the user-defined string used to generate unique file names. For example, the following specification:

```
-o student -c
```

Can result in the following output file name:

```
student
```

You can concatenate one or more of the database field names that you specify with the **-i** flag to generate a unique file name. For example, the following specification:

```
-o (sdate)(student)
-i "WHERE sdate='971025' AND student='001200340056'"
```

Can result in the file name:

```
971025.001200340056
```

When you use database field names to generate a unique file name:

- OnDemand verifies that the field names that you specify are valid for the application group(s) referenced by the folder specified with the **-f** flag.
- If the field name that you specify is a date field, the output format of the date is determined by the Format field setting on the application Load Information page.
- The field names must be delimited with parenthesis.
- You can specify the fields in any order. The order that you specify determines the file name that arsdoc generates.
- You cannot use a field name to represent a directory name. For example:

  ```
  -o (field_1)/(field_2)
  ```

  Is not valid.
- You cannot specify the **-c** flag to concatenate items in one output file. Each item that matches the query is stored in a separate output file.

If more than one item matches a query and you do not generate a unique file name using database field names, concatenate items in a single file with the **-c** flag, or specify the **-g** flag, the arsdoc command generates a unique file name for each item that matches the query by adding a *.n* extension to the file name. Where *n* is the number of the item that matched the query. For example, if you specify:

```
-o statements
```

And two items match the query, the arsdoc command creates the following files:

```
statements.1
statements.2
```

You must specify the **-o** flag when you specify the **-c** flag.

For the QUERY command function, determines the file name of the output file where arsdoc writes the list of items that match the query.

For the ADD command function, determines the name of the input file that contains the document to be added. The value that you specify is not checked for valid characters. You can specify a full path name, including the back slash and forward slash characters as part of a directory path. When adding a document, data can be provided by specifying the name of the input file that contains the data with the **-o** flag, an SQL query with the **-i** flag, or a public named query with the **-q** flag. Only one document can be added at a time.

-O For the ADD command function, you must specify this parameter if you intend to omit one or more database fields. However, you can never omit date or time fields. When you specify the **-O** flag, the arsdoc command stores a default value in any other database field you omit. The default value for string fields is an empty string. The default value for numeric fields is 0 (zero). Numeric fields include integer and decimal fields.

-p password
The password of the OnDemand user that you named with the **-u** flag. If there is no password assigned to the user that you specify, use double quotes to show a null password. That is, specify **-p** ″″. If you omit the **-p** flag, the arsdoc command prompts you to enter the password when you run the command. If there is no password assigned to the user that you specify, press the Enter key when prompted.

-P printer
For the PRINT command function, identifies the OnDemand server printer where you want to send the items that match the query.

-q named_query
The name of a public named query for the folder named with the **-f** flag. A named query is a set of search criteria previously saved on the library server that can be recalled by name to search a folder. A named query is typically defined to search a folder for a specific document or set of documents.

If you specify the **-q** flag, you cannot specify the **-i** flag or the **-S** flag.

-s seconds

For the GET command function, determines the number of seconds that the arsdoc command waits between query requests when you specify more than one query with the **-F** flag. If you do not specify this option, OnDemand does not wait between query requests. That is, the default is 0 (zero) seconds.

-S startdate,enddate

Provides a date range that the arsdoc command uses to limit a search to specific tables. When you specify this flag, the arsdoc command searches only tables that contain a segment within the specified date range. The search string is provided by the **-i** flag. Date values must be specified using the folder Display Format.

Note: For most queries, you should always specify this flag and a date range. Doing so limits the range of a query and can significantly improve the performance of a query. For the ADD command function, if you specify the **-o** flag, you cannot specify the **-S** flag. For all command functions, if you specify the **-q** flag, you cannot specify the **-S** flag.

-u userid

The OnDemand user permitted to perform the function. The arsdoc command verifies that the userid that you specify is a valid OnDemand user for the server that you name with the **-h** flag, that the userid is permitted to open the folder that you name with the **-f** flag, and that the userid has application group permission to perform the specified function. If you omit the **-u** flag, the arsdoc command prompts you for the userid when you run the command.

-v Enables verbose mode, to displays all messages (informational and error). By default, the arsdoc command displays error messages.

Examples

1. The following shows how to invoke the arsdoc command using the command line parameters to retrieve an item from OnDemand and save a copy of the item in a file in the current directory.

```
arsdoc get -h rockies -f "Student Information" -o student
 -c -S 1/1/97,12/31/97 -i "WHERE student='001200340056'" -v
```

The arsdoc command concatenates all items that match the query in the following output file:

```
student
```

2. The following shows how to invoke the arsdoc command and specify the name of a parameter file in the current directory that contains a set of queries. The results of each query are saved in separate files in the current directory.

```
arsdoc get -h rockies -F parmfile -v
```

   The parameter file contains two queries:

```
[-f "Student Information"] [-S 1/1/97,12/31/97] [-o (student)(type)] \
[-i "WHERE student='123420010056' AND type='B' OR type='G' OR type='T'"]

[-f "Student Information"] [-S 1/1/97,12/31/97] [-o (student)(type)] \
[-i "WHERE student='123450011917' AND type='B' OR type='G' OR type='T'"]
```

   Assuming items exist in the database for all types, the arsdoc command creates the following output files:

```
123420010056.Bills
123420010056.Grades
123420010056.Transcripts
123450011917.Bills
123450011917.Grades
123450011917.Transcripts
```

3. The following shows how to invoke the arsdoc command using the command line parameters to retrieve items from OnDemand, write the documents to a file, and generate and save generic index data for the items that match the query. The name of a public named query is identified.

```
arsdoc get -h rockies -f "Student Information"
 -a -c -g -o student -q "3rd yr students GPA>3.5" -N -v
```

   The number of output files generated is a factor of the number of application groups queried, the applications contained in the application groups, whether the data is AFP, and if so, the versions of resource groups in each application. At a minimum, for AFP data with one version of the resource group and one application group, the arsdoc command generates three output files. For example:

```
student.1.Student Information.TRANSCRIPTS.ind
student.1.Student Information.TRANSCRIPTS.out
student.1.Student Information.TRANSCRIPTS.res
```

   For AFP data, if there is more than one version of the resource group per application, the arsdoc command can generate additional output files. For example:

```
student.1.Student Information.TRANSCRIPTS.ind
student.1.Student Information.TRANSCRIPTS.out
student.1.Student Information.TRANSCRIPTS.res
student.2.Student Information.TRANSCRIPTS.out
student.2.Student Information.TRANSCRIPTS.ind
student.2.Student Information.TRANSCRIPTS.res
```

If the application group contains more than one application, the arsdoc command can generate additional output files. For example:

```
student.1.Student Information.BILLS.ind
student.1.Student Information.BILLS.out
student.1.Student Information.BILLS.res
student.1.Student Information.GRADES.ind
student.1.Student Information.GRADES.out
student.1.Student Information.GRADES.res
```

4. The following shows how to invoke the arsdoc command using the command line parameters to generate a list of items and save the list in a file in the current directory.

```
arsdoc query -h rockies -f "Student Information"
 -o query1.out -H -S 1/1/97,12/31/97
 -i "WHERE student='0012-0034-0056' AND
     type='B' OR type='G' OR type='T'" -v
```

5. The following shows how to invoke the arsdoc command and specify the name of a parameter file in the current directory that contains a set of queries. The result of each query is saved in a separate file in the current directory.

```
arsdoc query -h rockies -F parmfile -v
```

