

Content Manager OnDemand for z/OS
Version 10 Release 1

Administration Guide



Note

Before using this information and the product it supports, read the information in [“Notices” on page 387.](#)

This edition applies to the following products and to all subsequent releases and modifications until otherwise indicated in new editions:

- Version 10 Release 1 of IBM® Content Manager OnDemand for z/OS® (product number 5697-CM1)

© **Copyright 2017 - 2018 All Rights Reserved. UNICOM Systems, Inc. – a division of UNICOM Global.**

© **Copyright International Business Machines Corporation 2001, 2018.**

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

Contents

Chapter 1. Database concepts.....	1
Database.....	1
Table.....	1
Table space.....	1
Index.....	1
System catalog tables.....	2
Buffer pool.....	2
Chapter 2. Database maintenance.....	3
Collecting statistics.....	3
How to collect statistics.....	3
Removing index data.....	4
How to remove index data.....	5
Migrating indexes.....	6
Chapter 3. Migrating and importing index data.....	9
Configuring the system.....	9
System log messages.....	9
System log user exit program.....	9
Application groups.....	10
What happens when a user queries migrated data.....	10
Message to the user.....	10
Message to the system log.....	10
Importing index data.....	11
Verify temporary work space.....	11
Run the ARSADMIN program.....	11
After you import index data.....	11
Querying data.....	11
Expiring imported migrated indexes.....	11
Configuring index migration.....	12
Chapter 4. Using the administrative client.....	13
Installation.....	13
System requirements.....	13
Installing the administrative client.....	13
Running Setup.....	14
To install on a network file server.....	14
To install on a user's PC.....	14
To uninstall.....	14
Updating the administrative client software.....	15
Chapter 5. Administrative client introduction.....	17
How does the administrative client help me do my job?.....	17
Getting started.....	17
Starting the administrative client.....	17
Using online help.....	18
Adding a server.....	18
Logging on a server.....	19
Changing passwords.....	19
System parameters.....	19

Maximum Password Age.....	19
Minimum Password Length.....	20
Session Inactivity Time Out.....	20
System Logging.....	20
User Exit Logging.....	20
Login Processing.....	21
Annotations.....	21
System log comments.....	22
Login Processing.....	22
Administrative client start up parameters.....	22
Setting system parameters.....	23
Changing start up parameters.....	24
Displaying the Content Manager OnDemand splash screen or About window.....	24
Default directory for administrative client temporary files.....	24
Adding items to a server.....	25
Report Wizard.....	26
Starting the Report Wizard.....	26
Report Wizard screens.....	27
Chapter 6. Administration concepts.....	29
Introduction.....	29
Users.....	29
Groups.....	31
Printers.....	31
Storage sets and storage nodes.....	32
Application groups.....	32
Applications.....	32
Folders.....	33
Cabinets.....	33
Holds.....	34
About application groups, applications, and folders.....	34
Content Manager OnDemand permissions.....	35
Folder permissions.....	36
Application group permissions.....	36
Specifying permissions.....	37
Chapter 7. Advanced system administration.....	41
Overview.....	41
User types.....	41
Authority.....	42
Permissions.....	44
Users.....	44
Groups.....	44
Applications.....	44
Application Groups.....	45
Folders.....	45
Storage sets.....	46
Printers.....	46
Cabinets.....	47
Holds.....	47
System administration.....	47
Object Type model.....	48
Object Owner model.....	49
Summary.....	50
Helpful hints.....	50
Chapter 8. LDAP (Lightweight Directory Access Protocol) authentication support..	51

Chapter 9. Examples of using the administrative client.....	53
Overview.....	53
About the examples.....	53
System configuration.....	53
Adding a user.....	54
Adding a group.....	55
Adding a printer.....	58
Adding a storage set and storage node.....	60
Adding a report.....	64
Chapter 10. Loading AFP™ and line data.....	75
Overview.....	75
Before you begin loading input data.....	75
Estimating storage space.....	75
Defining the application group.....	76
Defining the application.....	76
Configuring the ARSLOAD program.....	77
Specifying input data to process.....	77
Input file names.....	77
Userid and password.....	78
Indexing input data.....	78
Processing the input data.....	78
Processing index data.....	79
Processing reports and resources.....	79
Loading index data.....	79
Loading storage objects.....	80
Cache storage.....	80
Archive storage.....	80
Resources.....	80
Verifying processing.....	80
Backing up databases.....	81
Chapter 11. Loading image files.....	83
Overview.....	83
Defining the application group.....	83
Defining the application.....	84
Defining the folder.....	85
Accessing the input files.....	86
Creating index data.....	86
Configuring the ARSLOAD program.....	88
Processing the input data.....	88
Chapter 12. Loading user-defined data.....	91
Overview.....	91
Defining the application group.....	91
Defining the application.....	92
Defining the folder.....	93
Accessing the input files.....	94
Creating the index data.....	94
Configuring the ARSLOAD program.....	95
Processing the input data.....	96
Chapter 13. Restarting a load process.....	99
Chapter 14. Deleting a report.....	101

Chapter 15. Importing and exporting administrative objects through an XML interface.....	103
Overview.....	103
Files provided.....	104
Importing an XML file into a Content Manager OnDemand system.....	104
Preparing an XML file for the import process.....	105
Creating an XML file.....	105
Importing the XML file by using the ARSXML command.....	111
Exporting Content Manager OnDemand administrative objects to an XML file.....	111
Chapter 16. Commands overview.....	113
Running Content Manager OnDemand server commands.....	113
Reading a syntax diagram.....	114
Using quotes in commands.....	116
Omitting a userid and password.....	117
Removing trailing blank characters.....	117
Using the h and I parameters.....	117
Chapter 17. ARSADMIN.....	119
Purpose.....	119
Description and syntax.....	119
Parameters.....	121
Examples.....	125
Notes.....	125
Files.....	126
Chapter 18. ARSAFPD.....	127
Purpose.....	127
Syntax.....	127
Description.....	127
Parameters.....	127
Examples.....	128
Chapter 19. ARSDATE.....	131
Purpose.....	131
Syntax.....	131
Description.....	131
Parameters.....	132
Examples.....	135
Notes.....	135
Files.....	136
Chapter 20. ARSDB.....	137
Purpose.....	137
Syntax.....	137
Description.....	137
Parameters.....	137
Examples.....	139
Files.....	139
Chapter 21. ARSDOC.....	141
Purpose.....	141
Description and syntax.....	141
GET function.....	141
QUERY function.....	143

ADD function.....	145
UPDATE function.....	147
DELETE function.....	149
PRINT function.....	150
HOLD_ADD function.....	151
HOLD_RELEASE function.....	153
CFSOD_FED function.....	154
FTI_ADD function.....	155
FTI_RELEASE function.....	156
Parameters.....	157
Examples.....	170
Exit codes used by ARSDOC.....	175
Notes.....	175
Files.....	176
Chapter 22. ARSEXOAM.....	177
Purpose.....	177
Input and output.....	177
Control.....	177
Job control statements.....	177
Utility control statements.....	178
Example.....	179
Chapter 23. ARSEXPRI.....	181
Purpose.....	181
Input and output.....	181
Control.....	181
Job control statements.....	181
Utility control statements.....	182
Example.....	183
Chapter 24. ARSLOAD.....	185
Purpose.....	185
Syntax.....	185
Description.....	186
Parameters.....	187
Examples.....	195
Sample JCL.....	195
About the JCL statements.....	196
Notes.....	197
Files.....	197
Chapter 25. ARSMAINT.....	199
Purpose.....	199
Syntax.....	199
Description.....	199
Parameters.....	200
Examples.....	204
Notes.....	204
Files.....	205
Chapter 26. ARSRPT.....	207
Purpose.....	207
Syntax.....	207
Description.....	207
Parameters.....	207
Examples.....	209

Notes.....	210
Files.....	210
Chapter 27. ARSSLU.....	211
Purpose.....	211
Syntax.....	211
Parameters.....	211
Examples.....	212
Chapter 28. ARSSOCKD.....	213
Purpose.....	213
Syntax.....	213
Description.....	213
Parameters.....	213
Examples.....	214
Notes.....	216
Chapter 29. ARSSTASH.....	217
Purpose.....	217
Syntax.....	217
Description.....	217
Parameters.....	217
Examples.....	218
Files.....	218
Chapter 30. ARSTBLSP.....	219
Purpose.....	219
Syntax.....	219
Description.....	219
Parameters.....	219
Examples.....	220
Notes.....	220
Files.....	221
Chapter 31. ARSXAFP.....	223
Purpose.....	223
Syntax.....	223
Description.....	223
Parameters.....	224
Examples.....	225
Files.....	225
Chapter 32. ARSXLIN.....	227
Purpose.....	227
Syntax.....	227
Description.....	227
Parameters.....	227
Examples.....	230
Chapter 33. ARSXML.....	231
Purpose.....	231
Syntax.....	231
Description.....	232
Parameters for ARSXML add, update, delete.....	233
Parameters for ARSXML export.....	235
Parameters for ARSXML validate.....	237

Examples.....	237
Example 1: Adding users.....	237
Example 2: Updating users.....	237
Example 3: Deleting users.....	238
Example 4: Exporting multiple objects.....	238
Example 5: Adding groups.....	238
Example 6: Exporting a group.....	239
Example 7: Updating groups.....	239
Example 8: Exporting all of the users.....	239
Example 9: Validating input XML file schema.....	239
Example 10: Adding printers.....	240
Chapter 34. ARSYSPIN.....	241
Introduction.....	241
JCL requirements.....	242
Control statements.....	243
Keyword descriptions.....	246
ARSYSPIN user exits.....	261
Overview.....	261
Processing environment.....	262
Data areas.....	263
Logical exit points.....	264
Security system (SAF/RACF) considerations.....	266
ARSYSPIN files.....	266
ARSYSPIN intermediate file structure.....	266
ARSYSPIN error codes.....	271
ABEND codes.....	271
Reason codes.....	272
ARSSPVIN: Sample APKACIF exit routine.....	278
Special considerations for APKACIF exits written in COBOL.....	279
Chapter 35. Operator commands.....	281
START command.....	281
MODIFY command.....	281
STOP command.....	284
Appendix A. Appendixes to the Administration Guide.....	285
Accessibility features.....	285
ARSSUPPORT utility.....	286
System control tables.....	287
List of tables.....	287
Application group table (ARSAG).....	288
Application group to folder field mapping table (ARSAG2FOL).....	289
Application group field table (ARSAGFLD).....	290
Application group field alias table (ARSAGFLDALIAS).....	291
Application group composite index table (ARSAGINDEX).....	291
Application group permissions table (ARSAGPERMS).....	292
Annotation table (ARSANN).....	292
Application table (ARSAPP).....	293
User logical views table (ARSAPPUSR).....	294
Cabinet table (ARSCAB).....	295
Cabinet to folder mapping table (ARSCAB2FOL).....	295
Cabinet permissions table (ARSCABPERMS).....	296
CFS-CMOD work table (ARSCFSODWORK).....	296
Folder table (ARSFOL).....	297
Folder field table (ARSFOLFLD).....	297
Folder user fields table (ARSFOLFLDUSR).....	298

Folder permissions table (ARSFOLPERMS).....	299
FTI work table (ARSFTIWORK).....	299
Group table (ARSGROUP).....	300
Hold table (ARSHOLD).....	301
Hold mapping table (ARSHOLDMAP).....	301
Hold permissions table (ARSHOLDPERMS).....	302
Hold work table (ARSHOLDWORK).....	302
Load table (ARSLOAD).....	303
Named query table (ARSNAMEQ).....	303
Node table (ARSNODE).....	304
ODF distribution definition table (ARSDFDCT).....	304
ODF distribution request table (ARSDFDRT).....	305
ODF document status table (ARSDFDST).....	306
ODF print processor table (ARSDFPPT).....	307
ODF recipient list table (ARSDFLIS).....	309
ODF recipient/list table (ARSDFUOT).....	309
ODF report bundle definition table (ARSDFBBDT).....	310
ODF report ID table (ARSDFCRT).....	311
Printer table (ARSPRT).....	312
Printer options table (ARSPRTOPTS).....	312
Printer user table (ARSPRTUSR).....	313
Resources table (ARSRES).....	313
Segment table (ARSSEG).....	314
Storage set table (ARSSET).....	315
System parameters table (ARSSYS).....	316
User table (ARSUSER).....	317
Users in groups table (ARSUSRGRP).....	319
User group ID table (ARSUSRGRPID).....	320
Application group data table.....	320
Tablespaces for Content Manager OnDemand system tables.....	321
Objects and data model used in the Content Manager OnDemand XML file.....	322
XML objects in the Content Manager OnDemand XML file.....	322
System Parameters object.....	324
User object.....	327
Group object.....	331
Printer object.....	332
Storage set object.....	333
Application group object.....	334
Application object.....	343
Folder object.....	358
Cabinet object.....	367
Hold object.....	369
ODF recipient object.....	371
ODF recipient list object.....	372
ODF report ID object.....	373
ODF distribution object with report bundle object.....	373
Data field limitations.....	376
Troubleshooting.....	377
Troubleshooting scenario 1: Unable to combine two applications in one application group.....	377
Troubleshooting scenario 2: ARSLOAD error.....	378
Troubleshooting scenario 3: ARSLOAD not loading data from HFS directory specified with -d parameter.....	378
Troubleshooting scenario 4: ARSLOAD failed while specifying the host with the -h parameter.....	379
Troubleshooting scenario 5: Error message EDC5084I	380
Troubleshooting scenario 6: ARSMAINT fails with ARS0020E.....	380
Troubleshooting scenario 7: ARSADMIN fails while attempting to run from USS.....	381
Troubleshooting scenario 8: ARS0025E messages in ARSLOAD.....	381
Troubleshooting scenario 9: Error occurred during the CD-ROM mastering process.....	382

Troubleshooting scenario 10: The .ARD file cannot be removed after the loading by using load service.....	382
Troubleshooting scenario 11: Error reading resource file while loading AFP documents with ARSLOAD.....	382
Troubleshooting scenario 12: While running ARSMANT to expire cache objects, the job ends with return code 13.....	383
Troubleshooting scenario 13: Empty file created while using ARSLOAD to monitor an HFS directory.....	383
Troubleshooting scenario 14: The ARSMANT -f command produces only usage instructions.....	384
Troubleshooting scenario 15: DB2 reports that OnDemand database is almost full.....	385
Notices.....	387
Trademarks.....	388
Terms and conditions for product documentation.....	389
IBM Online Privacy Statement.....	389
Privacy policy considerations	390
Index.....	391

ibm.com[®] and related resources

Product support and documentation are available from ibm.com[®].

Support and assistance

From ibm.com, click **Support & downloads** and select the type of support that you need. From the Support Portal, you can search for product information, download fixes, open service requests, and access other tools and resources.

IBM Knowledge Center

See your online product information in IBM Knowledge Center at https://www.ibm.com/support/knowledgecenter/SSQHWE_10.1.0/com.ibm.ondemandtoc.doc/ondemandzos_10.1.0.htm.

PDF publications

See the following PDF publications for your product at <http://www.ibm.com/support/docview.wss?uid=swg27050852>.

Contacting IBM

For general inquiries, call 800-IBM-4YOU (800-426-4968). To contact IBM customer service in the United States or Canada, call 1-800-IBM-SERV (1-800-426-7378).

For more information about how to contact IBM, including TTY service, see the Contact IBM website at <http://www.ibm.com/contact/us/>.

Chapter 1. Database concepts

This section provides an overview of the key database objects:

- Database
- Table
- Index
- Table space
- System catalog tables
- Buffer pool

Database

A collection of data that is stored in tables. In Content Manager OnDemand, generally speaking, there are two types of tables:

- System tables, which contain information about the objects you define to the system, such as users, groups, application groups, applications, holds, folders, storage sets, printers, and cabinets. There are also system tables that contain information Content Manager OnDemand uses to control and maintain the system.
- Application group tables, which contain the index data for the reports that you load on the system

A database is organized into parts called *table spaces*.

Table

A table consists of data logically arranged in columns and rows. For example, when you create an application group, the system creates a table definition that contains one column for each field that you define. When you load a report into an application group, the system adds one row to an application group table for each document contained in the report.

Table space

A database is organized into table spaces. A table space is a place to store tables.

Content Manager OnDemand system and application group tables exist in their own table spaces by default. The Content Manager OnDemand system table spaces are listed in [“System control tables” on page 287](#). The names of the application group table spaces are created by OnDemand.

Index

In Content Manager OnDemand, an index points to a document. An index allows more efficient access to documents by creating a direct path to a document through pointers. You can index all of the content in documents.

You define indexes when you create an application group. The indexes should contain information that uniquely identify a document, such as date, account number, and customer name. Indexes are populated by values extracted from a report when you load a report on the system. Each row in an application group table identifies one document.

However, keep in mind that you do not want lots of indexes on a report just to have indexes. You should have a good business reason to have an index. While indexes can help you find documents faster, having

too many of them can slow you down when you load reports on the system. Every time you add a new row (document) to a table, you have to add a row to each and every one of the indexes for that table. So the more indexes that you have, the longer it may take when you load a report.

System catalog tables

Each database includes a set of system catalog tables, which describe the logical and physical structure of the data. The database manager creates and maintains an extensive set of system catalog tables for each database. These tables contain information about the definitions of the database objects, such as user tables, views, and indexes, as well as security information about the authority that users have for these objects. They are created when the database is created, and are updated in the course of normal operation. You cannot explicitly create or drop them, but you can query and view their contents using the catalog views.