   The parameter file contains three queries:

```
[-f "Student Information"] [-i "WHERE type='B'"] [-o "query2.out"] \
[-S 1/1/97,12/31/97] [-H] [-N (student)(id)(p_date)]

[-f "Student Information"] [-i "WHERE type='G'"] [-o "query3.out"] \
[-S 1/1/97,12/31/97] [-H] [-N (student)(id)(p_date)]

[-f "Student Information"] [-i "WHERE type='T'"] [-o "query4.out"] \
[-S 1/1/97,12/31/97] [-H] [-N (student)(id)(p_date)]
```

6. To add a document to the Credit Card Statements folder:

```
arsdoc add -h rockies -o/newdata/crd.dat
  -n "crd_date=01/21/98" -n "account='000-000-000'"
  -n balance=123.45 -n "name='John Watpole'"
  -f "Credit Card Statements" -g CRD -a CRD -v
```

7. To update a document in the Credit Card Statements folder, changing the balance from 123.45 to 0.00:

```
arsdoc update -h rockies
  -i "where account='000-000-000' and name='John Watpole'"
  -n "balance=0.00" -S 1/1/97,12/31/97
  -f "Credit Card Statements" -g CRD -v
```

8. To delete a document from the Credit Card Statements folder:

```
arsdoc delete -h rockies
  -i "where account='000-000-000' and name='John Watpole'"
  -f "Credit Card Statements" -S 1/1/97,12/31/97 -v
```

9. To send items that match a query to the OnDemand server print facility:

```
arsdoc print -h rockies -P svrprt1
  -i "where account='000-000-000' and name='John Watpole'"
  -f "Credit Card Statements" -S 1/1/97,12/31/97 -v
```

10. You can use the arsdoc command to add an index that points to an existing document. For example, assume you previously loaded a bank statement for account number 000-000-000, date 5/23/97, and account name Joe Smith. You now want to add a new index, but point to the existing statement. The new index uses the same account number and date, but contains a different account name (for example, Sally Smith). After adding the index, if a query is run with account name Joe Smith or Sally Smith, the same bank statement will be retrieved. To add an index for an existing document:

    ```
    arsdoc add -h rockies
      -i "where sdate=10005 and account='000-000-000' and name='Joe Smith'"
      -n "sdate=5/23/97" -n "account=000-000-000" -n "name='Sally Smith'"
      -f "Credit Card Statements" -S 1/1/97,6/31/97 -v
    ```

Notes

The arsdoc add command will fail unless the application group Database Organization is Multiple Loads per Database Table and Expiration Type is Segment or Document.

Files

/usr/lpp/ars/bin/arsdoc
: The AIX executable program.

/opt/ondemand/bin/arsdoc
: The HP-UX and Solaris executable program.

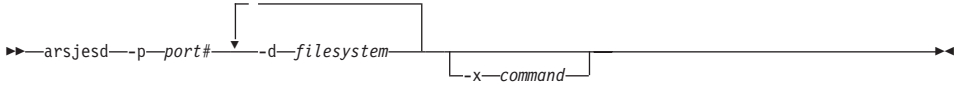
\Program Files\IBM\OnDemand for WinNT\bin\arsdoc
: The Windows NT executable program.

Chapter 27. arsjesd command reference

Purpose

The arsjesd command monitors a TCP/IP port for datasets transmitted by Download on an MVS or OS/390 system and stores data in file systems on the server.

Syntax



Description

The arsjesd command is the server component of Download. An optional feature of PSF/MVS (or PSF for OS/390), Download can be used to automatically transmit application program output datasets at high speed from the JES spool to file systems on OnDemand servers. Download also transmits values you specify in JCL parameters and can optionally transmit additional job information. You can use these values to specify the name of the application and application group where OnDemand loads the transmitted data. Download is made up of a pair of cooperating components, a Download writer that transmits the dataset and a program that receives and stores the data on the server. See *PSF for MVS: MVS Download Guide* and *PSF for OS/390: Download for OS/390* for details about Download, including how to configure and operate Download on MVS and OS/390 systems, a list of the JCL parameters transmitted with a file, and information about the user exit program you can use to provide additional job information to the arsjesd command.

The arsjesd command monitors a TCP/IP port for activity from an MVS or OS/390 system. Multiple instances of the arsjesd command can be started, each monitoring a specific port for data transmitted from a different Download writer. A Download writer and its corresponding arsjesd instance must specify the same port number. If you start multiple instances of the arsjesd command, make sure that you specify different file systems for each instance. Overlapping file systems can adversely affect performance.

The arsjesd command stores transmitted datasets in file systems on the server. You can specify up to ten file systems that the arsjesd command can use to store transmitted data. The arsjesd command always stores data beginning with the first file system specified. If there is not enough space available to store the dataset being transmitted on the first file system specified, the arsjesd command switches to the second file system specified. If there is not enough space available to store the dataset being transmitted on the last file system specified, the arsjesd command switches to the first file system specified.

Parameters

-p port#

The TCP/IP port number that the arsjesd command monitors for data from the MVS or OS/390 system. The valid range of port numbers is 5001 through 64000. However, avoid using port numbers 5001, 8251, and 8253. The Download writer and the arsjesd command must specify the same port number. See *PSF for MVS: MVS Download Guide* and *PSF for OS/390: Download for OS/390* for information about configuring Download on MVS and OS/390 systems.

-d filesystem

The name of the file system where the arsjesd command stores transmitted data. You can specify up to 10 file system parameters, separated by intervening blanks. The *Introduction and Planning Guide* shows a recommended naming convention for file systems that support Download.

-x command

The name of a user-defined program that the arsjesd command invokes to process the downloaded file, JCL parameters, and other job information after receiving and storing the file on the server. You can specify the full path name of the program. If you do not specify the full path name, the operating system searches the current directory for the program. If the file is not found in the current directory, the operating system continues to search for the file using the directories defined in the PATH environment variable.

You typically specify this parameter when you need to process the transmitted data before it can be loaded into OnDemand. See “Chapter 6. Download exits and processing” on page 43 for more information.

Examples

1. The following shows how to start the arsjesd command to monitor TCP/IP port number 6001 and place transmitted data in the /arsacif/acif1 file system:

   ```
   arsjesd -p 6001 -d /arsacif/acif1
   ```

2. The following shows how to start the arsjesd command to monitor TCP/IP port number 6002 and place transmitted data in the /arsacif/acif1 file system. If there is not enough space available in the /arsacif/acif1 file system to store the dataset being transmitted, the arsjesd command stores the dataset in the /arsacif/acif2 file system.

   ```
   arsjesd -p 6002 -d /arsacif/acif1 -d /arsacif/acif2
   ```

3. The following shows how to start the arsjesd command to monitor TCP/IP port number 6002 and place transmitted data in the /arsacif/acif1 file system. After transmitting the file to the server, the arsjesd command invokes the /usr/mvs_download user-defined program. A user-defined program can perform any required functions, such as determining the name of the application group and application to load, renaming or copying input files, and invoking other programs.

   Note: You must supply your own user-defined program.

   ```
   arsjesd -p 6002 -d /arsacif/acif1 -x /usr/mvs_download
   ```

Notes

A file transmitted by Download uses the following file naming convention:

```
MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD
```

- By default, the arsload command uses the FORMS part of the filename to identify the application group to load. When you run the arsload command, you can use the **-G** flag to specify a different part of the filename to identify the application group. For example, arsload -G JOBNAME.
- If the application group to load contains more than one application (source of data), you must identify the application to load. Otherwise, the load will fail. When you run the arsload command, you can use the -A parameter to specify the part of the filename that identifies the application. For example, to use the DATASET part of the filename to identify the application, run the arsload command with the -A DATASET parameter.

Choose from the MVS, JOBNAME, DATASET, and FORMS parts of the filename to identify the application group and application to load.

Note: Depending on the date format you specify to Download, the date part of the file name can be YYYYDDD or YYDDD. Apply APAR OW36539 to Download and use the DATE_FORMAT keyword in the routing file to

control the format of the date placed in the file name. See *PSF for MVS: MVS Download Guide* or *PSF for OS/390: Download for OS/390* for details.

Files

/usr/lpp/ars/bin/arsjesd
The AIX executable program.

/opt/ondemand/bin/arsjesd
The HP-UX and Solaris executable program.

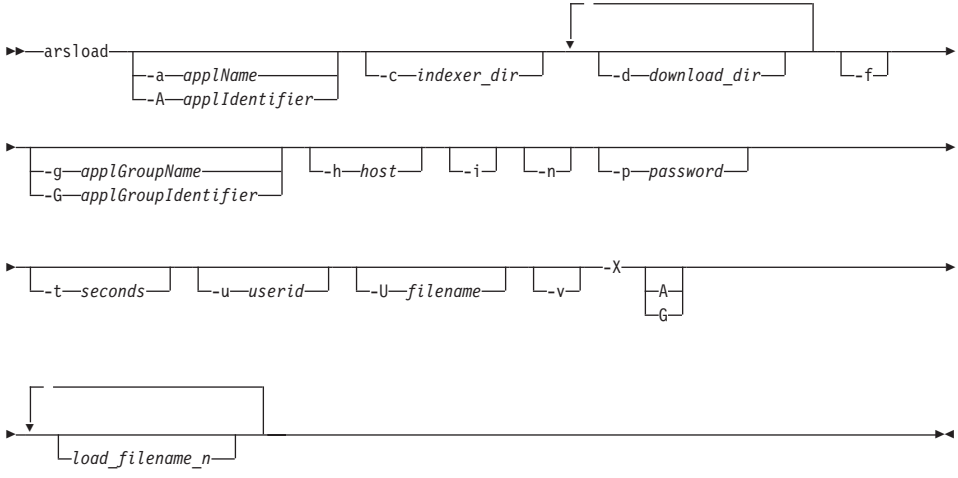
\Program Files\IBM\OnDemand for WinNT\bin\arsjesd
The Windows NT executable program.

Chapter 28. arsload command reference

Purpose

The arsload command can be used to index reports and load data into an application group. The arsload command determines if the input data needs to be indexed, and if so, calls the appropriate indexing program. The arsload command then calls the programs that load report data on storage volumes and update the OnDemand database with the index information extracted from or supplied with the report.

Syntax



Description

The arsload command is the main OnDemand data indexing and loading program. The arsload command calls the data conversion and indexing program if the data needs to be indexed, creates input files for OnDemand, and calls programs that load data on storage volumes and update the OnDemand database with index data. The arsload command writes processing messages to the system logging facility. You can open the System Log folder and review the messages generated during a report file load.

You typically set up the arsload command to run as a daemon (UNIX servers) or service (Windows NT servers), periodically checking specified file systems for input data downloaded from other systems and automatically processing the downloaded files.

When you run the arsload command, you must provide the userid and password of an OnDemand user with administrator authority for the application group being loaded. There are several ways you can provide the userid and password:

- Use the -u and -p parameters each time you run the arsload command
- Use the -U parameter each time you run the arsload command to name a file that contains the userid and password for a user with administrator authority for the application group
- On UNIX servers, set a default system administrator userid and password in the arsload.cfg file, which is used whenever you run the arsload command unless you specify otherwise (with the -u and -p parameters or if you specify a different userid/password file with the -U parameter)
- On Windows NT servers, the Load Data service uses the userid and password of the instance owner. If you run the arsload command from the prompt, you should use OnDemand unified login. See the -u and -U parameters for details.

Parameters

-a applName

The name of the application to load. If the application group contains more than one application, you must identify the application to load. Otherwise, the load will fail. You typically use this parameter when you run the arsload command from the prompt to process one or more input files to be loaded into the same application. If you plan to automate the data loading process of files for different application groups and applications, use the **-A** parameter to specify the part of the filename that identifies the name of the application.

-A applIdentifier

Determines the part of the filename used to identify the application to load. If the application group contains more than one application, you must identify the application to load. Otherwise, the load will fail. You typically use this parameter when running the arsload command as a daemon (UNIX) or service (Windows NT) to automate the loading of files for different application groups and applications. For example, you can use Download to transmit files from MVS and OS/390 systems to the server. A file transmitted by Download uses the following file naming convention:

```
MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD
```

- By default, the arsload command uses the FORMS part of the filename to identify the application group to load. You can use the **-G** flag to specify a different part of the filename that identifies the application group. For example, arsload -G JOBNAME.
- If the application group to load contains more than one application (source of data), you must identify the application to load. Otherwise, the load will fail. For example, to use the DATASET part of the filename to identify the application, run the arsload command with the -A DATASET parameter.

Choose one of the MVS, JOBNAME, DATASET, and FORMS parts of the filename to identify the application.

-c indexer_dir
The file system where OnDemand temporarily stores files created by the indexing program. The default is the directory where the arsload command was invoked.

For better performance, we recommend the **-c indexer_dir** point to a different file system than the **-d download_dir** parameter.

-d download_dir
The file system that contains report data transferred from other systems. For example, you can specify the name of the file system where the arsjesd command stores data transmitted by Download from an OS/390 system.

You can specify this parameter and value more than once, which causes the arsload command to use multiple download directories.

For better performance, we recommend the **-d download_dir** parameter point to a different file system than the **-c indexer_dir** parameter.

-f Unload the data if the report file load fails. If either the database manager or the storage manager step fails, OnDemand deletes any index data added to the database and storage objects placed on storage volumes during the report file load.

If the report file load fails, check the message log in the system logging facility. If the message log contains a Load ID, it means OnDemand stored some of the report file data in the application group. Before you restart the report file load, you must unload the data with the arsadmin command. “Chapter 22. arsadmin command reference” on page 153 provides details about using the arsadmin command.

-g applGroupName
The name of the application group to load. This parameter is required if you specify the **load_filename_n** parameter. This parameter is

optional if you specify the **-d download_dir** parameter. If you specify the **-d download** parameter, by default, the arsload command uses the FORMS part of the downloaded filename to determine the name of the application group to load. If you plan to automate the data loading process of files for different application groups and applications, use the **-G** parameter to specify the part of the filename that identifies the application group to load.

-G applGroupIdentifier

Determines the part of the file name used to identify the name of the application group to load. You typically use this parameter when running the arsload command as a daemon (UNIX) or service (Windows NT) to automate the loading of files for different application groups and applications. For example, you can use Download to transmit files from MVS and OS/390 systems to the server. A file transmitted by Download uses the following file naming convention:

```
HLQ.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD
```

- By default, the arsload command uses the FORMS part of the filename to identify the application group to load. You can use the **-G** flag to specify a different part of the filename that identifies the application group. For example, `arsload -G JOBNAME`.
- If the application group to load contains more than one application (source of data), you must identify the application to load. Otherwise, the load will fail. For example, to use the DATASET part of the filename to identify the application, run the arsload command with the `-A DATASET` parameter.