Buffer pool

A buffer pool is an allocation of main memory allocated to cache table and index data pages as they are being read from disk or being modified. The purpose of buffer pools is to improve database system performance. Data can be accessed much faster from memory than from a disk; therefore, the fewer times the database manager needs to read from or write to a disk, the better the performance.

The configuration of the buffer pool is the single most important tuning area, since you can reduce the delay caused by slow physical I/O.

Chapter 2. Database maintenance

You need to maintain the Content Manager OnDemand database to keep it performing in an optimal manner. Run the following database maintenance tasks on a regular basis:

- Collect statistics on tables to keep optimization information up-to-date
- Remove index data that has reached its life of data and indexes period
- Migrate indexes to archive storage, if your organization needs to keep indexes for some period of time after users no longer need to retrieve the documents to which they point

Collecting statistics

About this task

Statistics describe the physical and logical characteristics of a table and its indexes. You should collect table and index statistics periodically for each active table. These statistics are used by the database manager to determine a good way to access the data. If the data has changed significantly, to the extent that the information last collected no longer reflects the actual table data, then performance may begin to deteriorate when users are accessing data.

Collect statistics at least once a week. You may also want to collect statistics after loading data on the system. For example, when you load data into an application group that uses Multiple Loads per Database Table, Content Manager OnDemand may add rows to an existing table. After the load completes, the information used by the SQL query optimizer does not reflect the latest updates to the table. Before you allow users to access the data, you should collect statistics on the table.

Content Manager OnDemand lets you collect statistics for all of the tables in the database with the ARSMaint program. When you run the ARSMaint program to collect statistics, it collects statistics on all of the tables in the database that have changed since the last time that you collected statistics. You can automate the collection of statistics by scheduling the ARSMaint program to run with the appropriate options.

You can also manually collect statistics by running the ARSMaint program from the prompt. See [Chapter 25, “ARSMaint,” on page 199](#) for details about the parameters and options that you can specify.

How to collect statistics

You can control the automatic collecting of statistics by scheduling the ARSMaint program to run with the appropriate options.

You can also manually collect statistics by running the ARSMaint program from the prompt. For example, to collect statistics on all tables in the database that have changed since the last time that statistics were collected:

```
arsmaint -r
```

When the ARSMaint program collects statistics on a table, it saves the following message in the system log:

```
164 ApplGp Segment Maintenance (ApplGp) (Segment)
```

One message is saved in the system log for each table on which statistics were collected.

Removing index data

About this task

Indexes *expire* (are eligible for removal) because their life of data period has passed. The indexes, and the documents that they point to, can then be removed from the system. When you remove an index, information about the document to which it points is removed from the database (the document can no longer be retrieved). However, because indexes are eligible to be removed does not mean that they will be deleted from the database. Content Manager OnDemand does not delete expired index data from the database until expiration processing runs.

The application group expiration policy determines when index data is eligible for deletion from the database. You define the expiration policy when you create the application group. The following properties on the Storage Management page comprise the expiration policy:

- **Life of Data and Indexes.** The length of time in days to maintain index data and documents on the system. After the index data has been on the system for this number of days, it is eligible to be deleted.
Note: If you specify Never Expire, then expiration processing is disabled for the application group. (That is, index data will not be removed from the database.)
- **Expiration Type.** Determines whether individual indexes or an entire table of index data is deleted at a time. When Content Manager OnDemand deletes index data, it either deletes a row (if the Expiration Type is Document) or drops a table (if the Expiration Type is Segment) or deletes all the rows associated with the load (if the Expiration Type is Load). The amount of index data in a table and the number of reports the data represents is determined by the Database Organization. If the Database Organization is Multiple Loads per Database Table, then by default, a table of index data can hold up to 10 million indexes. These types of tables usually hold the indexes for many reports.

A table of index data is not eligible to be deleted until the latest date in any of its rows reaches the Life of Data and Indexes period. For example, suppose that the Life Of Data and Indexes is set to 365 days, the Expiration Type is set to Segment, and the Database Organization is set to Multiple Loads per Database Table. By default, a table will contain approximately 10 million rows. Further, suppose that a report is loaded into the application group once every month and that each report adds one million rows to the database. Each table can hold the index data from approximately ten reports. Using these assumptions, the data that is loaded into the application group in January will not be eligible to be deleted by expiration processing until November of the following year. If you need to remove the index data for a report as soon as it reaches its Life of Data and Indexes period, then set the Database Organization to Single Load per Database Table and set the Expiration Type to Segment or Load. (And run expiration processing at least once a month.)

Content Manager OnDemand and the archive storage manager delete the documents that expired index data points to independently of each other. Content Manager OnDemand uses the application group's expiration policy to determine when indexes and documents expire and should be removed from the system. The archive storage manager marks documents for removal based on the criteria specified in the archive copy group. However, you should specify the same criteria to Content Manager OnDemand and the archive storage manager. The Life of Data and Indexes, which is used by Content Manager OnDemand, and the Retention Period, which is used by the archive storage manager, should specify the same value.

Content Manager OnDemand does not explicitly delete data stored with segment or document expiration from Tivoli® Storage Manager, however, the data might still expire in the storage manager based on its expiration settings.

The following pictures show an example of expiration processing. For purposes of the example, assume that the Life of Data and Indexes is 365 days, the Database Organization is Single Load per Database Table, and the Expiration Type is Load. Further, assume that one report is loaded into the application group every month, beginning on January 15, 1999, and that expiration processing has never been run on this particular application group.

Figure 1 on page 5 shows an example of the application group index data before expiration processing begins. The table on the left represents the *segment table* for the application group. A segment table

contains one row for each table of application group data. In the example, a table of application group data contains the index records for one report. A row in the segment table contains the latest date found in the report (or the load date, if the report does not contain a date). For expiration processing, Content Manager OnDemand uses the date from the segment table to determine when to drop a table.

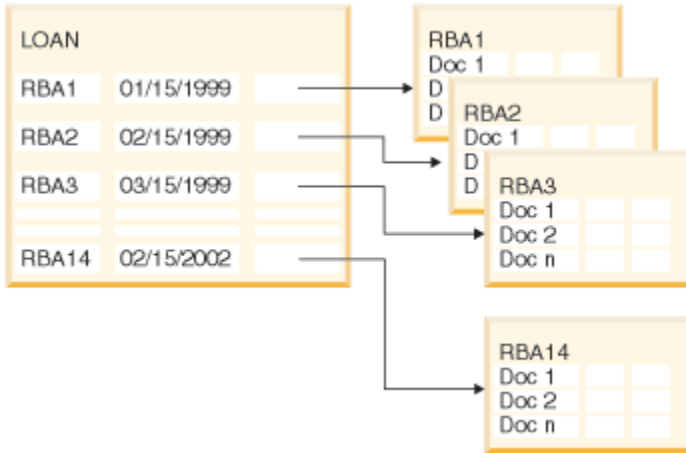


Figure 1: Removing index data. Part 1 of 2

Figure 2 on page 5 shows an example of the application group index data after expiration processing ends. For purposes of the example, assume that expiration processing ran on March 4, 2000. That date, along with the criteria specified in the expiration policy (specifically, the Life of Data and Indexes is 365 days) causes the ARSMAINT program to drop two tables of application group index data: 1RBA, which has a date of January 15, 1999, and 2RBA, which has a date of February 15, 1999. Content Manager OnDemand also deleted the rows in the segment table that pointed to the application group tables that were dropped.

Application Group - LOAN

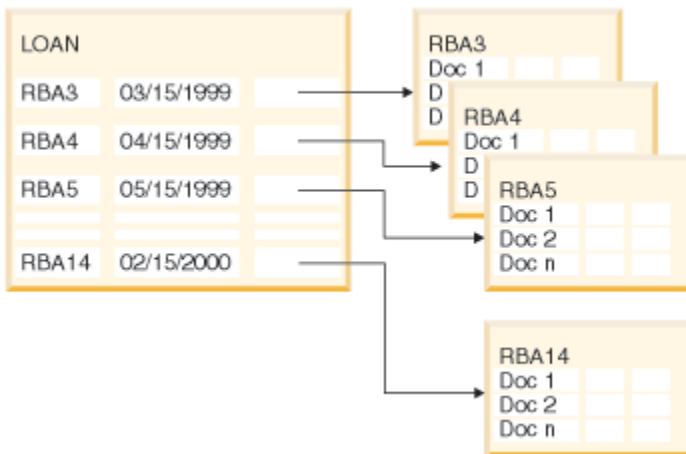


Figure 2: Removing index data. Part 2 of 2

How to remove index data

Content Manager OnDemand does not delete expired index data from the database until expiration processing runs. The ARSMAINT program is the expiration utility. You can schedule the ARSMAINT program to run automatically or you can run it manually. You should make sure that the ARSMAINT program command runs periodically so that Content Manager OnDemand deletes indexes when it is time to do so (so that expired documents can no longer be retrieved). See Chapter 25, “ARSMAINT,” on page 199 for details about the parameters and options that you can specify.

You can control automatic expiration processing by scheduling the ARSMAINT program to run with the appropriate options.

You can also manually start expiration processing by running the ARSMaint program from the prompt. For example, to run expiration processing, enter the following command at the prompt:

```
arsmaint -d
```

When the ARSMaint program removes indexes, it saves the following message in the system log:

```
128 ApplGrp Segment Expire (ApplGrp) (Segment)
```

One message is saved in the system log for each table that was dropped during expiration processing.

Migrating indexes

About this task

Note: This section provides an overview of the process of migrating index data from the database to archive storage. See "Migrating and importing index data" for information about configuring the system for migration processing.

Content Manager OnDemand provides automatic migration to move indexes from the database to archive storage to maintain seldom used indexes for long periods of time.

Important: If you use migration to move indexes to archive storage, make sure that you migrate them after there is no longer a need to retrieve the documents to which they point.

The ARSMaint program uses an application group's migration policy to control when migration of indexes for an application group occurs:

- **Migration of Indexes.** If you specify No Migration, then migration of indexes is disabled for the application group. (That is, index data will not be migrated.) If you specify Migrate After *n* Days, then index data is eligible to be migrated after reaching the specified number of days. Indexes will be migrated the next time that the ARSMaint program runs.
- **Life of Data and Indexes.** The length of time in days to maintain index data on the system. For migration, this value must be greater than the Migrate After *n* Days value.

Content Manager OnDemand does not migrate index data from the database to archive media until migration processing runs. The ARSMaint program is the migration utility. You can control automatic migration processing by scheduling the ARSMaint program to run with the appropriate options. You can also manually start migration processing by running the ARSMaint program from the prompt. See [Chapter 25, "ARSMaint," on page 199](#) for more information about the ARSMaint program.

The ARSMaint program migrates indexes from each file system listed in the table space file system file.

After a migrated table is successfully loaded into the System Migration application group, the table is dropped from the database. However, Content Manager OnDemand keeps track of all migrated tables. That way, if index data in a migrated table is needed, then Content Manager OnDemand can alert an administrator to take action (such as manually import the table back into the database).

You can control automatic migration processing by scheduling the ARSMaint program to run with the appropriate options. See [Chapter 25, "ARSMaint," on page 199](#) for more information about the ARSMaint program.

You can also manually start migration processing by running the ARSMaint program from the prompt. For example, to run migration processing, enter the following command at the prompt:

```
arsmaint -e
```

When the ARSMANT program migrates indexes, it saves the following messages in the system log. A set of three messages should be saved in the system log for each table that is migrated from the database to archive storage:

```
166 ApplGroup Segment Export (ApplGrp) (Segment)
14 DB Info Exported (SQL Code)
87 ApplGrp Load (System Migration)
```

The first message identifies a segment of application group index data that is to be migrated from the database to archive storage. The second message reports the status of exporting the table from the database to temporary storage. The third message reports the status of loading the migrated table into the System Migration application group. The System Migration application group must be assigned to a storage set that identifies a client node that is maintained by the archive storage manager.

Chapter 3. Migrating and importing index data

About this task

Migration is the process by which Content Manager OnDemand moves index data from the database to archive storage. 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 information, 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 from archive storage to the database. (And only then can the user's query be completed.) After maintaining the imported table in the database for a specified number of days, Content Manager OnDemand deletes it from the database.

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

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

Important: Before you can migrate index data, the index tables must be closed. If the Database Organization for the application group is set to Single Load per table, the index table is closed when the report is loaded. Otherwise, if the Database Organization is Multiple Loads per table, the index table is closed when the Maximum Rows value is reached. To close a table to loading before the Maximum Rows value is reached, use the ARSTBLSP command with the -a1 parameter. For more information, see [Chapter 30, "ARSTBLSP,"](#) on page 219.

Configuring the system

About this task

System log messages

Content Manager OnDemand provides the system log for administrators to monitor the system. When you install and configure Content Manager OnDemand, you initialize the system log tables. The system log is critical to the operation of the system.

When Content Manager OnDemand processes a query for application group data that has been migrated to archive storage, it saves a message in the system log, sends the message to the system log user exit program, and sends the message to the console.

You can configure the system to examine the messages that Content Manager OnDemand sends to the system log user exit and mail them to an administrator or send them to another program. You can also configure the system log user exit program to determine what action to take when a user queries for data that has been migrated to archive storage.

System log user exit program

When a client queries index data that has been migrated to archive storage, Content Manager OnDemand saves message number 168 ApplGp Segment Not Available in the system log. Content Manager OnDemand also sends the message to the system log user exit. If you have written your own system log user exit program, you can determine the action to take when Content Manager OnDemand sends the

message to the system log user exit. For example, you may want the user-written program to notify an administrator that a request for a table of migrated index data has occurred.

See the *IBM Content Manager OnDemand for z/OS: Configuration Guide* for more information about the system log user exit.

Application groups

When you define an application group, you specify the storage management information that determines how long Content Manager OnDemand maintains data stored in the application group and when Content Manager OnDemand takes certain actions. For example:

- **Life of Data and Indexes.** Determines the length of time that Content Manager OnDemand maintains the index data and report data that is stored in the application group.
- **Migration of Indexes.** Determines the number of days before Content Manager OnDemand moves index data from the database to archive storage.

You should plan to migrate index data only after users no longer need to access the reports to which it refers. Only in exceptional situations should users need to access index data that has been migrated. If a user needs to access index data that has been migrated to archive storage, the process of importing the table back into the database requires manual actions by an administrator, and usually results in a significant delay in completing the query. The import process also requires additional space in the database to hold the imported tables, additional log file storage, and temporary storage on the server to run the import process.

- **Keep Imported Migrated Indexes.** Determines how long Content Manager OnDemand maintains the imported index data in the database before it is scheduled for deletion.

If you need to maintain index data in archive storage, then you must configure the Migration of Indexes in your application groups. If you want Content Manager OnDemand to maintain the imported index data in the database for longer than 30 days, then you must specify the number of days in Keep Imported Migrated Indexes. Content Manager OnDemand will schedule imported index data for deletion from the database 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.

You can use the administrative client to configure your application groups.

What happens when a user queries migrated data

Message to the user

When the server determines that the index data required to complete a query has been migrated to archive storage, it sends a message to the client program. The message states that the data required to complete the query is not available and that the user should contact an administrator.

Message to the system log

When Content Manager OnDemand determines that the index data required to complete a query has been migrated to archive storage, it saves a message in the system log. An administrator can open the System Log folder to search for and display messages in the system log.

Content Manager OnDemand also sends a message to the console and the system log user exit program. You can configure the system to examine the message and send an alert to an administrator or call another program to take some action.

If you do not use some other facility that routes messages to an administrator or another program, IBM strongly encourages you to configure a user-defined program to process the message. The function of the program can vary, however at a minimum, IBM recommends that the program notify a system administrator when a query for migrated index data occurs.

Importing index data

About this task

Verify temporary work space

About this task

Importing migrated index data from archive storage back into the database requires temporary storage. Before you import the index data, you should verify that sufficient free space is available in the location that Content Manager OnDemand uses for temporary storage. The ARS_TMP parameter in the ARS.CFG file determines the location that Content Manager OnDemand uses for temporary storage.

Run the ARSADMIN program

About this task

Content Manager OnDemand provides the ARSADMIN program to import tables of index data from archive storage to the database.

The following examples shows how to run the ARSADMIN program on a USS system to copy a migrated index table from archive storage back into the database. The name of the application group and the index table to import can be obtained from the message that OnDemand saved in the system log.

```
arsadmin import -g 'Credit' -u admin -p "" AAA27
```

After the ARSADMIN program completes the import operation, you can open the System Log folder to see the messages that were generated by the import process.

See Chapter 17, “ARSADMIN,” on page 119 for a description of the ARSADMIN program and information about the parameters and options.

After you import index data

Querying data

About this task

After you import index data from archive storage back into the database, you should notify the user to retry the query.

Expiring imported migrated indexes

About this task

Content Manager OnDemand marks an imported index table for deletion after it resides in the database for the number of days specified by the Length of Time to Keep Imported Indexes property of the application group. After an imported index table reaches the specified value, the next time that the ARSMaint program runs, the imported index table is deleted from the database. (However, the table still exists in archive storage.)

You typically configure the ARSMaint program to run automatically on a regular schedule. This may be done using the CRON facility at installations using USS (UNIX System Services). You can also run the ARSMaint program command from the TSO OMVS command prompt. See Chapter 25, “ARSMaint,” on page 199 for a description of the ARSMaint program, information about the parameters that you can specify, and sample JCL.