Choose one of the MVS, JOBNAME, DATASET, and FORMS parts of the filename to identify the application group.

-h host

The fully qualified host name or IP address of the OnDemand library server.

You must specify this flag and the library server name, when you run the arsload command on an OnDemand object server that resides on a different workstation than the library server.

-i Run the data indexing program only, do not load report file data on storage volumes or indexing information in the database.

-n Determines whether OnDemand deletes the input files when the arsload command ends.

- If you specify **-n**, OnDemand does not delete the input files when the arsload command ends.
- If you do not specify **-n**, OnDemand deletes the input files when the arsload command ends.

In either case, if the arsload command fails in the load step because of a device or system problem, you can restart the load step after correcting the problem, using the intermediate files created by the arsload command. The arsload command stores the intermediate files in the work directory named with the **-c** flag or the directory where you started the arsload command, if you did not specify the **-c** flag. The intermediate files have the same filename as the original input file.

-p password

The password for the user specified with the **-u** parameter. If the user is not assigned a password, enter a null password (that is, `-p ""`).

-t seconds

Determines the polling time in seconds. This is the interval that the arsload command checks the data download directories. The default is 600 seconds (ten minutes).

-u userid

The userid of an OnDemand user with administrator authority for the application group. The user must have permission to add documents to the application group.

If you omit the **-u** parameter, the arsload command attempts to obtain the userid and password from the file named with the **-U** parameter. If the arsload command cannot locate or use the file named with the **-U** parameter, it attempts to log on to OnDemand and access the application group with a userid of `admin` and no password. If you omit the **-U** parameter, the arsload command attempts to obtain the userid and password from the arsload.cfg file. If the arsload command cannot locate or use the arsload.cfg file, it attempts to log on to OnDemand and access the application group with a userid of `admin` and no password. The *Installation and Configuration Guide for UNIX Servers* describes how to configure the arsload.cfg file.

-U filename

If you omit the **-u** parameter, determines the file that contains the userid and password the arsload command uses to log on to OnDemand and access the application group. If the arsload command cannot locate or use the file named with the **-U** parameter, it attempts to log on to OnDemand and access the application group with a userid of `admin` and no password. If you omit the **-U** parameter (and the **-u** parameter), OnDemand attempts to obtain the userid and password from the arsload.cfg file. If the arsload command cannot locate or use the arsload.cfg file, it attempts to log on to OnDemand and access the application group with a userid of `admin` and no password. The *Installation and Configuration Guide for UNIX Servers* describes how to configure the arsload.cfg file.

Important: After you set the userid and password in the file named with the -U parameter (or the arsload.cfg file), remember to change the password any time you change the user's password in OnDemand. Otherwise the load will fail. The arsload command accepts an expired password. However, the arsload command fails if you specify an incorrect password.

Note: The arsload.cfg file and the -U parameter are not used on Windows NT servers. The OnDemand Load Data service logs on to OnDemand and accesses the application group with the userid and password of the instance owner (typically an OnDemand system administrator user). If you run the arsload command from the prompt, OnDemand unified login should be used to log on to OnDemand and access the application group. The *Installation and Configuration Guide for Windows NT Servers* provides information about configuring services and unified login.

-v Enables verbose mode, to display all messages (informational and error). By default, the arsload command displays error messages.

-X indexer

Allows you to override the default indexer defined in the application. The possible values are A and G. Specify **-X A** to use ACIF. Specify **-X G** to use the OnDemand Generic indexer.

You must override the default indexer if you create index data in a format other than that supported by the indexer defined in the application. For example, you define an application and specify ACIF as the indexer. Later, you use the arsdoc command to extract documents from an application group. The arsdoc command generates index data in the Generic index format. To load the index data into the application, you must specify:

```
arsload -X G . . .
```

load_filename_n

Determines the filename of one or more input files to process. The arsload command concatenates the following filetype extensions with the filename that you specify: IND, OUT, and RES:

- The IND file contains the index data
- The OUT file contains the report data
- The RES file contains the resource data

For example, if you specify `arsload -g BILLS po3510`, the arsload command looks for the following files:

- po3510.ind

- po3510.out
- po3510.res

If the arsload command does not find a file with the IND filetype extension, it automatically calls the arsacif command to index the input file.

Note: If you do not specify a filename, the arsload command checks the directory specified with the **-d** flag for input files to process. Any files with a filetype extension of ARD or PDF will be processed. ARD files are those transmitted to the server by Download. PDF files are those created by Acrobat Distiller. The case of the filetype extension is not significant.

Examples

1. The following shows how to run the arsload command to check the specified directory for data transmitted by Download. Input files must have a file type of ARD or PDF. The arsload command stores work files in the specified directory. The arsload command uses the FORMS part of the file name to determine the application group to load, and the application group contains only one application. The arsload command logs on to OnDemand and accesses the application group with the userid and password from the arsload.cfg file.

   ```
   arsload -c /arsacif/acif1 -d /arsacif/acif2
   ```

2. The following shows how to run the arsload command to check the specified directory for data transmitted by Download. Input files must have a file type of ARD or PDF. The arsload command stores work files in the specified directory. The arsload command uses the JOBNAME part of the file name to determine the application group to load and the DATASET part of the file name to determine the application to load. The arsload command logs on to OnDemand and accesses the application group with the userid and password from the arsload.cfg file.

   ```
   arsload -c /arsacif/acif1 -d /arsacif/acif2 -A DATASET -G JOBNAME
   ```

3. The following shows how to run the arsload command to store data in the specified application group. An input filename is specified. The arsload command logs on to OnDemand and accesses the application group with the userid and password provided with the -u and -p parameters.

   ```
   arsload -g BILLS -u bob -p secret PO3510
   ```

4. The following shows how to run the arsload command to store data in the specified application group. The application group name contains an embedded blank character, and must be quoted. Two input filenames are specified. The arsload command logs on to OnDemand and accesses the application group with the userid and password provided with the -u and -p parameters.

```
arsload -g "ABC Credit" -u bob -p secret RW7505 RW8505
```

5. The following shows how to run the arsload command to store data in the specified application group. The IP address on the OnDemand library server is specified. This format of the command can be used to load data on an object server and store index data on the library server. An input filename is specified. The arsload command logs on to OnDemand and accesses the application group with the userid and password from the file named with the -U parameter.

```
arsload -g "BILLS" -h "9.99.111.222" -U userpw.cfg ARSP0MST
```

Notes

We recommend that you do not run the arsload command at the same time that you run database statistics (for example, running the arsmaint command with the -r option).

We recommend that you do not run the arsload command at the same time that you run the arsdb command with the -m or -s options.

If you do not specify an input filename, the arsload command checks the directory specified with the **-d** flag for input files to process. Any file with a filetype extension of ARD or PDF will be processed. ARD files are those transmitted to the server by Download. PDF files are those created by Acrobat Distiller. The case of the filetype extension is not significant.

Files

/usr/lpp/ars/bin/arsload
The AIX executable program.

/opt/ondemand/bin/arsload
The HP-UX and Solaris executable program.

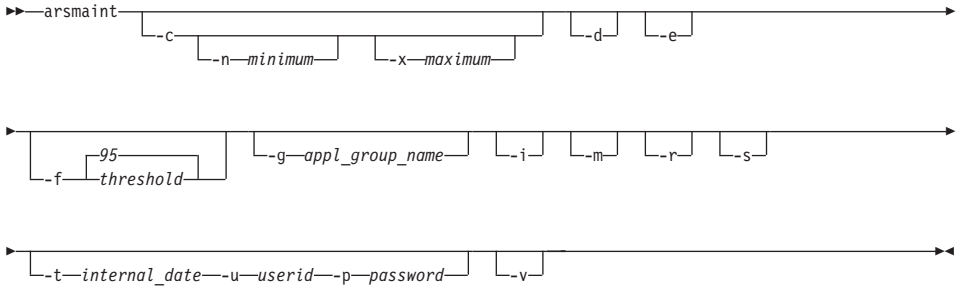
\Program Files\IBM\OnDemand for WinNT\bin\arsload
The Windows NT executable program.

Chapter 29. arsmaint command reference

Purpose

Maintain application group data in the OnDemand database and cache file systems. You typically run the arsmaint command on a regular schedule to migrate index data to archive media, delete index data from the database, migrate files from cache file systems to archive media, and delete files from cache storage.

Syntax



Description

The arsmaint command maintains application group data in the OnDemand database and cache storage based on the storage management values that you provide for application groups. The arsmaint command uses values that you specify for the life of data and indexes, length of time to cache data on magnetic, length of time before copying cache to archive media, length of time before migrating indexes to archive media, and length of time to maintain imported migrated indexes. The arsmaint command uses the expiration type to determine how to delete index data from an application group. The arsmaint command can expire a segment (table) of application group data at a time, a report file load at a time, or individual items.

The **-d**, **-e**, and **-i** flags cause the arsmaint command to migrate and expire index data from the database.

The **-c** and **-m** flags cause the arsmaint command to migrate and expire files from cache storage.

Note: The business and operational requirements of your organization and the configuration of your system determine when you can maintain data and the time available for maintenance, such as reorganizing data and backing up the database. Most database maintenance should take place when there is little or no other activity (especially loading data or administering the system). Some maintenance tasks must run when the system is idle.

You should carefully schedule maintenance processes to avoid problems, such as locking database tables. For example, we recommend that you do not run the arsload command at the same time that you run database statistics (for example, running the arsmaint command with the -r option or the arsdb command with the -s option) or perform maintenance on the database (for example, running arsdb with the -m option). There may be other processes that conflict; for example, you cannot load data or maintain the database while an offline backup of the database is in progress.

The time of day and frequency with which you run the maintenance programs and the processing options you specify will vary based on your environment. For example, if you do not load data every day or you add few rows to the database when you do load data, you can schedule database maintenance less often, perhaps once a month. Conversely, if you load a high volume of data every day, we recommend that you maintain the database after the load processing completes. In addition, if your system has plenty of space available to store index data and reports, you may be able to maintain the system less often.

The examples in this chapter demonstrate how to set up the database maintenance processes using typical options: the use of cache storage, migrating data to archive media, and maintaining the database and cache storage every day. Use these examples as a guideline. The exact syntax of the commands and when you can run them on your system will vary.

Parameters

-c Expire files from cache storage. We recommend that you schedule the arsmaint command to run every day using this parameter.

-d Expire indexes from the OnDemand database. We recommend that you schedule the arsmaint command to run every day using this parameter.

-e Migrate index data to archive media. We recommend that you schedule the arsmaint command to run every day using this parameter.

Migration is the process of moving tables of index data from the database to archive media. This process optimizes database storage space while allowing you to maintain index data for a very long time. You typically migrate index data after users no longer need to access the reports, but for legal or other requirements, you still need to maintain the data for some number of years or months.

You must configure your application groups for migration. OnDemand uses the length of time before migrating indexes value to determine when to migrate index data. After maintaining the imported index data for the number of days specified in the length of time to keep imported migrated indexes field, OnDemand expires the data from the database. Refer to the administrator interface online help for details.

Before OnDemand can migrate index data to archive media, you must assign the System Migration application group to a storage set that writes to archive media. Refer to the *Installation and Configuration Guide* for details.

-f threshold

Send an alert message when a cache file system is filled to the percentage specified. The default value is 95. The range is 0 (zero) to 100.

When a cache file system exceeds the threshold, it usually means that the arsmaint command cannot delete enough items from the file system to drop below the threshold. You may need to adjust the data migration and expiration values of the application groups using the file system. Otherwise, increase the size of the file system or add a new cache file system.

The arsmaint command sends the alert message to the console, the system log, and the system log user exit.

-g appl_group_name

The name of the application group to process. By default, the arsmaint command maintains the database or cache storage for all of the application groups defined to OnDemand. Specify this parameter and the name of an application group when you want to maintain the database or cache storage for a specific application group. The **-g** parameter is valid only with the **-c**, **-d**, **-e**, **-i**, **-m**, and **-r** parameters.

-i Expire imported index data from the database. We recommend that you schedule the arsmaint command to run every day using this parameter.

An administrator can import index data previously migrated to archive media back into the database to satisfy a query. After maintaining the imported index data for the number of days specified in the length of time to keep imported migrated indexes field, OnDemand expires the data from the database.

-m Migrate files from cache storage to archive media. We recommend that you schedule the arsmaint command to run every day using this parameter.

-n minimum
The low expiration threshold percentage for each cache file system. This value determines when the arsmaint command stops expiring files from cache file systems (see the **-c** and **-x** flags). The range is 0 (zero) to 100.

The default value is 80 (percent), which keeps a cache file system fairly full to provide good retrieval hit rates, while leaving free space for report file loads. However, we recommend that you periodically check the amount of available space in your cache file systems, especially before scheduling very large report file loads. You may need to increase the size of one or more cache file systems, add a new cache file system, or manipulate the data migration and cache management values to provide sufficient free space for large report file loads.

-p password
The password for the OnDemand userid identified with the **-u** flag. If the user is not assigned a password, enter a null password (that is, `-p ""`). If you do not specify this parameter, the arsmaint command prompts you to enter the password when you run the command. If the user is not assigned a password, press the Enter key when prompted.

-r Run database statistics. Causes the database manager to optimize index data. Makes access to information as efficient as possible. We recommend that you schedule the arsmaint command to run every day using this parameter.

-s Print a report that contains status information about the cache file systems. The report includes information such as the name of the file systems, the amount of space allocated to each file system, and the amount of free space for each file system. The arsmaint command places the report in the system log. We recommend that you schedule the arsmaint command to run every day using this parameter.

-t internal_date
Specify this parameter when you want to use the arsmaint command to process the database and cache storage using a date other than the

current system date (the default). For example, you can specify a future date to cause the arsmaint command to process data and files before the actual expiration/migration period. The value that you specify must be a valid OnDemand internal date value. You can use the *arsdate* command to display the internal date value for a given calendar date. When you specify this option, you must specify the **-u** and **-p** flags. The arsmaint command processes data and files belonging to the application groups, where the user (**-u**) has the delete document permission enabled.

Note: Use this option with caution. We recommend that you contact the IBM support center before running the arsmaint command with this option.

-u userid

An OnDemand user with administrator permission, or if you specify the **-g** flag, permission to delete documents from the application group. If you do not specify this parameter, the arsmaint command prompts you to enter the userid when you run the command.