Configuring index migration

About this task

If you find that your users are often querying for index data that has been migrated to archive storage, configure your application groups to increase the length of time that Content Manager OnDemand maintains the index data in the database. This should reduce the number of queries that need migrated index data.

Keeping imported migrated indexes

About this task

Content Manager OnDemand schedules imported index data for deletion after the index data 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 thirty days. If you want Content Manager OnDemand to maintain imported index data in the database for longer than thirty days, then you must change the value of Keep Imported Migrated Indexes for your application groups.

Chapter 4. Using the administrative client

This section contains information about the administrative client, a program that you can use to define and maintain Content Manager OnDemand objects on a Content Manager OnDemand system. For example, you can use the administrative client to define reports to Content Manager OnDemand and maintain Content Manager OnDemand users, groups, printers, and storage sets.

This section describes a typical system, including requirements about users of the system and data to be loaded on the system, and contains examples and ideas about how to use the administrative client to implement the requirements. In it you'll find:

- An introduction to the administrative client
- Advice about what is required to get your work done
- Pointers to detailed how-to information in the online Help

The information provided in this part gives you a frame of reference. If you need more information, the book tells you where to look in the online Help. If you can't find answers to your questions, the book tells you how to contact IBM. This part is organized in roughly the order that you define your first report to Content Manager OnDemand: it starts with adding users and groups to the system, shows how to define a printer, and then moves into procedures for adding a storage set and defining a primary storage node. Later chapters provide detailed instructions about defining a report: adding an application group, an application, and a folder.

Note: IBM assumes that you are familiar with the Windows operating environment and using applications that run under Windows. IBM assumes that you know how to start programs, use a pointing device, such as a mouse, and that you are familiar with terminology, such as window, dialog box, entry field, button, scroll bar, list, and check box.

Installation

About this task

System requirements

For administrative client system requirements, see: <http://www.ibm.com/support/docview.wss?uid=swg27049168>

Installing the administrative client

About this task

You can install all of the Content Manager OnDemand client features at once, or individual features as you need them.

Important: If you plan to use the report wizard or the graphical indexer from the administrative client to process PDF input files, then you must first install the full Adobe Acrobat product or Adobe Acrobat Approval. For information about how to obtain Adobe Acrobat products, see Adobe on the Web at <http://www.adobe.com>. OnDemand provides the ARSPDF32.API file to enable PDF viewing from the client. If you install the client after you install Adobe Acrobat, then the installation program will copy the API file to the Acrobat plug-in directory. If you install the client before you install Adobe Acrobat, then you must copy the API file to the Acrobat plug-in directory. Also, if you upgrade to a new version of Acrobat, then you must copy the API file to the new Acrobat plug-in directory. The default location of the API file is `\Program Files\IBM\OnDemand Clients\V10.1\PDF`. The default Acrobat plug-in directory is

\Program Files\Adobe\Acrobat x.y\Acrobat\Plug_ins, where x.y is the version of Acrobat, for example, 4.0, 5.0, and so forth.

Running Setup

About this task

When you run the Setup program, the Setup screens show the names of the Content Manager OnDemand client directories so you know where the files are being placed.

Information about network installations is available in the *IBM Content Manager OnDemand: Windows Client Customization Guide*.

To install on a network file server

About this task

For instructions on how to install the administrative client on a network file server, see the "Installing client software on a network" in *IBM Content Manager OnDemand: Windows Client Customization Guide*.

To install on a user's PC

About this task

To install, follow these instructions:

Procedure

1. From the Windows taskbar, click **Start**, and then choose **Run**.
2. Type `x:\client\win32\setup` (where *x* is the letter of the clients install image).
3. Click **OK**.

Results

After the Setup program starts, click **Next** to continue.

On the **Setup Type and Working Directory** dialog box, select **Custom** and then click **Next** to continue.

On the **Installation Components** dialog box, select **Administrator**. (You can also select other components that you want to install on the PC.)

Click **Next** to continue. Follow the instructions on the screen to complete the installation.

To uninstall

About this task

You can remove the Content Manager OnDemand client from your PC by using the program removal function of Windows. The program removal function removes the directories for Content Manager OnDemand and any references to Content Manager OnDemand in system files.

Procedure

Do the following steps:

1. From the Windows taskbar, click Start. Choose Settings, and then choose Control Panel.
2. Select the appropriate program removal function.
 - For Windows 7, select Uninstall a program from **Control Panel > Programs > Programs and Features**.
 - For other versions of Windows, double-click **Add/Remove Programs**.

3. From the list, select the appropriate Content Manager OnDemand program to remove.
 - For Windows 7, select **IBM OnDemand Clients V10.1**
 - For other versions of Windows, select **Content Manager OnDemand Clients V10.1**.
4. Click on the appropriate program removal function.
 - For Windows 7, click **Uninstall**.
 - For other versions of Windows, click **Add/Remove**.
5. Click OK.

Updating the administrative client software

About this task

Note: When upgrading the administrative client, the Setup program automatically replaces all out-of-date files with new ones; the user is not permitted to add or remove components. To add or remove components, the user must run the Setup program after successfully upgrading the administrative client.

The following information can be used to obtain and install the latest version of the Content Manager OnDemand client software from IBM service on the Web.

To get the latest version of the Content Manager OnDemand client software, see <http://www.ibm.com/support/fixcentral/>

Results

After the download is complete, you can expand the ZIP file or store the ZIP file in a shared location, depending on how you plan to distribute the administrative client software:

- A user can expand the ZIP file to a temporary directory and then run the Setup program to upgrade the administrative client on a PC. After installing the administrative client, the user can delete the temporary directory and the ZIP file.
- An administrator can store the ZIP file in a shared location so that other users can access it.
- An administrator can expand the ZIP file to a shared location so that other users can run the Setup program from the shared location.
- If your organization shares a copy of the Content Manager OnDemand administrative client software from a network server, then an administrator can expand the ZIP file and run the Setup program to upgrade the administrative client on the network server.
- If your organization distributes user-defined files with the Content Manager OnDemand client software, an administrator must expand the ZIP file to a shared location and then copy the user-defined files to that location. Users can then run the Setup program from the shared location to upgrade the client software and the user-defined files on their PCs.

Chapter 5. Administrative client introduction

How does the administrative client help me do my job?

Welcome to the Content Manager OnDemand administrative client, a powerful and flexible program designed to help you maintain your Content Manager OnDemand system.

The administrative client provides tools to:

- Define reports to the system
- Add and maintain Content Manager OnDemand users and groups
- Add and maintain server printers
- Add and maintain storage sets and storage nodes
- Create summaries about users, groups, applications, application groups, storage sets, folders, printers, holds, and cabinets
- Add and maintain servers
- Set system parameters for Content Manager OnDemand servers and client programs
- Copy items from one Content Manager OnDemand server to another
- Track changes made to the system. When you use the administrative client to add or update an object in Content Manager OnDemand, information about the object is saved in the system log. The information includes the changes that you made.

Getting started

Starting the administrative client

About this task

To start the administrative client:

Procedure

1. Click **Start**.
2. Select **Programs**, then choose IBM OnDemand Clients V10.1.
3. Click OnDemand Administrator V10.1.

When you start the administrative client, Content Manager OnDemand opens the administrator window that contains a menu bar, toolbar, navigator pane, list pane, and status bar.

- You can collapse and expand areas in the navigator pane (on the left) to make it easier to see. A plus sign next to an area means there are items inside.
- When you click on an area in the navigator pane, the items appear in the list pane (on the right).
- To make the panes narrower or wider, point to the vertical bar between the two panes of the window until the pointer turns into a two-headed arrow. Then click and hold the left mouse button and drag it in either direction.
- Use buttons on the toolbar to switch between the different ways to look at items: large or small icons, a list, or details.
- To query the server for a new list of items, press the F5 key or select Refresh List from the View menu.

- After you log on to a server, the status bar shows the Content Manager OnDemand user ID and the name and version number of the Content Manager OnDemand server.

Using online help

About this task

Online help provides information to assist you with completing tasks. The administrative client online help contains information about the options, fields, and commands on the windows, dialog boxes, and property sheets that you see when using the program.

To display the online help, press F1 any time the administrative client is active in Windows. Help is available for dialog box commands and options. The main help topic for each dialog box usually contains information about the kinds of tasks you can perform. For example, the online help about Logical Views lets you learn how to create public and private logical views. The online help provides brief procedures rather than lengthy descriptions.

To display an index of help topics, select the Index tab from the Help window. You can type keywords to locate topics in the help file.

To search the index of help topics, select the Search tab from the Help window. You can type search words to locate related topics in the help file.

Adding a server

About this task

You can use the New Server command to add a server.

Procedure

1. Select Content Manager OnDemand Servers.
2. From the File menu, select **New Server** to open the **Add a Server** window.
3. Type the name of the server in the Server field. The server name identifies the server in the navigator pane of the administrator window. You can use an alias, the actual computer or network name of the workstation, or any other identifier you choose. By default, the administrative client copies what you type to the Host Name field.
4. Verify the value of the Host Name field. (By default, the Host Name field contains the same value as the Server field.) The host name can be a host name alias, fully-qualified host name, or IP address of the server.
5. Verify the Protocol. Choose from TCP/IP and Local:
 - TCP/IP. Use TCP/IP (Transmission Control Protocol/Internet Protocol) as the network protocol. To use TCP/IP, the server and the client must include TCP/IP in the protocol stack.
 - Local. Define a server that uses the file system on this workstation. After you define a local server, you must use the Setup command to initialize the database on the local server.
6. For TCP/IP, verify the Port Number over which the library server and the client communicate.

The default value, 0 (zero), means that the server and client communicate over the port number assigned to the Content Manager OnDemand TCP/IP service. If the Content Manager OnDemand TCP/IP service is not assigned a port number, the server and client will attempt to communicate over port number 1445. If the Content Manager OnDemand TCP/IP service is not assigned a port number and you plan to use a port number other than 1445, then you must enter a valid port number. The value range is from 0 to 65535.
7. If you are adding a Local server, specify a Directory and select an Operating System and a Database. See the online help for assistance.
8. If you are adding a server that supports Secure Sockets Layer (SSL), select **User Secure Sockets Layer**. If your server does not support SSL, SSL is not used even if you select this check box.

9. Click **OK** to add the server.

Logging on a server

Procedure

1. Point to the library server and double click the left mouse button to open the log on dialog box.
2. Type your Content Manager OnDemand userid and password in the spaces provided and click OK.

Changing passwords

About this task

Note: Depending on how you set the Login Processing system parameters, the case of a password may be significant. See [“System parameters” on page 19](#) for details.

Changing a password

Procedure

To change a user's password:

1. Select and expand the library server.
2. Select Users.
3. In the User ID list, point to the userid and click the right mouse button.
4. From the pop-up menu, select Update to open the Update a User dialog box.
5. Type the new password in the Password field.
6. Verify the new password by retyping it into the Verify Password field.
7. Click OK. Content Manager OnDemand updates the database and returns to the main window.

Changing an expired password

About this task

Content Manager OnDemand provides password expiration processing to help you manage security on the system. You can set a value that represents the time in days that passwords assigned to users remain valid. After a user's password reaches the value that you specify, the user must change the password. See [“System parameters” on page 19](#) for information about how Content Manager OnDemand expires passwords.

After a password reaches the expiration value, the next time the userid is used to log on to a server, Content Manager OnDemand prompts the user to enter a new password.

The user must enter the current password for the userid, a new password, and verify the new password by retyping the new password

System parameters

Content Manager OnDemand system parameters allow you to establish the operational settings for client programs and servers.

Maximum Password Age

Sets a time limit for passwords and determines when Content Manager OnDemand prompts users to change passwords. The default setting is `Password Never Expires`, meaning that passwords do not expire and Content Manager OnDemand never prompts users to change passwords. If you select `Password Always Expires`, then users must change to new passwords each time that they log on to a

server. To set a specific time limit for passwords, select `Expires In __ Days` and enter the number of days that passwords are valid in the space provided. The value can be from 1 (one) to 365.

Minimum Password Length

Determines whether passwords are required. If passwords are required, also determines the fewest number of characters that passwords can contain. The default value is `At Least 8 Characters`, meaning that passwords must contain at least eight characters.

When a user changes a password, the client checks the number of characters that the user typed. The new password must contain the minimum number of characters. Otherwise, the client issues an error message. If your organization has not implemented the security user exit and the user types more than eight characters, then the client issues an error message.

Session Inactivity Time Out

Determines when Content Manager OnDemand terminates sessions between inactive clients and the server. The default setting, `Never Time Out`, means that Content Manager OnDemand does not terminate a session, regardless of how long the client remains inactive. To set a specific inactivity time out, select `Time Out In __ Minutes` and enter the number of minutes in the space provided. The value can be from 1 (one) to 1440 (24 hours).

The period of inactivity is measured between requests to a server. For example, when a user enters a query, Content Manager OnDemand searches the database and builds the document list. This completes a request to the server. If the user does not work with the items in the document list, open another folder, or invoke another query before the inactivity timeout occurs, Content Manager OnDemand automatically terminates the session with the client.

Use caution when you set the inactivity time out. For example, assume that you set the inactivity time out to 10 (ten). You log on to Content Manager OnDemand to add an application group. Creating the application group takes you 15 minutes to complete. After entering all of the information about the application group, you click OK to create the application group. Content Manager OnDemand issues a message that shows a timeout has occurred. You must logoff the server, and you cannot save the information you entered about the application group.

System Logging

Determines the messages that Content Manager OnDemand saves in the system log. Content Manager OnDemand provides the system log to help you track activity and monitor the system. Content Manager OnDemand save messages that are generated by the various programs, such as the ARSLOAD program. Content Manager OnDemand can save a message in the system log when the following events occur:

- A user logs on to the system
- A user logs off the system
- A user logon fails
- Application group data is queried, retrieved, loaded, updated, deleted, or maintained

User Exit Logging

Content Manager OnDemand provides a user exit at each of the four system log event points. These exits allow you to filter the messages that Content Manager OnDemand saves in the system log and take action when a particular event occurs. For example, you can provide a user exit program that sends a message to a security administrator when someone attempts and fails to log on to the system. You can also use a user exit to determine what information appears in the system log

If you plan to migrate index data to archive storage, then we recommend that you configure the system to save application group messages in the system log and send them to a system log user exit program. You should design a system log user exit program to notify an administrator when a query for migrated data occurs. Before a query for migrated data can be completed, an administrator must import a copy of the table or tables that are required from archive storage to the database.

See the *IBM Content Manager OnDemand for z/OS: Configuration Guide* for help with configuring the system log user exit.

Login Processing

The login processing system parameters allow you to specify whether userids and passwords are case sensitive.

By default, userids and passwords are case insensitive. When you add a user, Content Manager OnDemand converts lowercase letters in the userid to uppercase. A person can type letters in a userid in uppercase, lowercase, or mixed case letters. For example, if you add the userid LaGuarde, a person can enter LAGUARDE, laguarde, or LaGuarde to log on to the server.

If you select UserID Case Sensitive, then a user must type the userid exactly as it was entered when the user was added. For example, if you add the userid LaGuarde, then the user must enter LaGuarde to log on to the server.

If you select Password Case Sensitive, then a user must type the password exactly as it was entered when the user was added. For example, if you set the password to Spring2Far, then the user must enter Spring2Far to log on to the system.

Important: You should decide whether to make userids and passwords case sensitive when you install the system, change the defaults if necessary, and do not change the settings again. Otherwise:

- If userids are initially case insensitive and you later choose UserID Case Sensitive, then userids that were added before you changed the parameter must be entered in uppercase. The same is true for passwords.
- If userids are initially case sensitive and you later clear UserID Case Sensitive, then the userids that were added before you changed the parameter that contain mixed or lowercase letters will no longer be valid. The same is true for passwords.

If users log on to Content Manager OnDemand with the CICS® client program, you should configure the system to ignore the case of userids and passwords.

Annotations

Specify the following options:

- Which types of annotations (referred to as "notes" in the Content Manager OnDemand client) can be added by a user.
- The default for the viewing scope for all annotations.
- Whether annotations can expire. If they can expire, specify the number of days to keep the annotation. Afterward, Content Manager OnDemand removes the annotation.

These options apply to all users with authority to add annotations in the system.

There are three types of annotations that a user can add:

Allow Public

Allows the user to add public annotations. Public annotations to a document can be viewed by anyone who opens that document.

Allow Private to User

Allows the user to add private annotations to a document, and those annotations can be viewed only by the user that created the note, application group administrators, and system administrators.

Allow Private to Group

Allows the user to add annotations to a document, and those annotations can be viewed only by a specific group of users.

The **Default Annotation Type** section specifies the default for the viewing scope for annotations.

The **Life of Annotations** section specifies whether annotations can expire. If you want annotations to be removed, select **Expire After**, then specify the number of days to keep an annotation. After the number of days passes, Content Manager OnDemand removes all of the annotations.

System log comments

The system log comments specify whether the administrative client displays the System Log Comments window when you perform an add, update, or delete operation.

Enable comments

Select this option to display the System Log Comments window when you perform an add, update, or delete operation.

Require comments

This option requires the user to enter one or more characters in the Comments field.

Login Processing

The login processing system parameters allow you to specify whether userids and passwords are case sensitive.

By default, userids and passwords are case insensitive. When you add a user, Content Manager OnDemand converts lowercase letters in the userid to uppercase. A person can type letters in a userid in uppercase, lowercase, or mixed case letters. For example, if you add the userid LaGuaRde, a person can enter LAGUARDE, laguearde, or LaGuaRde to log on to the server.

If you select UserID Case Sensitive, then a user must type the userid exactly as it was entered when the user was added. For example, if you add the userid LaGuaRde, then the user must enter LaGuaRde to log on to the server.

If you select Password Case Sensitive, then a user must type the password exactly as it was entered when the user was added. For example, if you set the password to Spring2Far, then the user must enter Spring2Far to log on to the system.

Important: You should decide whether to make userids and passwords case sensitive when you install the system, change the defaults if necessary, and do not change the settings again. Otherwise:

- If userids are initially case insensitive and you later choose UserID Case Sensitive, then userids that were added before you changed the parameter must be entered in uppercase. The same is true for passwords.
- If userids are initially case sensitive and you later clear UserID Case Sensitive, then the userids that were added before you changed the parameter that contain mixed or lowercase letters will no longer be valid. The same is true for passwords.

If users log on to Content Manager OnDemand with the CICS client program, you should configure the system to ignore the case of userids and passwords.

Administrative client start up parameters

Content Manager OnDemand provides parameters that you can specify as properties that the operating system uses when you start the administrative client program. The parameters can be used, for example, to automate the logon process and to select the areas that appear in the navigator pane. [Table 1 on page 22](#) lists the start up parameters for the administrative client.

Parameter	Purpose	Example
/A	Include Cabinets in the navigator pane.	/A
/B	Include Applications in the navigator pane.	/B
/C	Include Data Distribution in the navigator pane. This parameter is valid only if you install the Production Data Distribution feature on the library server.	/C

<i>Table 1: Administrative client start up parameters (continued)</i>		
Parameter	Purpose	Example
/D	Include Folders in the navigator pane.	/D
/D 2	Include Folders in the navigator pane, but display only the Permissions and Field Information pages.	/D 2
/E	Include Storage Sets in the navigator pane.	/E
/I	Include Users in the navigator pane.	/I
/N	Include IBM Content Manager OnDemand Report Distribution for Multiplatforms in the navigator pane.	/N
/O	Include Groups in the navigator pane.	/O
<i>/P password</i>	The password for the Content Manager OnDemand user identified with the /U parameter. Use with the /S and /U parameters to log on to a specific server, without displaying the Logon dialog box. If the logon is not successful, then Content Manager OnDemand opens the Logon dialog box to allow the user to log on to the system.	/P password
/Q	Include Application Groups in the navigator pane.	/Q
/R	Include Printers in the navigator pane.	/R
<i>/S server</i>	Identifies the logon server. Use with the /U and /P parameters to log on to a specific server, without displaying the Logon dialog box. If the logon is not successful, the Content Manager OnDemand opens the Logon dialog box to allow the user to log on to the system.	/S broncos
/T	Determines the name that appears on the title bar of the administrator window. The default name is Content Manager OnDemand Administrator.	/T Customer Service
<i>/U userid</i>	Identifies the Content Manager OnDemand user. Use with the /S and /P parameters to log on to a specific server, without displaying the Logon dialog box. If the logon is not successful, Content Manager OnDemand opens the Logon dialog box to allow the user to log on to the system.	/U admin

Setting system parameters

To set the system parameters for a Content Manager OnDemand server:

1. Log on to the library server.
2. Point to the library server and click the right mouse button.

3. From the pop-up menu, select **System Parameters** to open the **System Parameters** dialog box.
4. Type information in the fields and make selections for each page.
5. When you have finished making changes to the system parameters, click **Update**. Content Manager OnDemand stores the changes in the database and returns to the administrator window. (Click **Cancel** to close the **System Parameters** dialog box without saving your changes.)

Changing start up parameters

About this task

To change start up parameters, you need to change the shortcut you use to start the administrative client.

Procedure

To change start up parameters, do the following tasks. Remember that the specific steps vary between different levels of Windows.

1. Right-click on the administrative client shortcut.
2. From the pop-up menu, select **Properties**.
3. Click the **Shortcut** tab.
4. In the **Target** field, enter the startup parameters.
5. Click **OK**.

Results

The next time that you start the administrative client, Windows uses the start up parameters and values that you specified.

Displaying the Content Manager OnDemand splash screen or About window

About this task

When the administrative client is first started, a Content Manager OnDemand splash screen is displayed.

To disable the splash screen, set the SHOWLOGO parameter to 0.

To change the display time, locate the section `My Computer\HKEY_CURRENT_USER\Software\IBM\OnDemand32\Preferences`. Add a new string under Preferences called SHOWLOGO. Set the string value to a value of zero or more seconds.

For more information about customizing the Content Manager OnDemand administrative client, read the technical document [Customizing the IBM Content Manager OnDemand Administrative Client](#).

Default directory for administrative client temporary files

This topic introduces the default directory for administrative client temporary files.

The administrative client creates temporary files such as summaries for users, groups, and folders.

The administrative client stores these temporary files in the following default directory: `%APPDATA%\IBM\OnDemand Client\REPORT`, where `%APPDATA%` is an environment variable that resolves to different values depending on the operating system and user name.

Each user in a multiple user environment has a temporary directory for summaries. The summary window includes a text field that contains a default value for the name of the file that contains the summary. If a different directory or file name is required, you can edit the text field before you create the summary.

Adding items to a server

This section explains how to add items to a server. You can use commands or a drag-and-drop operation to add items to a server.

Note: When you use the administrative client to add or update the database, Content Manager OnDemand adds a record to the system log that shows the changes you made.

New command

After logging on to a server, select the area, for example, Users. From the File menu, select the New command to open the Add dialog box.

Copy command

After logging on to a server, select the area. In the list pane, point to the item that you want to copy and click the right mouse button. From the pop-up menu, select Copy to open the Add dialog box. The fields in the Add dialog box contain information copied from the item you selected. Before you can add the item, you must change the item name. Depending on the item you want to add, you may need to change other fields.

Export command

The Export command is like the Copy command, except Content Manager OnDemand adds the item to a different library server. You can use the Export command to export items from one library server (the source server) and add them to another (the destination server).

After logging on to the server that contains the item you want to export, select the area. In the list pane, point to the item that you want to export and click the right mouse button. From the pop-up menu, select Export to open the Export dialog box. Verify the destination server. Then click Export to add the item to the server.

Note: If the item exists on the destination server, the export fails.

To retain permissions when you export users or groups from one library server to another library server, export them in the following order:

1. Printers
2. Users
3. Groups
4. Storage sets
5. Application groups
6. Folders
7. Cabinets
8. Holds

If you have added users or groups to an application group or folder that you want to export, you should export the users and groups first. Otherwise, OnDemand issues a warning and does not export the data. However, if you want to export the application group or folder without the users or groups, click the **Ignore Warnings** check box. OnDemand generates a warning message for each user or group that cannot be added.

If you export applications to a target server, note the following:

- You can export an application only to an application group with the same name as the application group that is associated with the application being exported.
- An application group with the same name as the application group that is associated with the application to be exported must already exist on the target server.

- The application group on the target server must have the same database fields as the application group that is associated with the application to be exported.
- The application to be exported does not already exist in the application group on the target server.
- The application identifier of the application that is being exported is not already used by an application in the application group on the target server.
- If the application identifier of the application being exported is not already defined in the application group, the `Displayed Value` and `Database Value` pair is added to the application group as part of the export process.

Drag and drop operation

You can copy and export items using a drag-and-drop operation. For example, to export items from one library server and add them to another, select one or more items from the list pane and, while holding the left mouse button down, point to the destination server. Then release the mouse button. If you are logged on to the destination server, then Content Manager OnDemand opens the Export dialog box. If you are not logged on to the destination server, then Content Manager OnDemand opens the Logon dialog box. After verifying options in the Export dialog box, click Export to copy the items to the server.

Note: If the item exists on the destination server, the export fails.

You can also use a drag-and-drop operation to copy an item. For example, to copy a user, select the userid from the User ID list and, while holding the left mouse button down, point to the same server on which the user is listed. Then release the mouse button to open the Add dialog box.

Report Wizard

Content Manager OnDemand provides user assistance and easy-to-use tools to help you administer the system. The Report Wizard assists you in adding reports to the system. The Report Wizard helps you add a report to the system by asking questions, which allows you to progress in an organized manner toward completing an application group, application, and folder. The Report Wizard assists you in adding applications to an application group.

Here are a few things to remember about the Report Wizard:

- You move through the Report Wizard by answering questions that appear on the screen.
- You can return to the previous screen at any time by clicking **Previous**.
- You can advance to the next screen at any time by clicking **Next**.
- You can advance to the final screen by clicking **Finish**. By choosing **Finish**, you permit the Report Wizard to make all remaining decisions for you.
- You can obtain online help for a screen at any time by clicking **Help** or pressing F1.

You can use the Report Wizard to add an application group, application, and folder for a report. These actions include defining indexing information, defining database and folder fields, configuring data and storage management, specifying whether the application group can contain more than one application, and naming the application group, application, and folder.

You can also use the Report Wizard to add an application to an existing application group. This action includes defining indexing information, specifying storage information, and identifying the application within the application group. To add an application to an application group, the application group must have a database field to hold the values that uniquely identify an application within the application group. The field must contain at least one unassigned application identifier. For more information about application identifiers, see the **Field Information** page in application groups.

Starting the Report Wizard

About this task

From the administrative client, log on to the library server to which you want to add the report.

- To define a new application group, application, and folder, click **Report Wizard** on the toolbar.
- To add an application to an existing application group:
 - Under the server, select **Application Groups**.
 - Select the name of the application group to which you want to add the application.
 - Click **Report Wizard** on the toolbar.

Follow the on-screen instructions to add the report.

Report Wizard screens

As you move through the Report Wizard, standard options are selected for you. Unless you have a clear reason not to, accept the defaults. Depending on how you use the Report Wizard, you might not see all of the screens described.

Introduction screen

Provides a brief explanation of the Report Wizard. Your first step is to select the indexer that you want to index the data. For all indexers, you specify the type of data you want to store. For indexers other than Generic and XML, you specify the location of sample data.

Report window

Displays the sample data and provides easy-to-use tools to help you define indexing information, database fields, and folder fields. Press F1 to display the online help for options and commands available from the Report window. Use the online help to learn how to define triggers, fields, and indexes, database fields, and folder fields.

Important: When you finish defining the indexing, database, and folder information, save your changes.

View information screen

If you specify User Defined as the data type, specify a file extension. If you specify Line as the data type, specify the code page, carriage control, and record format.

Managing fields screen

When you select the Generic indexer or XML indexer, you add and remove database and folder fields on this page.

When you click on **Add** or **Properties**, the Report Wizard displays a window where you specify the properties of a field.

Managing data screen

When you load a report into the system, you can specify that you want report data to be stored using large object support. You also need to specify how you want Content Manager OnDemand to manage annotations that users attach to pages of the report.

Application identifier screen

When you use the Report Wizard to add an application to an existing application group, you must specify the name of the application and select a value that uniquely identifies the application within the application group.

Storage management screen

Determines where the storage manager maintains copies of reports, and how and when Content Manager OnDemand deletes report data from the system.

Applications in the application group screen

If the report that you are defining is one of several that will be stored in the same application group, then you can use the Report Wizard to define the database field that contains the values that identify an application within the application group, define the folder field that users use to search a specific application, and specify the length of the application ID field.

If you select the check box for **Document Size Field**, Content Manager OnDemand adds a field to the application group and folder. Content Manager OnDemand stores the size of the document in the application group field when data is loaded. If you select the check box for **Page Count Field**, Content Manager OnDemand adds a field to the application group and folder. Content Manager OnDemand stores the number of pages in the document in the application group field when data is loaded. You must provide the folder names for the two fields. You do not need to specify names for the application group fields because they are predefined.

Enhanced Retention Management and Interoperate with FileNet P8 Platform screen

Configure the application group to work with the following features:

- Enhanced Retention Management feature of Content Manager OnDemand
- Interoperability between Content Manager OnDemand and FileNet® P8

Full text search screen

Specify the name of the Content Manager OnDemand full text indexing server, if installed, and, optionally, add Full Text Index folder fields.

Name screen

Specify the names of the application group, application, and folder. After you enter the names, Content Manager OnDemand queries the library server to make sure that the names are valid and unique.

Wizard complete screen

Confirms the selections that you made for the report. Click Display to view a summary of the application group, application, and folder definitions. From the summary window, choose the Print icon from the toolbar to print a copy of the definitions.

Note: When you are satisfied with the selections that you made for the report, click Finish to complete defining the report. Content Manager OnDemand adds the application group, application, and folder to the library server, closes the Report Wizard, and returns to the administrator window.

Chapter 6. Administration concepts

Introduction

You can use the administrative client to maintain a variety of Content Manager OnDemand objects such as users, groups, printers, storage sets, storage nodes, application groups, applications, folders, cabinets, and holds.

An object cannot be updated by multiple users at the same time. If the same object is updated by multiple users simultaneously, changes might be overridden, and invalid data might be created in the database.

Users

When you define a Content Manager OnDemand user, you create a userid with which a person in your organization logs on to the Content Manager OnDemand library server. You can optionally add the userid to folders and application groups, which is one way to let the user open folders and access data.

Each person in your organization logs on to the library server using a Content Manager OnDemand userid. Content Manager OnDemand authenticates the userid and determines the usage and administrative authority available to that person, based on the userid.

Content Manager OnDemand can store a maximum of 1,080,000 separate user IDs.

A Content Manager OnDemand userid does not necessarily have to identify an individual by name. However, for accounting purposes, most customers assign a Content Manager OnDemand userid to each person that will use the system. When you initialize the system, Content Manager OnDemand automatically creates the ADMIN userid. The ADMIN userid has system administrator authority. A system administrator can do the basic user functions, such as logging on the system and opening folders. A system administrator can also do administrative functions, such as maintaining users and groups and creating, updating, and deleting application groups, applications, folders, storage sets, printers, holds, and cabinets.

Note: The ADMIN userid does not have an initial password assigned. Because the ADMIN userid has system authority, IBM recommends that you assign a password to the ADMIN userid immediately after installation.

When naming Content Manager OnDemand users, the name that you specify:

- Can contain one to 128 characters (bytes)
- Cannot include the ' (apostrophe), *, %, +, [(left bracket),] (right bracket), " (double quotation mark), or blank character
- Must be unique to the library server

You can specify a userid in mixed case. By default, Content Manager OnDemand ignores the case (for example, LaGuarde is the same as lAguarde). Content Manager OnDemand converts lowercase letters in a user name to uppercase (LaGuarde is stored as LAGUARDE). However, depending on how you configure the Login Processing system parameters, userid processing on your system may be different (the case may be significant). See [“System parameters” on page 19](#) for more information.

User types

When you add a user to Content Manager OnDemand, you specify the *User Type*. The User Type and the Authority determines the types of tasks that the user can do when logged on to the system. You can choose from the following User Types:

User

Users can log on to Content Manager OnDemand, open folders that they are authorized to access, and search for and retrieve data from application groups that they are authorized to access. Users can be given authority to do other things on the system.

User Administrator

A user that can also add, update, and delete users and user administrators. A user administrator can be given authority to do other things on the system.

Application Group/Folder/Cabinet Administrator

A user that can also add, update, and delete application groups, applications, folders, and cabinets. An application group/folder/cabinet administrator is automatically given Logical Views permission to all application groups. An application group/folder/cabinet administrator can be given authority to do other things on the system.

Hold Administrator

A user that can add, update, delete, or view hold objects.

System Administrator

A user that can also add, update, and delete any user, group, application group, application, folder, holds, or cabinet on the system. A system administrator is automatically given Logical Views permission to all application groups. A system administrator is also the only user that can maintain storage sets and printers.

Note: When adding or updating a user, you are not permitted to set the User Type or Authority to a level that exceeds your own. For example, a user with Create Users and Create Groups authority cannot create a user with Create Folders authority.

By default, only the user, the user that created the user, user administrators, and system administrators can view or maintain the user. See the User Permissions page for more information

Only a system administrator, an application group/folder/cabinet administrator, a user with administrator authority for an application group, or a user with add document permission can store data in an application group.

Only a system administrator, an application group/folder/cabinet administrator, or a user with delete document permission can delete data from an application group.

Authority

The Authority optionally lets the user do other things on the system. For example, A User Type of User can be permitted to create users. The authority options that you can select depend on the User Type. For example, if the User Type is Application Group/Folder/Cabinet Administrator, then by definition, the user can create application groups and folders. Therefore, the only additional authorities that the user can be given are Create Users and Create Groups. Choose from the following:

Create Users

An optional authority for users and application group/folder/cabinet administrators:

- If the User Type is User, lets the user create users with a User Type of User.
- If the User Type is Application Group/Folder/Cabinet Administrator, lets the user create users with a User Type of User or a User Type of Application Group/Folder/Cabinet Administrator.

Note: Users with Create Users authority can maintain the users that they create, as long as they remain an administrator of the user.

Create Groups

An optional authority for users, user administrators, and application group/folder/cabinet administrators. Lets the user create groups. Users with Create Groups authority can maintain the groups they create, as long as they remain a group owner.

Create Application Groups

An optional authority for users and user administrators. Lets the user create application groups. Users with Create Application Groups authority can maintain the application groups that they create, as long as they remain an administrator of the application groups.

Create Folders

An optional authority for users and user administrators. Lets the user create folders. Users with Create Folders authority can maintain the folders that they create, as long as they remain an administrator of the folders.

Create Cabinets

An optional authority for users and user administrators. Lets the user create cabinets. Users with Create Cabinets authority can maintain the cabinets that they create, as long as they remain an administrator of the cabinets.