-x maximum

The high expiration threshold percentage for each cache file system. This value determines when the arsmaint command begins expiring files from cache file systems (see the **-c** flag). When a cache file system reaches this threshold, the arsmaint command searches the file system and expires every file that meets the data migration and caching values. The arsmaint command stops expiring files when the amount of used space in a cache file system drops below the low expiration threshold (see the **-n** flag). The range is 0 (zero) to 100.

The default value is 80 (percent), which keeps a cache file system fairly full to provide good retrieval hit rates, while leaving free space for report file loads. However, we recommend that you periodically check the amount of space available in your cache file systems, especially before scheduling very large report file loads. You may need to increase the size of one or more cache file systems, add a new cache file system, or manipulate data migration and cache management values, to provide sufficient free space for large report file loads.

-v Validates cache storage. When you specify this option, the arsmaint command inspects all cache file systems to ensure that they are correctly linked with the proper file permissions.

Examples

1. The following shows how to run the arsmaint command with options to migrate and expire files from cache storage, migrate and expire indexes from the database, optimize database index data, print statistics, and validate cache storage. These options are typical of those used to maintain a library server and object server with archive media on the same workstation.

   ```
   arsmaint -cdeimrsv
   ```

2. The following shows how to run the arsmaint command with options to migrate and expire files from cache storage, print statistics, and validate cache storage. These options are typical of those used to maintain archive media on an object server that resides on a different workstation or node than the library server.

   ```
   arsmaint -cmsv
   ```

Notes

Please refer to the administrator interface online help for details about defining storage management information for application groups.

OnDemand server programs can remain active while the arsmaint command processes. However, we recommend that you schedule the arsmaint command at a time of little or no OnDemand client program activity.

We recommend that you do not run the arsload command at the same time that you run the arsmaint command with the -r option.

We recommend that you create a backup image of the OnDemand database and the ADSM database before you run the arsmaint command.

Files

/usr/lpp/ars/bin/arsmaint
The AIX executable program.

/opt/ondemand/bin/arsmaint
The HP-UX and Solaris executable program.

\Program Files\IBM\OnDemand for WinNT\bin\arsmaint
The Windows NT executable program.

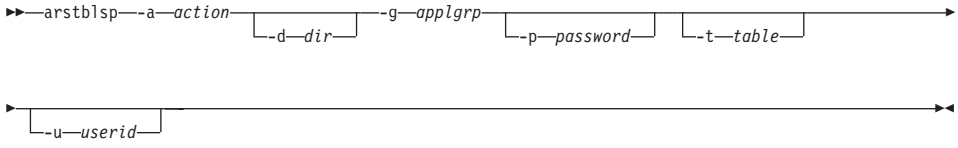
Chapter 30. arstblsp command reference

Purpose

Migrate tables of application group data from the default table space to their own table spaces.

Note: The arstblsp command is available only for DB2.

Syntax



Description

The arstblsp command can be used to migrate existing tables of application group data from the default table space to their own table spaces. We recommend that all existing customers (OnDemand version 2.1.1.1 or earlier) use this command to migrate application group data to table spaces as soon as possible.

Before you can migrate application group data to table spaces, all of the tables in the application group must be closed. Use the arstblsp command with the **-a1** option to close a table.

If you plan to use ADSM to manage DB2 log files and backup images, you must define the ADSM storage hierarchy before you migrate application group data to table spaces. Refer to the *Installation and Configuration Guide* for details.

Before you begin to migrate application group data to table spaces, we recommend that you create a full backup image of the database.

When you migrate application group data to table spaces, we strongly recommend that you do so during off-peak hours when little or no other OnDemand activity takes place.

Parameters

-a action

The action to perform. The action can be one of the following values:

0 Migrate one or more tables of application group data to their own table spaces. Specify the application group to migrate with the -g flag. Optionally specify a table to migrate with the -t flag. A table must be closed before it can be migrated to a table space.

1 Close a table that is still open for loading. This action causes OnDemand to close the table that is currently open for loading in the specified application group. The next time data is loaded into the application group, the data is loaded into a table space. Optionally specify a table to close with the -t flag.

2 List tables of application group data that have not been migrated to their own table spaces.

3 List the tables of application group data that are open for loading. An open table must be closed before it can be migrated to a table space.

-d dir Forces the arstblsp command to export the existing application group data to disk. If you specify this parameter, you must identify the name of a directory or filesystem with sufficient space to hold a copy of the existing application group table. Depending on the size of the table you are migrating, this can be a significant amount of disk space. For example, a table that contains 2.5 million rows, where each row is 100 bytes, may require up to 500 MB of free disk space to complete the migration.

Note: By default, on UNIX servers, OnDemand uses named pipes to export the existing application data, which provides the best performance with no filesystem overhead. We strongly encourage you to use named pipes. On Windows NT servers, OnDemand uses a temporary directory to export data. You can set a default temporary directory with the OnDemand configurator.

-g applgrp

The name of the application group to process.

-p password

The password for the userid specified with the -u flag. If the user is not assigned a password, enter a null password (that is, `-p ""`). If you do not specify this flag, the arstblsp command will prompt you to enter the password when you run the command. If the user is not assigned a password, press the Enter key when prompted.

-t table

With the -a1 option, identifies the table to close. With the -a0 option, optionally identifies the table of application group data to migrate. If you do not specify this flag with the -a0 option, the arstblsp command migrates all tables for the specified application group to their own table spaces.

-u userid

The userid of a user with administrator authority for the application group. If you do not specify this flag, the arstblsp command will prompt you to enter the userid when you run the command.

Examples

1. The following shows how to list the tables of application group data that have not been migrated:

   ```
   arstblsp -a 2 -g credit
   ```

2. The following shows how to list the tables of application group data that are open for loading:

   ```
   arstblsp -a 3 -g credit
   ```

3. The following shows how to close any open tables of application group data:

   ```
   arstblsp -a 1 -g credit
   ```

4. The following shows how to close the specified table of application group data:

   ```
   arstblsp -a 1 -g credit -t AAA26
   ```

5. The following shows how to migrate tables of application group data. The arstblsp command migrates each table of application group data to its own table space:

   ```
   arstblsp -a 0 -g credit
   ```

6. The following shows how to migrate a specific table of application group data:

   ```
   arstblsp -a 0 -g credit -t AAA26
   ```

Notes

Before running the arstblsp command on a UNIX server for the first time, you must relink the server programs. See the *Installation and Configuration Guide* for details.

Do not load data into OnDemand while migrating tables with the arstblsp command.

Files

/usr/lpp/ars/bin/arstblsp
The AIX executable program.

/opt/ondemand/bin/arstblsp
The HP-UX and Solaris executable program.

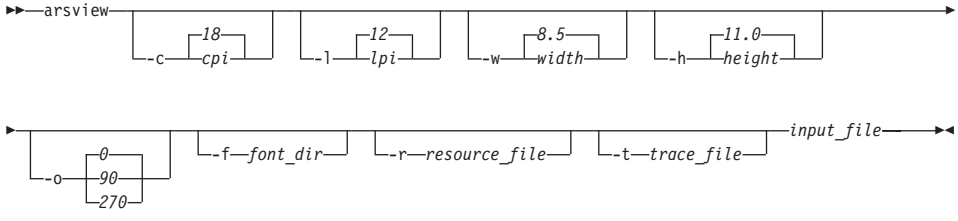
\Program Files\IBM\OnDemand for WinNT\bin\arstblsp
The Windows NT executable program.