Create Holds

An optional authority for users and user administrators. Lets the user create holds. Users with Create Holds authority can maintain the holds they create, as long as they remain an administrator of the holds.

Note: When adding or updating a user, you are not permitted to set the User Type or Authority to a level that exceeds your own. For example, a user with Create Users and Create Groups authority cannot create a user with Create Folders authority.

Groups

Content Manager OnDemand groups are a means to organize users of the system by function, authorization, or any other purpose that you might require. You do not have to assign a user to a group, however doing so can simplify administration of users with similar requirements and capabilities.

When you define a group, you can add users to the group and specify folder and application group permissions that are common to all of the users that belong to the group. The permissions determine the types of actions users assigned to the group can perform on the system.

When naming groups, the name that you specify:

- Can contain one to 128 characters (bytes)
- Cannot include the ' (apostrophe), * (asterisk), % (percent), + (plus), [(left bracket),] (right bracket), " (double quotation mark), or blank character
- Can be mixed case. However, the case does not create a unique name (LaGuaRde is the same as laguearde)
- Must be unique to the library server

You can assign a 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, folders, cabinets, or be an application group/folder/cabinet administrator or a system administrator. If you do not assign a group owner, only a system administrator user can maintain the group.

Printers

A Content Manager OnDemand printer is an interface between the user and a print device that is controlled by a server.

The Content Manager OnDemand server print function is the highest performance print option that is available to end-users of the system. The server print function is designed to allow many documents to be selected for reprint from the document list, without retrieving the documents to the user's PC. The Content Manager OnDemand server print function supports printing by using the Direct Server printing method, to submit a print job to a JES output class or by using the Infoprint Server printing method, to submit a print job to an IBM Infoprint Server for z/OS print queue.

When printing AFP documents to server printers, the resource group that was archived at the time that the document was loaded into the system is put into the print data stream, to make sure that the document prints with the correct resources.

See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for information about configuring server printing on the Content Manager OnDemand library server.

Storage sets and storage nodes

A storage set is a named collection of primary storage nodes that support application groups with similar storage management requirements, such as the length of time that files are maintained in the application group and the type of media on which the files are stored.

Most customers define one or two storage sets for a Content Manager OnDemand system, based on the storage requirements of the reports that they plan to store in the system. For example, you might define one storage set to maintain reports for a period of ten years and another storage set to maintain reports for a period of 180 days.

A storage set can contain one or more primary storage nodes. A primary storage node identifies an object server on which Content Manager OnDemand copies data to cache storage. A storage set can write data to one and only one primary storage node at a time.

When you define an application group, you assign the application group to a storage set. The storage set must support the storage requirements of the data that you plan to store in the application group.

Application groups

An application group represents the data that you store in Content Manager OnDemand and the documents that you query, view, print, and FAX using Content Manager OnDemand client programs. For example, the data can be reports generated by a z/OS application program, index data generated by ACIF, and annotations created by users.

When you define an application group, you specify properties of the application group, such as the organization of the database and the storage characteristics for the files that are to be stored in the application group. You also define the database fields that will hold index data extracted from the reports that you store in the application group.

Content Manager OnDemand extracts index data from the reports that you load into an application group and places the data in the database fields that you define. Content Manager OnDemand uses the index values to identify the documents that meet the search criteria entered by a user.

When you define an application group, you can also select the types of application group messages that Content Manager OnDemand saves in the system log.

When you define an application group, you specify permissions that let users access and maintain the application group and application group data. You can identify the groups and users that can access data stored in the application group with Content Manager OnDemand client programs. You can specify the types of functions that users can perform, such as viewing, printing, and annotating reports. You can assign administrator authority to a user or a group. Administrator authority allows a user to update the application group, for example, to authorize other users to access data stored in the application group.

Content Manager OnDemand organizes information about an application group into sections: General, Message Logging, Storage Management, Permissions, Field Definition, Field Information, and Advanced Index Information. Each section contains options, fields that you can use to enter information about the application group, and command buttons.

Applications

You typically define an application for each different type of report or source of data that you plan to store in Content Manager OnDemand.

When you define an application, you assign the application to an application group and specify the physical and logical characteristics of the report. The physical characteristics of a report include the code page, the type of data found in the input file, and information about carriage control characters. The logical characteristics of a report include the different ways that you want to present the information contained in the report to your users.

The Content Manager OnDemand data indexing, loading, and viewing programs process the report by using the information that you specify to the application. For example, you can specify the parameters that the Content Manager OnDemand indexing program uses to locate and extract index data from the

report. You can create logical views for the application. Logical views represent different ways to display pages of the report. You can set up printing options, such as defining a default printer for users and printing options for AFP and line data documents.

Content Manager OnDemand organizes information about an application into sections: General, View Information, Indexer Information, Load Information, Logical View Fields, Logical Views, and Miscellaneous Options. Each section contains options, fields that you can use to enter information about the application, and command buttons.

Folders

A folder provides users with the means to access the reports that you store in Content Manager OnDemand. A user opens a folder to construct a query and search for documents in one or more of the application groups that can be accessed from the folder. The user can also view, print, annotate, and e-mail documents from the folder.

When you define a folder, you specify the properties of the folder, such as the name and description of the folder, create the search and display fields that appear when the user opens the folder, and map the folder fields to application group database fields.

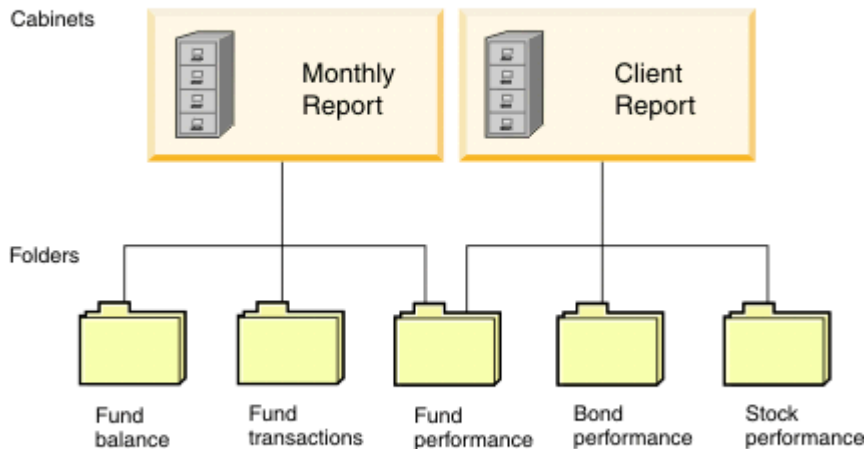
You can also specify the groups and users that can open the folder with Content Manager OnDemand client programs and other folder permissions. For example, you can authorize a user to be the folder administrator. The folder administrator can authorize other users to open the folder and make changes to the folder fields.

Content Manager OnDemand organizes information about a folder into sections: General, Permissions, Field Definition, Field Information, and Field Mapping. Each section contains options, fields that you can use to enter information about the folder, and command buttons.

Cabinets

A cabinet is a container for folders. You can use cabinets to manage folders and enable users to navigate to folders more easily. A folder can belong to one or more cabinets. The following diagram describes the relationship between two cabinets and five folders. The **Monthly Report** cabinet contains three folders: **Fund balance**, **Fund transactions**, and **Fund performance**. The **Client Report** cabinet also contains three folders: **Fund performance**, **Bond performance**, and **Stock performance**. The two cabinets share the **Fund performance** folder. In this example, you can use the **Client Report** cabinet to quickly retrieve information about a client's portfolio, and use the **Monthly Report** cabinet to obtain fund information for internal analysis.

Optional: You can organize folders in cabinets to enable users to navigate to folders more easily. Each cabinet is a collection of folders.



A folder can belong to more than one cabinet.

Figure 3: Cabinets and folders

Holds

In Content Manager OnDemand, documents are stored and retained for a specified period of time. After that specified period of time, the documents are removed from the system. In certain situations, you might need to keep one or more documents beyond the expiration date by using hold objects. You can put documents on hold in several ways:

- Placing a hold on a single document
- Placing a hold on a load of documents

You can also add one or more documents to existing hold objects.

About application groups, applications, and folders

Before you can load a report into the system, you must create an application group and an application. Before users can search for and retrieve data, you must create a folder.

- Users open a folder to search for the reports that you load into the system. You define one or more search fields for the folder. A folder search field is mapped to an application group database field.
- Each database field that you define represents a category of information in the report, such as a customer name, invoice number, or balance. When you add an application group, Content Manager OnDemand creates a database table. The database fields that you define are columns in the table.
- You define an application for each report that you plan to store in Content Manager OnDemand. When you add an application to the system, you define information for the Content Manager OnDemand viewing, indexing, and loading programs. When you define indexing information, you identify the name, location, and length of each category of index information that you want to extract from the report. When you define loading information, you map index fields in the report to application group database fields.
- When you load a report into the system, Content Manager OnDemand stores the index values that are extracted from the report into database fields in records that are added to an application group table.
- Content Manager OnDemand uses the index values to identify the documents that meet the search criteria that the user entered into the folder search fields.

When you want to define a report to Content Manager OnDemand, your first task is to identify the application group from which Content Manager OnDemand obtains information about the index fields and how documents are to be maintained on the system. When you define an application group, you specify how you want Content Manager OnDemand to structure information in the database and define the database fields. When you define an application group, you also specify how you want Content Manager OnDemand to maintain data on the system. For example, you might specify that report data should be maintained in cache storage for 60 days and in archive storage for five years. Content Manager OnDemand maintains all of the data stored in the application group the same way. Content Manager OnDemand maintains each report that you store in the application group for the same length of time.

- You can store the report in an existing application group. However, you must be able to index the report using the database fields that are already defined in the application group. The storage management information for the application group must support the length of time that you want Content Manager OnDemand to maintain the report on the system and how and where that you want Content Manager OnDemand to store and maintain the report data.

You can verify information about an existing application group with the Properties command. The General page shows the database organization for the application group. The Storage Management page shows the data caching and migration information. The Field Definition page shows the application group database fields.

- If there are no application groups defined to Content Manager OnDemand or there are no application groups that support the database and storage management requirements of the report, then you must add an application group to the system.

After you add an application group, you must define an application for the report. Most customers create a Content Manager OnDemand application for each different type of report or source of data that they plan to store in Content Manager OnDemand. When you create an application, you must assign it to an application group. The application group determines where Content Manager OnDemand will store the report data. When you create an application, you also specify information that the Content Manager OnDemand client programs use to view and print pages of the report and you specify instructions for the data indexing and loading programs.

The last step in the process of adding a report to Content Manager OnDemand is to create a folder. Users open the folder to search for, display, and print reports. When you define a folder, you select the application group that contains the data that you want users to search when they open the folder. By creating folders that can search specific application groups, you can determine the reports that are available to users when they open a folder.

When you define a folder, you define search and display fields. You specify characteristics of the folder fields, such as default search operators and whether Content Manager OnDemand displays default search values for the fields when a user opens the folder. You also map the folder fields to application group database fields.

Content Manager OnDemand permissions

Permissions are the means by which Content Manager OnDemand determines who can open folders and search for documents stored in application groups. Content Manager OnDemand also uses permissions to determine who can maintain folders, application groups, and other objects with the administrative client.

By default, only the person that adds the folder, an application group/folder/cabinet administrator, or a system administrator can open and maintain the folder. By default, only the person that adds the application group, an application group/folder/cabinet administrator, or a system administrator can access data stored in the application group or maintain the application group.

Content Manager OnDemand provides several ways for you to specify permissions. You can specify permissions when you add or update a folder or an application group. You can also add, remove, or update a user's or group's folder or application group permissions when you add or update the user or group.

As both a convenience and security measure, you can assign a user to a group. Content Manager OnDemand groups allow you to organize users by function, authorization, or any other logical grouping

that you might require. When you assign a user to a group, the user obtains the permissions that are in effect for the group. For example, suppose that you create a group and authorize the group to open the Student Information folder. Any user that you assign to the group automatically obtains permission to open the Student Information folder. If you assign a user to more than one group, the user normally obtains the permissions of all of the groups. However, there are exceptions. See [“Specifying permissions” on page 37](#) for details.

You can specify a default set of permissions for folders and application groups. Content Manager OnDemand uses the default permissions when users and groups do not have specific permissions for the folder or application group. If you specify permissions for a group, then the group permissions take precedence over the default permissions. If you specify permissions for a user, then the user permissions take precedence, regardless of any group that the user may belong to or the default permissions that you specified.

Folder permissions

You can specify default folder permissions and folder permissions for specific groups and users. The default permissions provide every user and group defined to the library server with the permissions that you specify. Permissions for a group provide the users that you add to the group with the permissions that you specify for the group. Permissions for a user provide the user with the permissions that you specify. By default, only the person that adds the folder, an application group/folder/cabinet administrator, or a system administrator can open the folder or maintain the folder.

You can specify the following types of folder permissions:

Authority

Determines the types of folder functions that users can perform. For example, Access lets users open the folder with Content Manager OnDemand client programs. To search for and retrieve data from the application groups referenced by the folder, the user must be given access permission to the application groups.

Named Queries

A named query is a set of search criteria, saved by name, that can be selected and restored into folder search fields. Content Manager OnDemand supports two types of named queries: public and private. A public named query is available to any user that can access the folder. A private named query is available to the user that created the named query. A system administrator or application group/folder/cabinet administrator can also access private named queries. A user can be given authority to view, create, modify, and delete named queries.

User/Group Fields

Lets users open the folder with Content Manager OnDemand client programs and modify the folder field information with the administrative client.

Maximum Hits

Determines the maximum number of documents that Content Manager OnDemand displays in the document list, regardless of the number that match the query. By default, Content Manager OnDemand lists all of the documents that match the query. The Content Manager OnDemand CICS client program lists a maximum of 200 documents that match a query, regardless of the number that match a query or the value that you specify.

Secondary Folder

Use to mark the folder as a secondary folder. When a user logs on to a server, the client lists the primary folders that a user can open. A user can list all of the folders that they can open, including those marked as secondary, by choosing the All Folders option from the Open a Folder dialog box.

Full Report Browse

Use to allow a user of the Content Manager OnDemand client to select a document, retrieve that document, and view the entire report (load) to which the document belongs.

Application group permissions

You can specify default application group permissions and application group permissions for specific groups and users. The default permissions provide every user and group defined to the library server with

the permissions that you specify. Permissions for a group provide the users that you add to the group with the permissions that you specify for the group. Permissions for a user provide the user with the permissions that you specify. By default, only the person that adds the application group, an application group/folder/cabinet administrator, or a system administrator can access data stored in the application group or maintain the application group.

You can specify the following types of application group permissions:

Authority

Determines the types of application group functions that users can perform. For example, Access lets users search for and retrieve data stored in the application group using Content Manager OnDemand client programs.

Document

For users that can access the application group, determines the types of document functions that users can perform.

Annotation

For users that can access the application group, determines the types of annotation functions that users can perform.

Query Restriction

Limits access to application group data. You typically set up a query restriction to limit the data that a specific user or group of users can access.

Specifying permissions

About this task

To ease the administration of Content Manager OnDemand, most customers organize their users into groups, add the groups to folders and application groups, and specify permissions for the groups. You should plan your groups before you begin creating them. After you start using the system, you may find it difficult to change the organization of your groups.

When you add a user to a group, the user automatically obtains the permissions that were specified for the group. When you add a user to more than one group, the user normally obtains the permissions of all of the groups. For example, using the group properties listed in [Table 2 on page 37](#), a user that belongs to both groups can open the Student Bills and Student Transcripts folders.

Table 2: Group permissions

Group	GID	Folders	Permission
Admissions	1080100	Student Transcripts	Access
Accounting	1080101	Student Bills	Access

Most situations involve adding a group to a folder, specifying permissions for the group, and then adding users to the group. However, there may be situations when you need to deny a group of users access to a folder. When you use groups to deny access to a folder, you must understand how Content Manager OnDemand determines folder permissions for a group (and users assigned to the group). For example, consider the group properties listed in [Table 3 on page 37](#).

Table 3: Group permissions

Group	GID	Folders	Permission
Admissions	1080100	Student Bills	None
Admissions	1080100	Student Transcripts	Access
Accounting	1080101	Student Bills	Access
Accounting	1080101	Student Transcripts	Access

A user that belongs to both groups can not open the Student Bills folder.

When a user belongs to more than one group, Content Manager OnDemand uses the group identifier (GID) to determine the user's permissions. When two (or more) groups provide permissions for the same folder, the user obtains the permissions of the group with the lowest GID. In the example depicted in [Table 3 on page 37](#), both groups have been added to the Student Bills folder. Since the Admissions group has a lower GID than the Accounting group, Content Manager OnDemand uses the permissions specified for the Admissions group to determine the permissions of a user that is assigned to both groups. Consequently, users assigned to both groups cannot access the Student Bills folder.

You're probably asking yourself, "why would I assign a user to more than one group" or "why would I create a group with no access to a folder?" Perhaps some examples will help answer these questions (and clarify the Content Manager OnDemand permission hierarchy). As you review the examples, please remember the following rules:

- By default, the person that created the folder, a system administrator, and an application group/folder/cabinet administrator can access the folder
- You can use the *PUBLIC name to specify default permissions for all other users
- You can specify permissions for specific groups and users:
 - All of the users that belong to a group that you add to a folder will obtain the permissions that you specify for the group
 - A user that belongs to two (or more) groups that have been added to the same folder will obtain the permissions of the group that has the lowest GID
 - The permissions that you specify for a user override all other permissions, including any default permissions (*PUBLIC) and any groups to which the user belongs and that are added to the folder

Examples

The examples that follow show how to add groups to folders and specify folder permissions. The same considerations hold true for adding groups to application groups and specifying application group permissions.

Providing a group of users access to a folder

Let's say that you want to provide a single group of users access to a folder. Complete the following steps:

1. With *PUBLIC selected, clear all of the permissions check boxes (this is the default).
2. Add the group to the folder.
3. Select the Access check box.

Users assigned to the group automatically obtain permission to open the folder.

Denying a group of users access to a folder

Let's say that you want to prohibit a single group of users from accessing a folder, while allowing all other users defined to the library server to open the folder. Complete the following steps:

1. With *PUBLIC selected, select the Access check box (this lets all users open the folder).
2. Add the group to the folder.
3. Clear all of the permissions check boxes.

Users assigned to the group cannot open the folder.

Providing one group of users access and denying another group of users access to the same folder

Let's say that you want to allow a group of users to access a folder. However, you need to prohibit certain users in the group from accessing the folder. You could exclude the users from the group that can access the folder. However, there may be other folders that you want the users to access as part of the group. To

solve this, create two groups, one without access to the folder and the other with access to the folder, and assign the users to the respective groups. For example:

1. Create the "no access" group. This group must have a lower GID than the "access" group. Add users to the group.
2. Create the "access" group. Add users to the group.
3. With *PUBLIC selected, clear all of the permissions check boxes (this is the default).
4. Add the "access" group to the folder.
5. Select the Access check box.
6. Add the "no access" group to the folder.
7. Clear all of the permissions check boxes.

If you later need to deny other users access to the folder, simply add the users to the "no access" group. You can also move users from one group to the other.

Denying one user in a group access to a folder

Assume that you want to prohibit one user in a group from accessing a folder. After adding the group to the folder and specifying the access permission, all users assigned to the group can open the folder. To override the group permissions, add an individual user to the folder and set permissions at the user level. Complete the following steps.

1. With *PUBLIC selected, clear all of the permissions check boxes (this is the default).
2. Add the group to the folder.
3. Select the Access check box.
4. Add the user to the folder.
5. Clear all of the permissions check boxes.

Even though the user belongs to the group, the user cannot open the folder.

Providing one user in a group administrator authority

Let's say that you want to provide one user in a group the ability to administer the folder. Complete the following steps.

1. With *PUBLIC selected, clear all of the permissions check boxes (this is the default).
2. Add the group to the folder.
3. Select the Access check box.
4. Add the user to the folder.
5. Select the Administrator check box.

Only the user that you added can administer the folder; the other users in the group can open the folder.

Specifying default permissions

The default permissions that you specify for an application group or a folder will apply to every user or group defined to the library server who is not provided with specific permissions.

For example, suppose that you specify Access as the default permission for an application group. Every user and group that is not provided with specific permissions can access the data that is stored in the application group. Then, you specify Access as the default permission for a folder. Every user and group that is not provided with specific permissions can open the folder. Later, you add a user, without specifying application group or folder permissions. The user can open the folder and access the data stored in the application group.

While default permissions do provide flexibility to maintain your system, you must make sure that using the default permissions does not circumvent your security strategy. Rather than specifying default permissions for application groups and folders, you may want to use groups as a means to implement

your security strategy. For example, you can clear all of the permissions under *PUBLIC and then add groups to a folder and specify the appropriate permissions for each group. When you add a user to the system, you can assign the user to a group. The user automatically obtains the permissions of the group. If the group does not have access to a particular application group or folder, then neither does the user. With this strategy, until you assign the user to a group, or provide the user with specific permissions, the user cannot access the folder.

Chapter 7. Advanced system administration

Overview

Applications, application groups, folders, printers, storage sets, and cabinets are the objects that represent how Content Manager OnDemand stores, manages, prints, retrieves, and views reports and index data. Controlling and limiting access to the reports and index data is accomplished by defining users and groups and giving them the level of authority that is required to meet the data security strategy of an organization.

User types

Content Manager OnDemand provides the ability to centralize or decentralize the administration of the system. Content Manager OnDemand also provides the flexibility to control access to objects from different levels. The most basic level of control is how the user is defined to the system. When a user is added, a user type is specified. Each user type has a different level of authority:

System Administrator

A system administrator has the highest level of authority on the system. A system administrator can perform all tasks on all of the objects that are defined to a Content Manager OnDemand system. The objects are users, groups, applications, application groups, folders, printers, storage sets, and cabinets. The tasks are add, update, delete, copy, export, create a summary, and view properties. A system administrator also has the authority to modify the system parameters.

System Administrator (Read-Only)

A read-only system administrator can log on to the OnDemand Administrator client and the ARSXML batch administration program with read-only access to all Content Manager OnDemand objects. In the OnDemand Administrator client, objects can be viewed and summarized. Add, Update, Delete, Copy, and Export options are not available; Trace Parameters and System Parameters can be viewed but not updated. With ARSXML, all objects can be exported, including Trace Parameters and System Parameters.

Application Group/Folder/Cabinet administrator

An application group/folder/cabinet administrator has the authority to perform all tasks on all of the applications, application groups, folders, and cabinets that are defined to a Content Manager OnDemand system. The tasks are add, update, delete, copy, export, create a summary, and view properties.

Hold Administrator

A user that can add, update, delete, or view hold objects.

User Administrator

A user administrator has the authority to perform all tasks on all of the users that are defined to a Content Manager OnDemand system. The tasks are add, update, delete, copy, export, create a summary, and view properties. **Note:** A user administrator cannot create or delete a system administrator or an application group/folder/cabinet administrator or change the user type of a system administrator or an application group/folder/cabinet administrator.

User

A user has the lowest level of authority on the system. A user does not have access to any object on the system and therefore, cannot perform any tasks. The user must be given the authority to access an object and to perform a task on the object.

Authority

A user's authority can be extended beyond the authority that is built into the user type of the user. Depending on the user type, up to six additional levels of authority can be given:

Create Users

An application group/folder/cabinet administrator or a user can be given the authority to add users to the system. By default, when a user is added to the system, the user that performs the add task automatically has the authority to perform all other tasks on the user. The tasks are update, delete, create a summary, and view properties. When a user is added, the system automatically gives the user with Create Users authority the permission to access and administer the newly created user. (See [“Permissions” on page 44](#) for information about permissions at the user level.) If the permissions are later taken away, then the user with Create Users authority no longer has access to the user or the authority to administer the user.

A user with Create Users authority is similar to a user administrator in that both can create users. However, they differ because a user with Create Users authority can access and administer only the users that they create, so long as their access and administrator authority is not taken away. **Note:** A user with Create Users authority cannot create or delete a system administrator, an application group/folder/cabinet administrator, or a user administrator or change the user type of a system administrator, an application group/folder/cabinet administrator, or a user administrator.

A user with Create Users authority also has the authority to perform copy and export tasks on a user because both tasks add a user. An export task involves two userids: one on the server where the user exists and one on the server where the user will be added. The userid on the server where the user will be added must have the authority to add a user.

Create Groups

A user administrator, an application group/folder/cabinet administrator, or a user can be given the authority to add groups to the system. The only other type of user that can add groups is a system administrator

By default, when a group is added to the system, the user with Create Groups authority is designated as the group owner. The group owner has the authority to perform all other tasks on the group. The tasks are update, delete, create a summary, and view properties. If the owner is changed to a different user or group, then the system automatically takes the authority to perform tasks on the group away from the user that originally created the group.

In general, a user has access to a group if the user is a system administrator, the owner of the group, a member of the group that has been designated as the owner, or a member of the group.

A user with Create Groups authority also has the authority to perform copy and export tasks on a group because both tasks add a group. An export task involves two userids: one on the server where the group exists and one on the server where the group will be added. The userid on the server where the group will be added must have the authority to add a group.

Create Application Groups

A user administrator or a user can be given the authority to add application groups to the system. By default, when an application group is added to the system, the user that performs the add task automatically has the authority to perform all other tasks on the application group. The tasks are update, delete, create a summary, and view properties. When the application group is added, the system automatically gives the user with Create Application Groups authority the permission to access and administer the newly created application group. (See [“Permissions” on page 44](#) for information about permissions at the application group level.) If the permissions are later taken away, then the user with Create Application Groups authority no longer has access to the application group or the authority to administer the application group.

A user with Create Application Groups authority is similar to an application group/folder/cabinet administrator in that both can create application groups. However, they differ because a user with Create Application Groups authority can access and administer only those application groups that they create, so long as their access and administrator authority is not taken away. An application

group/folder/cabinet administrator can access and administer all of the application groups that are defined to the system.

Because applications are considered by Content Manager OnDemand to be part of an application group, the permissions for accessing and administering applications are defined by the permission for the application group. For this reason too, a user with Create Application Groups authority can also create applications for the application group.

A user with Create Application Groups authority also has the authority to perform copy and export tasks on an application group because both tasks add an application group. An export operation involves two userids: one on the server where the application group exists and one on the server where the application group will be added. The userid on the server where the application group will be added must have the authority to add an application group.

Create Folders

A user administrator or a user can be given the authority to add folders to the system. By default, when a folder is added to the system, the user that performs the add task automatically has the authority to perform all of the other tasks on the folder. The tasks are update, delete, create a summary, and view properties. When a folder is added, the system automatically gives the user with Create Folders authority the permission to access and administer the newly created folder. (See [“Permissions” on page 44](#) for information about permissions at the folder level.) If the permissions are later taken away, then the user with Create Folders authority no longer has access to the folder or the authority to administer the folder.

A user with Create Folders authority is similar to an application group/folder/cabinet administrator in that both can create folders. However, they differ because a user with Create Folders authority can access and administer only the folders that they create, so long as their access and administrator authority is not taken away. An application group/folder/cabinet administrator can access and administer all of the folders that are defined to the system.

A user with Create Folders authority also has the authority to perform copy and export tasks on a folder because both tasks add a folder. An export task involves two userids: one on the server where the folder exists and one on the server where the folder will be added. The userid on the server where the folder will be added must have the authority to add a folder.

Create Cabinets

A user administrator or a user can be given the authority to add cabinets to the system. By default, when a cabinet is added to the system, the user that performs the add task automatically has the authority to perform all of the other tasks on the cabinet. The tasks are update, delete, create a summary, and view properties. When a cabinet is added, the system automatically gives the user with Create Cabinets authority the permission to access and administer the newly created cabinet. (See [“Permissions” on page 44](#) for information about permissions at the cabinet level.) If the permissions are later taken away, then the user with Create Cabinets authority no longer has access to the cabinet or the authority to administer the cabinet.

A user with Create Cabinets authority is similar to an application group/cabinet administrator in that both can create cabinets. However, they differ because a user with Create Cabinets authority can access and administer only the cabinets that they create, so long as their access and administrator authority is not taken away. An application group/cabinet administrator can access and administer all of the cabinets that are defined to the system.

A user with Create Cabinets authority also has the authority to perform copy and export tasks on a cabinet because both tasks add a cabinet. An export task involves two user IDs: one on the server where the cabinet exists and one on the server where the cabinet will be added. The user ID on the server where the cabinet will be added must have the authority to add a cabinet.

Create Holds

A user administrator or a user can be given the authority to add holds to the system. By default, when a hold is added to the system, the user that performs the add task automatically has the authority to perform all of the other tasks on the hold. The tasks are update, delete, create a summary, and view properties. When a hold is added, the system automatically gives the user with Create Holds authority the permission to access and administer the newly created hold. (See [“Permissions” on page 44](#) for

information about permissions at the hold level.) If the permissions are later taken away, then the user with Create Holds authority no longer has access to the hold or the authority to administer the hold.

A user with Create Holds authority also has the authority to perform copy and export tasks on a hold because both tasks add a hold. An export task involves two user IDs: one on the server where the hold exists and one on the server where the hold will be added. The user ID on the server where the hold will be added must have the authority to add a hold.

Permissions

Access and control of users, groups, applications, application groups, folders, printers, and storage sets can be given at various levels. In [“User types” on page 41](#), the level of control was determined by the user type of the user. In this section, a different level of control is described. Permissions on an object can be set from the Permissions page of the object. The following topics describe the objects and the permissions that can be set.

Users

A user can be given the authority to view documents that have been archived. A user can also be given the authority to perform administrative tasks on the system. A user can retrieve a list of users from the server with the administrative client. The list contains the users that the user has the authority to access. If the user is a system administrator or a user administrator, then the user also has the authority to administer any user in the list.

Access authority means the user can see the user in any list that contains users and can print or view the properties of the user. Access authority can be given to an individual user or to a group.

Access authority is especially helpful to application group/folder/cabinet administrators, because they can give any user in the list access to application groups and folders. This is also true for group owners; access authority allows them to add users to the groups that they own.

Administrator authority of a user can be given to another user or to a group. Having administrator authority for a user means that the user or group given the authority can delete or update the user.

Groups

Groups can be created by a system administrator or a user that has Create Groups authority. Updating or deleting a group can be performed by a system administrator or the group owner. A group owner can be a user or another group. Allowing the group owner to be another group provides the ability to allow multiple users to administer the group.

To see a group in a list, a user must be a member of the group, the owner of the group, or a system administrator. For example, an application group/folder/cabinet administrator can give groups access to application groups and folders. However, the application group/folder/cabinet administrator must be able to see the groups in the list on the Permissions page of the application group, folder, or cabinet. This means that the application group/folder/cabinet administrator must be a member of any group that requires access authority to an application group or a folder.

A system administrator, a group owner, or a member of the group can view the properties of the group and create a summary.

Applications

Because applications are considered by Content Manager OnDemand to be part of an application group, the permissions for accessing and administering applications are defined by the permission for the application group. (See [“Application Groups” on page 45](#).)

Application Groups

Application Groups can be created by system administrators, application group/folder/cabinet administrators, and users with Create Application Groups authority. After an application group is created, only a system administrator, an application group/folder/cabinet administrator, a user with administrator authority for the application group, or a member of a group that has administrator authority for the application group can update or delete the application group. When a user with Create Application Groups authority creates an application group, the system automatically gives the user the authority to administer the application group. The user can update or delete the application group, so long as the administrator authority is not taken away.

To allow other users to see an application group in a list with the administrative client or to search for documents with the end-user client, the users must be given access authority to the application group. A user with access authority can also print or view the properties of the application group. Access is given on the Permissions page of an application group. There are three ways to give a user access to an application group:

1. Add the user's userid to the access list.
2. Add the name of a group to which the user belongs to the access list. The user and all of the other members of the group will have access to the application group.
3. Set the access permission for the reserved name *PUBLIC. All users on the system will have access to the application group. (The *PUBLIC name is used to set permissions for all users on the system.)

The levels of authority within the application group have a precedence order in which the permissions are enforced. The permissions that have been set for a user take precedence over any permissions that have been set for any groups that the user may belong to. User permissions also take precedence over permissions that have been set using the *PUBLIC name.

A user can also be given the authority to save a specific set of viewing attributes such as zoom, background color, and so forth. The viewing attributes can be used when a document is viewed with the end-user client. The set of viewing attributes, called a logical view, is accessible only to the user that created the logical view.

Document and Annotation permissions can also be set for users by using the *PUBLIC name, group names, and userids. Document permissions include add, delete, update, view, copy, print, and FAX. Annotation permissions include add, delete, update, view, and copy.

Adding fields to an existing application group

To add fields to an application group, do the following tasks:

1. Identify the tables that will be affected by the addition of the field or fields. You will need to know the names of the tables to be able to run the REORG utilities at the end of these instructions.
2. Choose a time to make this change that follows these requirements:
 - The Content Manager OnDemand server is not loading data to the application group.
 - You have enough time to run the REORG utility against all the tables affected by this change.
3. Start the administrative client, then add the fields.
4. After you click on **OK** to save your changes, the Content Manager OnDemand server updates all the existing data tables and supplies the default values for the fields. After the Content Manager OnDemand server completes the update, the server returns control of the administrative client to you.
5. Run the REORG utility on the tables you identified in step 1. After the utility completes the update, it displays a message indicating success or failure.

Folders

Folders can be created by system administrators, application group/folder/cabinet administrators, and users with Create Folders authority. After a folder is created, only a system administrator, an application group/folder/cabinet administrator, a user with administrator authority for the folder, or a member of a group that has administrator authority for the folder can update or delete the folder. When a user with

Create Folders authority creates a folder, the system automatically gives the user the authority to administer the folder. The user can update or delete the folder, so long as the administrator authority is not taken away.

To allow other users to see a folder in a list with the administrative client or to open a folder with the end-user client, the users must be given access authority to the folder. A user with access authority can also print or view the properties of the folder. Access is given on the Permissions page of a folder. There are three ways to give a user access to a folder:

1. Add the user's userid to the access list.
2. Add the name of a group to which the user belongs to the access list. The user and all of the other members of the group will have access to the folder.
3. Set the access permission for the reserved name *PUBLIC. All users on the system will have access to the folder. (The *PUBLIC name is used to set permissions for all users on the system.)

The levels of authority within the folder have a precedence order in which the permissions are enforced. The permissions that have been set for a user take precedence over any permissions that have been set for any groups that the user may belong to. The user permissions also take precedence over permissions that have been set using the *PUBLIC name.

In addition to allowing a user to access or administer a folder, a user can also be given the authority to customize the appearance of the folder search and display fields with the administrative client. The authority can be given to a specific user or to a group. If the authority is given to a group, then any member of the group has the authority to customize the appearance of the search and display fields. Only the authorized user or members of the group see the customized search and display fields with the end-user client. All other users will see the search and display fields that have been defined using the *PUBLIC name in the folder.