Chapter 31. arsview command reference

Purpose

The arsview command reads an AFP document and creates a line data file suitable for viewing on a 3270 or other line-mode terminal.

Syntax



Description

The OnDemand CICS/ESA client program displays items retrieved from the database on 3270, line-mode terminals. If the selected item contains AFP data, OnDemand first calls the arsview command to read the AFP data, process the command options, and create a line data file. OnDemand then passes the line data file to the client program. The arsview command can also be used as a general purpose AFP to line data batch conversion utility. You can invoke the arsview command from a user-defined program or type the command at the prompt.

You can optionally specify the directory where the arsview command can locate user-defined character set and code page files to process the input file. Refer to the *Installation Guide for OnDemand Clients* for information about defining character set and code page files for use with OnDemand client programs.

The arsview command ignores any graphics, images, and typographic fonts in the input file. All fonts in the input file are mapped to a single point, monospaced font. You can specify the name of the resource group file that contains the form definition and overlay associated with the input file. The arsview command can process information contained in the form definition, such as the offset, and write the overlay text to the output file.

Parameters

-c cpi Characters per inch in the horizontal direction of the smallest (or only) font in the input file. For example, if the input file contains text at 12 characters per inch and 10 characters per inch, you would specify **-c 12**. The default value is 18 characters per inch.

-l lpi Lines per inch in the vertical direction of the smallest (or only) font in the input file. The default value is 12 lines per inch.

-w width
The width of a page in the input file. The **-w** and **-h** values define the paper size used to print a page of the input file. The default value is 8.5 inches.

-h height
The height of a page in the input file. The **-w** and **-h** values define the paper size used to print a page of the input file. The default value is 11.0 inches.

-o orientation
The orientation of the data on the page in the input file. You can specify that the data appears rotated 0, 90, or 270 degrees. The default value is 0 degrees.

-f font_dir
The name of the directory that contains user-defined character set and code page files required to process the input file. Refer to the *Installation Guide for OnDemand Clients* for information about defining character set and code page files for use with OnDemand client programs.

-r resource_file
The name of the resource group file that contains the form definition and overlay associated with the input file. The arsview command can process information contained in the form definition, such as the offset, and include text from an overlay in the output file.

-t trace_file
Specifies the name of the file where the arsview command writes debugging information. You can use this option to obtain detailed information about how the arsview command translates the input file.

input_file
The name of the source (AFP) file.

Examples

1. The following shows how to read the file 37271600.AFP (located in the current directory) and redirect the line data output to the file 37271600.LIN in the current directory. By default, the input file contains 8.5 x 11 inch pages:

   ```
   arsview -c 12 -l 8 -o 90 37271600.AFP > 37271600.LIN
   ```

2. The following shows how to process an input file using the default options for characters per inch, lines per inch, width, height, and orientation. The output is redirected to stdout:

   ```
   arsview 96089900.AFP
   ```

Files

/usr/lpp/ars/bin/arsview
The AIX executable program.

/opt/ondemand/bin/arsview
The HP-UX and Solaris executable program.

\Program Files\IBM\OnDemand for WinNT\bin\arsview
The Windows NT executable program.

Part 6. Appendixes

Index

A

accessing
  files 23, 33
  image files 23
  index data 5
  input data 33
  reports 5
  resources 5
adding storage volumes 81
ADSM
  adding storage volumes 81
  administrative client 75
  command line interface 75
  database
    backup and recovery 58
  database recovery 61
  device configuration file 60
  halting 75
  labeling storage volumes 79
  managing storage 65
  private volumes 77
  recovering the database 61
  recovery log 59
  removing storage volumes 83
  report backup and recovery 62
  restarting 75
  scratch volumes 77
  starting 75, 89, 117
  stopping 75, 91, 93, 117
  storage 65
  storage volumes 79, 81, 83
  volume history file 60
alerts 95
allocating storage space
  for data loading 4
API 98
application
  DATASET parameter 203, 207, 208
  requirements for data loading 5, 21, 31
application group
  FORMS parameter 203, 207, 208
  requirements for data loading 5, 19, 30
application programming interface
  system logging user exit 98
  user exit 98
applications
  loading image files 19
  loading index data 3, 19, 29
  loading reports 3
  loading user-defined data 29
  storing data 3, 19, 29
ars_adsm
  ADSM database backup 16
  backup the ADSM database 16
  command reference 165
  database backup 16
arsadm
  command reference 139
arsadmin
  command reference 153
  unloading a report 41
arsdate
  command reference 169
arsdb
  backup the OnDemand database 16
  command reference 173
  database backup 16
  OnDemand database backup 16
  starting 89
arsdoc
  command reference 181
arsjesd
  command reference 201
  starting 90, 117
  stopping 91, 117
arsload
  arsload.cfg file 7
  checking messages 17
  command reference 205
  configuring 7, 24, 34
  messages 17
  password 7
  starting 91, 117
  userid 7
arsload.cfg file
  configuring 7
  password 7
  userid 7
arslog
  system log user exit 98
  user exit 98
arsmaint
  command reference 213
arsobjd
  starting 90, 117
  stopping 91, 93, 117
arssockd
  starting 90, 117
  stopping 91, 93, 117
arstblsp
  command reference 219
arsview
  command reference 223

B

backing up databases
  after loading reports 15
backup and recovery
  ADSM database 58, 61
  ADSM device configuration file 60
  ADSM recovery log 59
  ADSM volume history file 60
  control files 51
  database 52, 56
  DB2 UDB 52
  OnDemand 51
  OnDemand database 52, 56
  overview 51
  product software 51
  recovery log 59
  reports and resources 62
backup the ADSM database 16
backup the OnDemand database 16
batch processing
  add documents 181
  arsdoc 181
  create a document list 181
  create a hit list 181
  delete documents 181
  retrieve documents from the database 181
  retrieve items from the database 181
  update the database 181

C

cache file systems
  expiration processing 213
  migration processing 213
  monitoring utilization 103
CD-ROM creation manifest
  reviewing 97

check messages and verify totals
arsload processing 17
column delimiter 163
command reference
overview 133
commands
ars_adsm 89, 165
arsadm 139
arsadmin 153
arsdate 169
arsdb 89, 173
arsdoc 181
arsjesd 90, 201
arsload 91, 205
arslog 98
arsmaint 213
arsobjd 90
arssockd 90
arstblsp 219
arsview 223
compress
arsadmin function 155
decompress
arsadmin function 155
how to read a syntax diagram 135
import
arsadmin function 154
indexer
arsadmin function 156
load
arsadmin function 153
load_db
arsadmin function 153
postprocess_cmd
arsadmin function 155
res_diff
arsadmin function 155
retrieve
arsadmin function 154
starting ADSM 89
starting arsjesd 90
starting arsload 91
starting arsobjd 90
starting arssockd 90
starting DB2 UDB 89
starting Download for OS/390 90
starting MVS Download 90
starting the database 89
starting the server program 90
store
arsadmin function 154
unload
arsadmin function 153
compress
arsadmin command function 155
compression
estimating 155
concurrent users
monitoring 97, 121
configuring the arsload command 7, 24, 34