A user can also be given the authority to save a specific set of search criteria when using the end-user client. The user can restore the set of search criteria when needed, into the search fields of a folder. The set of search criteria, called a named query, can be made available to all of the users that have access to the folder (a public named query) or it can be made available only to the user that created the named query (a private named query). When giving Named Query authority to a user, the user can be given the authority to view named queries, but not create them; the user can also be given the authority to create public named queries, private named queries, or both. A user can get Named Query authority from a group, if the group has been given Named Query authority for the folder.

Storage sets

Any user on the system can view the properties of a storage set and create a summary. However, only a system administrator can add, delete, update, copy, or export a storage set.

Printers

Printers are maintained by system administrators. Only a system administrator can add, delete, update, copy, or export a printer. By default, only system administrators can see printers in a list and therefore, only a system administrator can view the properties of a printer or create a summary. However, it is almost always necessary for other users to have access to printers. For this reason, a user or group can be given access to a particular printer. A printer's access list is maintained on the Permissions page under printers, by using the administrative client. Access to a printer can be given to all users and groups defined to the system, individual groups, and individual users.

Limiting access to printers provides the ability to control which printers can be used to print archived documents by Content Manager OnDemand users. For example, suppose that there is a printer in the customer service department. Only people in the customer service department should be permitted to print on the printer. You can accomplish this by creating a Content Manager OnDemand group that contains only the department members and giving only that group access to the printer.

Cabinets

You must have Create Cabinets authority, or be an application group/folder/cabinet administrator or system administrator to work with a cabinet.

If you have the appropriate permission, you can add, update, delete, copy, export, summarize, find, explore, or view the properties of a cabinet.

You can also give a user one of the following two types of authority:

Access

Access authority enables the user to see the cabinet in a list of cabinets and view the contents of the cabinet.

Administrator

Administrator authority enables the user to update and delete the cabinet in addition to the authority that is provided by Access.

Hold

You must have Create Holds authority, or be a hold administrator or system administrator to work with a hold.

If you have the appropriate permission, you can add, update, delete, copy, export, summarize, find, explore, or view the properties of a hold.

You can also give a user one of the following two types of authority:

Access

Access authority enables the user to see the hold in a list of holds and view the contents of the hold.

Administrator

Administrator authority enables the user to update and delete the hold in addition to the authority that is provided by Access.

System administration

Content Manager OnDemand provides the ability to centralize or decentralize the administration of the system. A centralized environment means that one type of user, a system administrator, controls the creation and access to all of the objects defined on the system. A decentralized environment means that the tasks of the system administrator are divided and assigned to other users. The responsibilities of the other users may vary from user administration, group administration, application group administration, folder administration, or any combination of the administrative tasks.

The decision to centralize or decentralize the administration of the system should be made before objects are added to the system. While the decision is reversible, the amount of work required to change from one type of administration to the other can be significant if a large number of users, groups, folders, and application groups have already been added.

There are many ways to decentralize the administration of the system, because of the various user types and the additional authority levels that can be specified for users. Two specific models will be discussed in this section: the Object Type model and the Object Owner model.

- In the Object Type model, all of the objects on the system are logically grouped into administrative domains according to the type of the object. The administrator of a domain maintains all of the objects within the domain. For example, an application group/folder/cabinet administrator maintains all of the application, application group, folder, and cabinet objects on the system.
- In the Object Owner model, the objects on the system are logically grouped into administrative domains according to the creator/owner of the object. An administrator maintains only the objects that they create. For example, a user with create application groups and create folders authority can maintain only the applications, application groups, and folders that they created. The Object Owner model can be used to separate the objects on the system into logical parts, such as a department, a company, or some other entity. Each part is independent of the other and should be maintained separately. Each part

typically requires two administrative users. One user has the responsibility for creating and maintaining users and groups. The other user has the responsibility for creating and maintaining applications, application groups, and folders. However, you can also define one user with the authority to create and maintain users, groups, applications, application groups, and folders. In effect, the one user would be the system administrator for a logical part of the system.

Object Type model

In the Object Type model, the system administrator defines two new users. One user is responsible for administering applications, application groups, and folders and is defined as an application group/folder/cabinet administrator. The second user is responsible for administering users and groups and is defined as a user administrator with Create Groups authority. [Table 4 on page 48](#) shows the administrative users and the tasks assigned to the users.

User Type	Tasks
System Administrator	<ul style="list-style-type: none"> Create an application group/folder/cabinet administrator Create a user administrator with Create Groups authority Create and maintain storage sets Create and maintain system printers Create and maintain cabinets
User Administrator with Create Groups authority	<ul style="list-style-type: none"> Create and maintain users Create and maintain groups
Application Group/Folder/Cabinet Administrator	<ul style="list-style-type: none"> Create and maintain application groups Create and maintain applications Create and maintain folders Create and maintain cabinets

When maintaining application groups and folders, the application group/folder/cabinet administrator must give other users access to the application groups, folders, and cabinets. The recommended and simplest way to do this task is to give access to a group, rather than to individual users. No additional work is required by the application group/folder/cabinet administrator when another user needs access to the application group, folder, or cabinet. When a new user is added to the group, the user automatically gets access to the application group, folder, or cabinet. Adding the user to the group is the responsibility of the user administrator since the user administrator owns all of the groups in this model.

Another reason for giving groups rather than individual users access to application groups and folders is that the application group/folder/cabinet administrator does not have access to the users and groups in this model. Because the application group/folder/cabinet administrator must first be given access to any users or groups that require access to application groups, folders, or cabinets, it is simpler and less time consuming to give access to a few groups rather than hundreds or even thousands of users. The application group/folder/cabinet administrator is given access to a group by adding the application group/folder/cabinet administrator to the group. This task is done by the user administrator with Create Groups authority. As a group member, the application group/folder/cabinet administrator will be able to see the group in the list and will therefore be able to give the group access to any application groups and folders on the system.

To give an application group/folder/cabinet administrator access to a user, the user administrator with Create Groups authority must update each user and give the application group/folder/cabinet administrator access to the user. Once access has been given, the application group/folder/cabinet administrator will be able to see the user in the list and will therefore be able to grant the user access to

any application groups, folders, and cabinets on the system. Again, this is not the recommended approach because this task will have to be repeated each time that a user is added to the system.

Object Owner model

In the Object Owner model, the system administrator defines two users for each logical part of the system. One user is responsible for maintaining the users and groups for a logical part of the system. The other user is responsible for maintaining the applications, application groups, folders, and cabinets for a logical part of the system. The Object Owner model allows you to store data from several sources on one Content Manager OnDemand system and let only one set of users access each set of data. [Table 5 on page 49](#) shows the administrative users and the tasks assigned to the users.

User Type	Tasks
System Administrator	<ul style="list-style-type: none"> Create a user with Create Users and Create Groups authority Create a user with Create Application Groups and Create Folders authority Create and maintain storage sets Create and maintain system printers
User with Create Users and Create Groups authority	<ul style="list-style-type: none"> Create and maintain users Create and maintain groups
User with Create Application Groups, Create Folders, and Create Cabinets authority	<ul style="list-style-type: none"> Create and maintain application groups Create and maintain applications Create and maintain folders Create and maintain cabinets

In addition to the tasks listed in [Table 5 on page 49](#) under System Administrator, it is also necessary for the system administrator to give the user with Create Users and Groups authority access to the user with Create Application Groups, Create Folders, and Create Cabinets authority. Otherwise, the user that creates groups will not be able to add the user that creates application groups, folders, and cabinets to any groups. To simplify the explanation that follows, the user with Create Users and Groups authority will be called the user administrator and the user with Create Application Groups and Folders authority will be called the application group/folder/cabinet administrator.

When maintaining application groups folders, and cabinets, the application group/folder/cabinet administrator must give access to application groups, folders, and cabinets to other users on the system. The recommended and simplest way to do this task is to give access to a group, rather than to individual users. No additional work is required by the application group/folder/cabinet administrator when another user needs access to the application group, folder, or cabinet. When a new user is added to the group, the user automatically gets access to the application group, folder, or cabinet. Adding the user to the group is the responsibility of the user administrator since the user administrator owns the groups in this model.

Another reason for providing access to application groups, folders, and cabinets from a group rather than to an individual user is that the application group/folder/cabinet administrator does not have access to the users and groups in this model. Since the application group/folder/cabinet administrator must first be given access to any users or groups that require access to application groups, folders, or cabinets, it is simpler and less time consuming to give access to a few groups rather than hundreds or even thousands of users. The application group/folder/cabinet administrator is given access to a group by adding the application group/folder/cabinet administrator to the group. This is done by the user administrator in this model. As a group member, the application group/folder/cabinet administrator will be able to see the group in the list and will therefore be able to grant the group access to the application groups, folders, and cabinets that have been defined by the application group/folder/cabinet administrator in this model.

To give an application group/folder/cabinet administrator access to a user, the user administrator must update each user and give the application group/folder/cabinet administrator access to the user. Once access has been given, the application group/folder/cabinet administrator will be able to see the user in the list and will therefore be able to grant the user access to the application groups, folders, and cabinets. Again, this is not the recommended approach because this task will have to be repeated each time that a user is added to the system.

To illustrate how the Object Owner model can be used, assume that a company installs a Content Manager OnDemand system to provide data archival and retrieval services for other organizations. The company provides the hardware and software required to administer the system and archive and retrieve the data. An administrator from each organization defines application groups and folders for their data. Another administrator defines the users that can access the data. The system must be able to limit access to an organization's application groups and folders. Only users defined by an organization should have access to the application groups and folders that are owned by the organization. The system must also be able to limit access to the data. Only users defined by an organization should have access to the data that is owned by the organization. By using the Object Owner model, both requirements can be met.

Summary

There can be many different variations of the two models that have been described. For example, in the Object Owner model, rather than one user administering both application groups and folders, one user can be defined to administer application groups and another user can be defined to administer folders. Choosing the right model or variation is an important decision that should be made early in the planning process. Changing to a different model later is not impossible but may require additional work if there are a large number of objects defined on the system.

Helpful hints

1. To simplify the task of providing access to application groups, folders, and cabinets, give access to a group rather than a user. When a new user needs access, add the user to the group.
2. To allow an application group/folder/cabinet administrator to see groups in the permissions list, add the application group/folder/cabinet administrator to the groups that require access to application groups, folders, and cabinets.
3. To allow multiple users to administer the same groups, create a group of users and make that group the group owner for any groups that need to be administered by multiple users.
4. The Create Groups authority is most effectively used if it is combined with the Create Users authority or added to a user administrator. Because the purpose of a group is to give a set of users permissions to another object, it is not very useful if the user that creates the group does not have access to any users. Otherwise, the user that creates a group must be given access to each user that needs to be added to the group.

Chapter 8. LDAP (Lightweight Directory Access Protocol) authentication support

The Lightweight Directory Access Protocol (LDAP) is an open industry standard that has evolved to share information between distributed applications on the same network, organize information in a clear and consistent manner, and prevent unauthorized modification or disclosure of private information. In recent years, LDAP has gained wide acceptance as the directory access method of the Internet, and becomes strategic within corporate intranets.

You can use LDAP to manage basic login authentication directly on the server, in other words, you no longer need to use the user security exit.

Requirements

The following LDAP servers are supported:

- Novell eDirectory Version 8.8 SP2
- Sun Java™ System Directory Server Enterprise Edition 6.3
- IBM Tivoli Directory server (TDS)
- Microsoft Active Directory Application Mode (ADAM) server
- Microsoft Active Directory (AD)

How Content Manager OnDemand works with LDAP

The following diagram illustrates how Content Manager OnDemand works with LDAP:

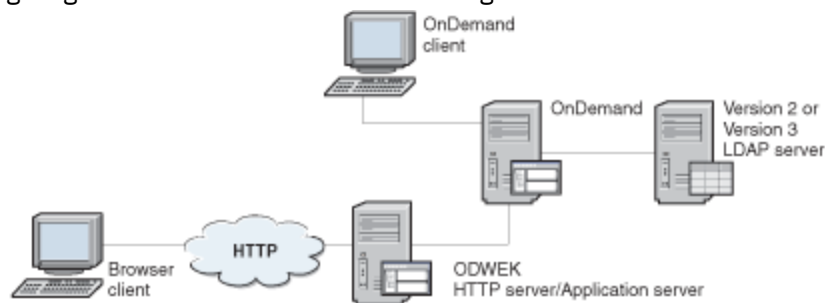


Figure 4: How OnDemand works with LDAP

When a Content Manager OnDemand client makes a login request to the Content Manager OnDemand server, if you enabled LDAP authentication in the server, the Content Manager OnDemand server makes an authentication request to the LDAP through either an anonymous or credentialed bind.

This initial call accesses the LDAP server, searches for the user's credentials and finds the user's distinguished name (DN). If the user's DN is found, the Content Manager OnDemand server makes another call to the LDAP server using that DN to confirm that the password that was given by the user is correct. If the password is correct, the LDAP server returns a mapped attribute in LDAP, which is usually the Content Manager OnDemand user ID. The Content Manager OnDemand server takes the attribute, and proceeds with its login.

Enabling LDAP authentication

To enable LDAP authentication, in the administrative client, right-click your Content Manager OnDemand server, and select **System Parameters**. In the System Parameters window, under LDAP Authentication, select the **Enable** check box. Under Login Processing, select the **Password Case Sensitive** check box.

To disable LDAP authentication, clear the **Enable** check box.

Chapter 9. Examples of using the administrative client

Overview

About the examples

The following examples demonstrate how to use the administrative client to:

- Add users and groups to the system. Users on the system obtain permission to open folders and access application group data from a group.
- Define a server printer that can be used as the default printer for an application.
- Specify the primary node for an HFS storage set, if you are using the Content Manager OnDemand cache storage feature. Content Manager OnDemand caches data on the object server identified by the primary storage node.
- Specify a collection name, if you plan to maintain data in archive storage on devices that are managed by Open Access Method.
- Specify a high level qualifier, if you plan to maintain data in archive storage on VSAM data sets.
- Add a report to the system. To add a report, you must add an application group, an application, and a folder.

System configuration

Here are the assumptions about the system:

- Two groups of users: the customer service department and users at Customer XYZ.
Note: This example assumes that customers access the system by using the Content Manager OnDemand Web Enablement Kit (ODWEK) feature. However, for purposes of demonstrating how the system works and how to use the administrative client to implement the requirements, these users could easily represent another department within the organization.
- Add the telephone bill report to the system. The telephone report is generated by an application program running on a z/OS system. Download is used to transmit the report data to the Content Manager OnDemand server. The report is indexed on the server.
- Content Manager OnDemand servers. The library server and the object server reside on the same workstation.
- Database. Use table spaces and create incremental backups each time that a report is loaded into the system. The database resides on RAID storage devices. Use Tivoli Storage Manager to maintain the database log files and backup image files. Tivoli Storage Manager maintains the files in an automated tape library.
- Cache storage. When a report is loaded into the system, Content Manager OnDemand stores a copy in cache storage and maintains the report for 60 days. Reports are cached on RAID storage devices attached to the object server.
- Archive storage. When a report is loaded into the system, Content Manager OnDemand stores a copy of the report in archive storage. Tivoli Storage Manager maintains the report in an optical storage library for five years.
- System parameters. Users must log on to the system with a password. Set the minimum password length to six characters. Accept the defaults for the other system parameters.

Adding a user

About the example

First, review the requirements of the users that need to access the telephone bill reports.

- One set of userids for the customer service department. Identify one user as a user administrator. The user administrator can add other users and reset passwords.
- One userid for Customer XYZ.

On the example system, users obtain permissions from groups. That means that the users are not added to application groups and folders. When groups are defined, the users are added to the groups.

The basics

In general, here is how you work in the administrative client to add a user. See the online help for details.

1. Choose a library server and select Users.
2. Pick one of two ways to add a user.
3. Define the properties of the user by completing fields in the Add a User dialog box.
4. Optionally add the user to groups.
5. Optionally add the user to application groups and set application group permissions.
6. Optionally add the group to folders and set folder permissions.
7. When finished, add the user by clicking OK in the Add a User dialog box.

Note: To add a user, the logon userid must be a user with create users authority, a user administrator, or a system administrator.

Choose a server

1. On the left side of the administrator window, click the name of the library server to which you want to add the user.
2. Expand the areas of the library server. Double click the server name or click the + (plus) to the left of the server name.
3. Select Users.

Two ways to add a user

You can add a user by using the New User command. You can also add a user by copying an existing user definition.

New User command

From the File menu, select New User to open the Add a User dialog box.

Copy command

You can use the copy command to add a user. In the User ID list, point to the user that you want to copy and click the right mouse button. Select Copy from the pop-up menu to open the Add a User dialog box. The fields in the dialog box contain information copied from the user you selected. At a minimum, you need to change the User ID (userids must be unique to the library server).

Adding the user

In the Add a User dialog box, click OK. The administrative client adds the user to the database and returns to the main window.

Adding the customer service users

Note: The following shows how to add two of the users. You can repeat the steps to add the others.

Use the New User command to add a user.

1. First, point to Users and click the right mouse button. From the pop-up menu, select New User to open the Add a User Dialog box.
2. In the User ID field, enter the name of the user: *CSR1*
3. Accept the UID generated by OnDemand.
4. Set the user's initial password to the userid. In the Password field, enter: *service1*
5. In the Verify Password field, enter: *service1*. The password needs to be between 6 and 8 characters.
6. In the Description field, enter: *Customer Service Representative*
7. Under User Type, select User Administrator. The user will be able to maintain userids on the system.
8. Click the User Information tab.
9. Complete the fields on the User Information page, such as the Name, Department, and Phone Number.
10. Click the General tab.