D

data
Download for OS/390 5, 43
download user exits 43
indexing 23, 33
indexing on OnDemand servers 10
indexing on the MVS and OS/390 systems 11
loading 1, 3, 19, 29
loading, user exits 43
MVS Download 5, 43
database
backup 15, 53
backup and recovery 52
backup image 54
expiration processing 213
expiring index data 45
importing index data 45, 154
importing tables 154
index maintenance 45
loading data 13
logging 55
maintaining index data 45
migrating index data 45
migration 45, 154
migration processing 213
recovery 56
segment table 13
starting 89
updating 181
DATASET parameter
application 203, 207, 208
arsload command 203, 207, 208
date
format 169
internal format 169
obtaining OnDemand internal format 169
OnDemand internal format 169
DB2 UDB
backup and recovery 52
processes 100
starting 89, 117
stopping 91, 94, 117
decompress
arsadmin command function 155
deleting a report 41
deleting index data 45
document list
generating 181
documents
adding 181
deleting 181
retrieving 181
Download for OS/390
arsjesd command reference 201
exits 43
processing 43
starting 90, 117
starting related programs 90
transmitting data 5

E

errors and alerts 95
estimating and allocating storage space
for data loading 4
expiration processing 213
expiring index data 45

F

file systems
monitoring 103
folder
requirements for data loading 21, 32
system log 96
FORMS parameter
application group 203, 207, 208
arsload command 203, 207, 208

G

generic index data
creating 23, 33
generic index file
creating 23, 33
Generic Index File 24, 34

H

halting
OnDemand 91, 117
servers 91, 117
hit list
generating 181
how to read a syntax diagram 135

I

image data
indexing 23

image file
indexing 23
loading 19
processing 25
storing 19
import
arsadmin command
function 154
index data from archive
media 154
importing index data 45
index data
accessing 5
creating 23, 33
downloading 5
loading 3, 13, 19
loading the database 3, 19
processing 12, 25, 35
transferring 5
index file
creating 23, 33
index maintenance
database 45
expiring index data 45
importing 45
maintaining index data 45
migration 45
indexer
arsadmin command
function 156
indexing
restarting a load process 39
indexing files 23, 33
indexing reports on MVS and OS/390 systems 11
Indexing Reports on MVS and OS/390 Systems 11
indexing reports on OnDemand servers 10
Indexing Reports on OnDemand Servers 10
internal date format 169

L

labeling storage volumes 79
load
arsadmin command
function 153
load_db
arsadmin command
function 153
loading data 1, 3, 19, 29
accessing
files 23
image files 23
index data 5
loading data 1, 3, 19, 29 *(continued)*
accessing *(continued)*
reports 5
resources 5
accessing input files 33
application group
requirements 5, 19, 30
application requirements 21, 31
arsload.cfg file 7
arsload command 7, 24, 34
folder requirements 21, 32
loading image files 19
loading index data 3, 13, 19, 29
loading reports 1, 3
preparing 4
loading storage objects 14
loading user-defined data 29
logging facility 96
API definition 98
user exit 98

M

maintaining data 1
maintaining index data 45
managing ADSM storage 65
managing archive media 65
manifest
reviewing CD-ROM contents 97
messages
arsload processing 17, 26, 36
system logging facility 17, 26, 36
viewing the system log 96
migrating
table spaces 219
migrating index data 45
migration 154
migration processing 213
monitoring
ADSM processes 101
cache file systems 103
concurrent users 97, 121
DB2 UDB processes 100
disks and file systems 103
file systems 103
I/O activity 103
memory usage 102
network 105
OnDemand processes 98
performance 121
processes 98
server 95, 119
storage volumes 104
system load 101
system paging space 102
monitoring *(continued)*
users 97, 121
MVS Download
arsjesd command reference 201
exits 43
processing 43
starting 90, 117
starting related programs 90
stopping 91, 117
transmitting data 5

O

obtaining OnDemand internal date format 169
OnDemand
alerts 95
command reference 133
errors and alerts 95
internal date format 169
logging facility 96
operating servers 87, 107
starting 89
system logging facility 96
OnDemand database
backup image 54
deleting database indexes 45
expiring database indexes 45
index maintenance 45
maintaining index data 45
migrating database indexes 45
recovery 56
operating
servers 87, 107
starting servers 89
operating the server
backup and recovery 51
monitoring 95, 119
MVS programs 90
OS/390 programs 90
recovery 51
restarting 94
starting MVS programs 90
starting OS/390 programs 90
stopping 91, 117

P

performance
monitoring 121
postprocess_cmd
arsadmin command
function 155
preparing to load reports 4
private volumes 77
processes
ADSM 101

processes *(continued)*
monitoring 98
OnDemand 98
processing image files 25
processing index data, reports, and resources 12
processing index data and image files 25
processing index data and input files 35
processing input files 35

R

recovering the ADSM database 61
recovery 51
removing a report 41
removing storage volumes 83
report
accessing 5
backup and recovery 62
deleting from OnDemand 41
indexing on MVS and OS/390 systems 11
indexing on OnDemand servers 10
loading 1, 3
preparing to load 4
processing 12
removing from OnDemand 41
restarting a load process 39
storing 3
unloading from OnDemand 41
res_diff
arsadmin command function 155
resource file
input to arsadmin load command 162
input to arsadmin res_diff command 163
processing 12
resources
accessing 5
downloading 5
transferring 5
restarting a load process 39
restarting OnDemand 94
restarting servers 94
retrieve
arsadmin command function 154
reviewing a CD-ROM creation manifest 97

S

scratch volumes 77
searching for and viewing messages 96
segment table 13
server
ADSM storage 65
archive media 65
backup and recovery 51
managing ADSM storage 65
managing archive media 65
monitoring 95, 119
operating 87, 107
recovery 51
starting 89
stopping 91, 117
server program
starting 90, 117
stopping 93
services
starting 117
stopping 117
shutting down
OnDemand 91, 117
servers 91, 117
starting
ADSM 89, 117
arsjesd 90, 117
arsload 91, 117
arsobjd 90, 117
arssockd 90, 117
database 89
DB2 89, 117
Download for OS/390 90
MVS Download 90, 117
MVS programs 90
OS/390 programs 90
server program 90, 117
stopping
ADSM 91, 93, 117
arsjesd 91, 117
arsobjd 91, 117
arssockd 91, 117
DB2 91, 94, 117
Download for OS/390 117
MVS Download 91, 117
OnDemand 91, 117
servers 91, 117
storage objects
loading 14
storage volumes
adding 81
ADSM 79, 81, 83
labeling 79
removing 83
store
arsadmin command function 154
syntax diagram
how to read 135
system log folder 96
system logging facility 96
API definition 98
arsload messages 17, 26, 36
messages 17, 26, 36
user exit 98

T

table spaces
migrating 219
tables
segment 13
TIFF file
loading 19
storing 19

U

unload
arsadmin command function 153
unloading a report 41
user-defined data
indexing 33
loading 29
processing 35
user exit 98
user exits
data download 43
data loading 43
Download for OS/390 43
downloading data 43
loading data 43
MVS Download 43
users
monitoring 97, 121

Readers' Comments — We'd Like to Hear from You

EDMSuite OnDemand
Administrator's Reference
Version 2.2

Publication No. S544-5293-08

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

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.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
S544-5293-08

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Information Development
IBM Software Solutions
Department MYGA Building 001L
PO Box 1900
Boulder, CO 80301-9817

Fold and Tape **Please do not staple** Fold and Tape

S544-5293-08

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

S544-5293-08

Spine information:

EDMSuite OnDemand Administrator's Reference Version 2.2