The user will obtain application group and folder permissions from a group. Add the user to the group when the group is added to the system. Therefore, do not add the user to application groups, folders, or groups at this time.
11. At this point, the properties of the user meet the requirements. Click OK to add the user.

Use the Copy command to add another user.

1. Point to CSR1 and click the right mouse button. From the pop-up menu, select Copy to open the Add a User Dialog box.
2. In the User ID field, replace *CSR1* with *CSR2*
3. Accept the UID generated by OnDemand.
4. Set the user's initial password to the userid. In the Password field, enter: *service2*
5. In the Verify Password, enter: *service2*
6. Under User Type, select User.
7. Click the User Information tab.
8. Replace the information in the fields on the User Information page.
9. Click the General tab.
10. At this point, the properties of the user meet the requirements. Click OK to add the user.

Adding a group

About the example

Review the requirements of the groups that need to access the telephone bill reports.

- Customer service group. Users that belong to the group can open the telephone bill report folder and query documents stored in the telephone bill report application group. When you define the report to the system, add the group to the application group and the folder. Identify a group owner. The group owner can add new customer service users to the group and remove users from the group.
- Customer XYZ group. Users that belong to the group can also open the telephone bill report folder and query documents stored in the telephone bill report application group. However, you can limit access to documents that contain a specific customer name and account number. When you define the report to the system, add the group to the application group and the folder and specify the necessary restrictions.
- Users. Add the users that were defined in [“Adding a user” on page 54](#) to the groups.

The basics

In general, here is how you work in the administrative client to add a group. See the online help for details.

1. Choose a library server and select Groups.
2. Pick one of two ways to add a group.
3. Define the properties of the group by completing fields in the Add a Group dialog box.
4. Optionally assign a group owner.
5. Optionally add users to the group.
6. Optionally add the group to application groups and set application group permissions.
7. Optionally add the group to folders and set folder permissions.
8. When finished, add the group by clicking OK in the Add a Group dialog box.

Note: To add a group, the logon userid must be a user with create groups authority or a system administrator.

Choose a library server

1. On the left side of the window, click the name of the library server to which you want to add the group.
2. Expand the areas of the library server. Double-click the server name or click the + (plus) to the left of the server name.
3. Select Groups.

Two ways to add a group

This section explains how to add a group by using the New Group command. You can also add a group by copying an existing group definition.

New Group command

From the File menu, select New Group to open the Add a Group dialog box.

Copy command

You can use the Copy command to add a group. In the Name list, point to the group that you want to copy and click the right mouse button. Select Copy from the pop-up menu to open the Add a Group dialog box. The fields in the dialog box contain information copied from the group you selected. At a minimum, you need to change the group name (group names must be unique to the library server).

Adding users

You can add one or more users to the group. Complete the following steps to add a user to a group.

1. From the List of Users list, select the user.
2. Click Add. The administrative client moves the user to the Users in the Group list.

To remove a user from the group, select the user in the Users in the Group list and click Remove.

Adding the group

In the Add a Group dialog box, click OK. The administrative client adds the group to the database and returns to the main window.

Adding the customer service group

Use the New Group command to add the group.

1. First, point to Groups and click the right mouse button. From the pop-up menu, select New Group to open the Add a Group Dialog box.

2. In the Name field, enter the name of the group: *CustomerService*. Do not include any blank in the group name.
 3. Accept the GID generated by OnDemand.
 4. From the Group Owner list, select the user that can add users to and remove users from the group: CSR1
- Note:** The user that you select does not obtain permissions from the group unless you add the user to the group (see step “6” on page 57). However, a group owner can add their userid to the group at any time.
5. In the Description field, enter: *Access to Telephone Bill Reports; call Leonard Little, x90565, for more information*
 6. From the List of Users list, select and add users to the group: CSR1, CSR2, CSR3, CSR4, and CSR5

Figure 5 on page 57 shows the Add a Group dialog box with the basic information completed.

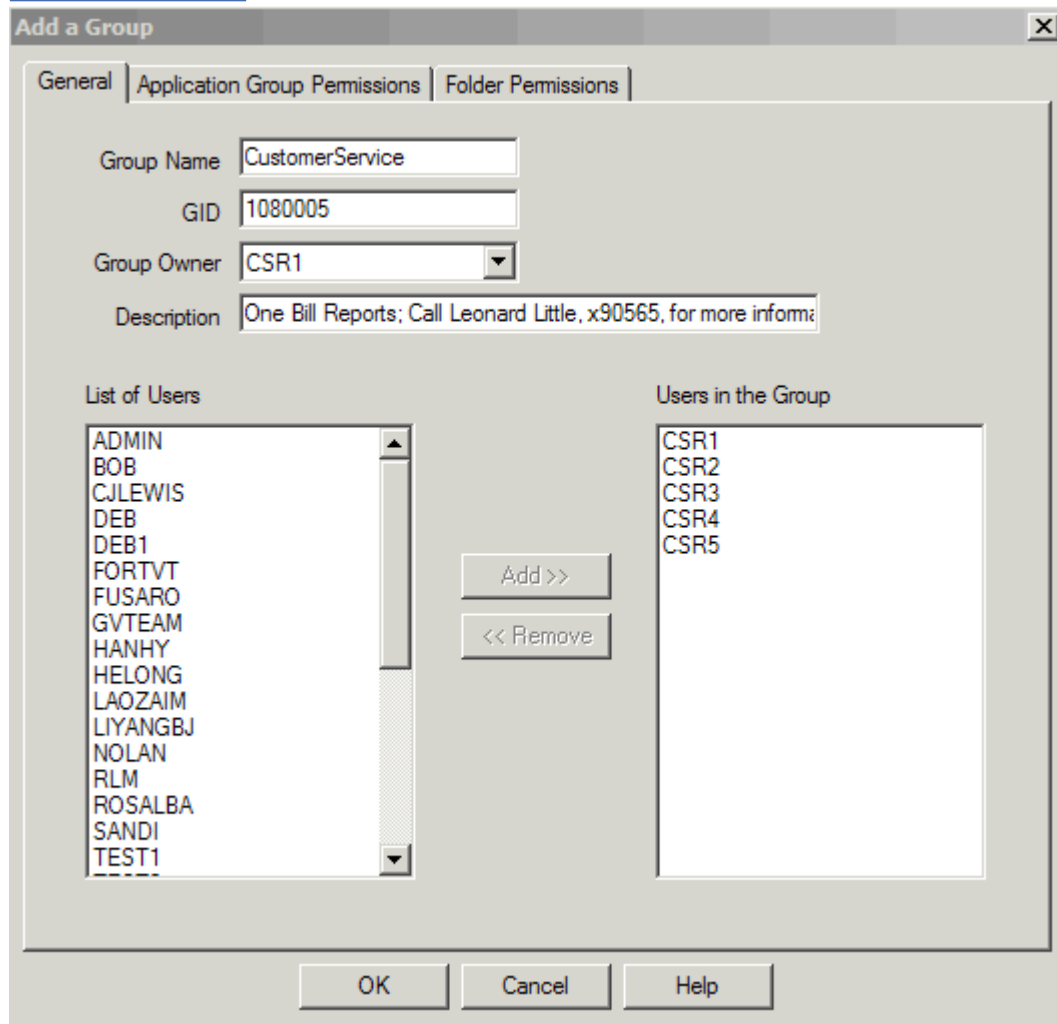


Figure 5: Add a Group dialog box

7. At this point, the properties of the group meet the requirements. Click OK to add the group.

Adding the Customer XYZ group

Use the Copy command to add the group.

1. Point to Customer Service and click the right mouse button. From the pop-up menu, select Copy to open the Add a Group Dialog box.
2. In the Name field, replace *CustomerService* with *CustomerXYZ*

3. Accept the GID generated by OnDemand.
4. Use a system administrator to maintain the group. Therefore, do not assign a group owner. In the Group Owner list, replace CSR1 with *NONE.
5. Replace the contents of the Description field with: *Access to Telephone Bill Reports by Customer XYZ*
6. From the Users in the Group list, remove CSR1, CSR2, CSR3, CSR4, and CSR5.
7. From the List of Users list, add XYZ1.

Figure 6 on page 58 shows the Add a Group dialog box with the basic information completed.

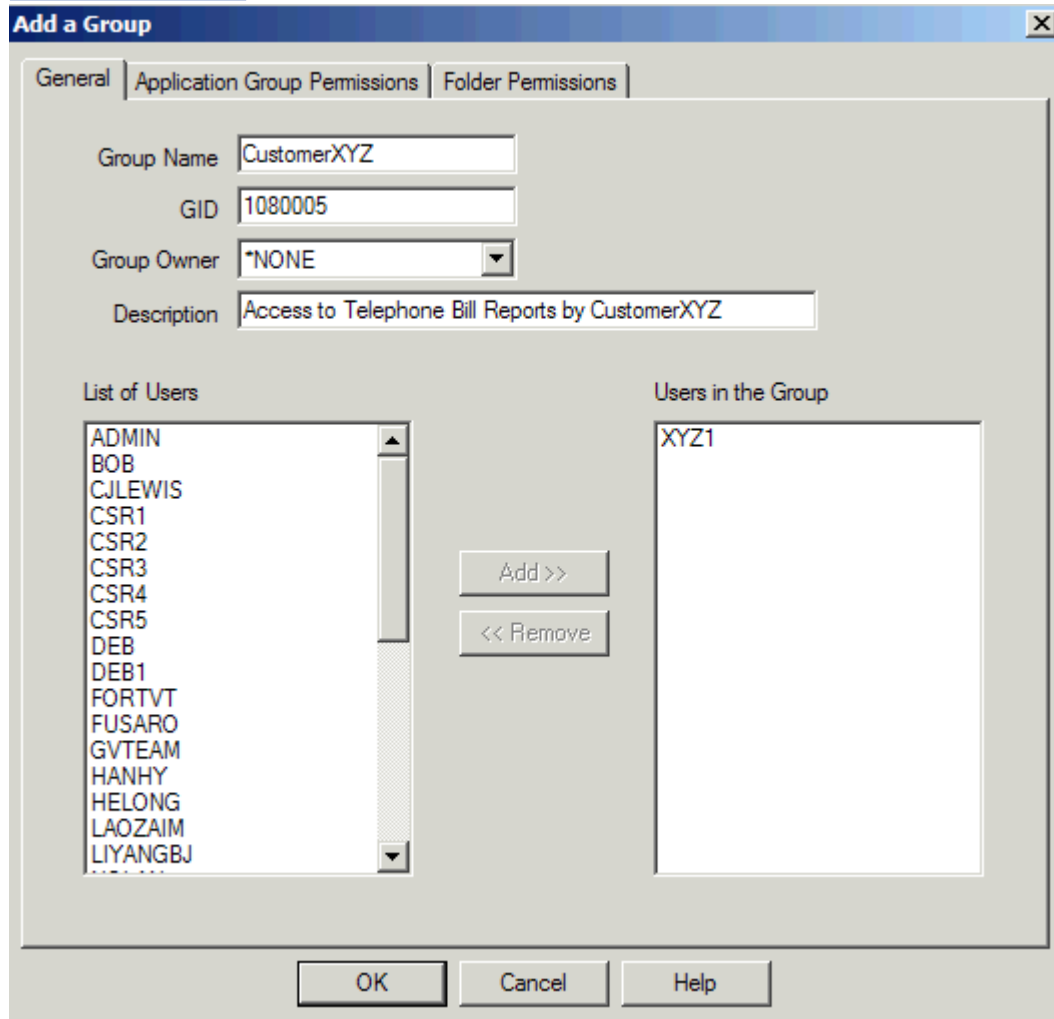


Figure 6: Add a Group dialog box

8. Click OK to add the group.

Adding a printer

About the example

Add a printer that can be used as the default server printer for the application that will be defined in “Adding a report” on page 64. The physical printer resides in the customer service department. By default, when users of the telephone bill report application select a document and choose the printer command, OnDemand sends the document to this print device.

The basics

In general, here is how you work in the administrative client to add a printer. See the online help for details.

1. Choose a library server and select Printers.
2. Pick one of two ways to add a printer.
3. Define the properties of the printer by completing fields in the Add a Printer dialog box.
4. When finished, add the printer by clicking OK in the Add a Printer dialog box.

Note: To add a printer, the logon userid must be a system administrator.

Choose a library server

1. On the left side of the main window, click the name of the library server to which you want to add the printer.
2. Expand the areas of the library server. Double click the server name or click the + (plus) to the left of the server name.
3. Select Printers.

Two ways to add a printer

You can add a printer by using the New Printer command. You can also add a printer by copying an existing printer definition.

New Printer command

From the File menu, select New Printer to open the Add a Printer dialog box. [Figure 7 on page 60](#) shows an example of the Add a Printer dialog box.

Copy command

You can use the copy command to add a printer. In the Name list, point to the printer that you want to copy and click the right mouse button. Select Copy from the pop-up menu to open the Add a Printer dialog box. The fields in the dialog box contain information copied from the printer you selected. At a minimum, you need to change the name (printer names must be unique to the library server).

Adding the printer

Use the New Printer command to add the printer.

1. First, point to Printers and click the right mouse button. From the pop-up menu, select New Printer to open the Add a Printer dialog box.
2. In the Name field, enter the name of the printer: *Customer Service Printer*.
3. In the Description field, enter: *Customer service printer for the telephone bill report application*.
4. In the Server Queue Name, enter: *ip60cs*.
5. Accept the default Printer Type of *Printer*.

[Figure 7 on page 60](#) shows the completed Add a Printer dialog box.

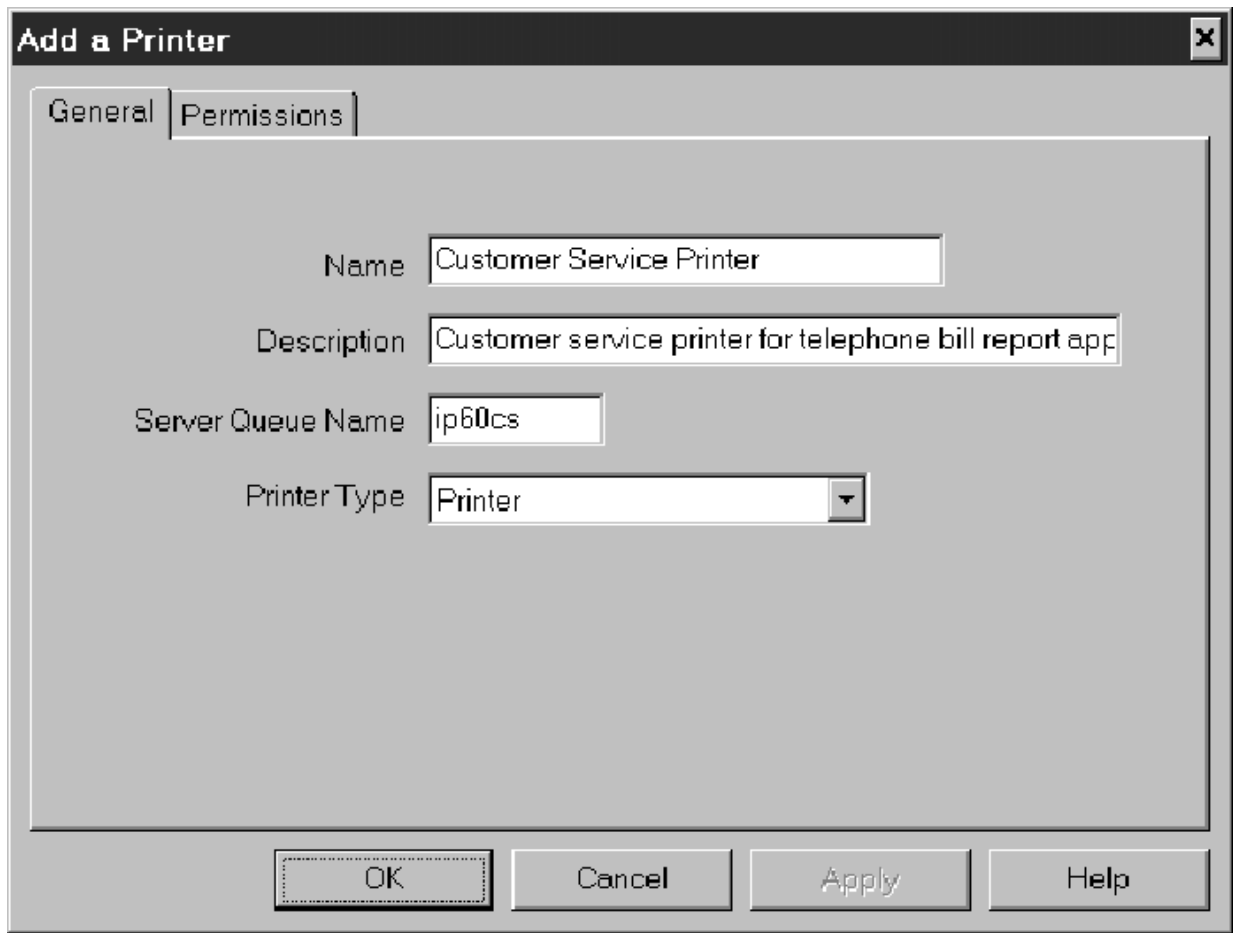


Figure 7: Add a Printer dialog box

6. At this point, the properties of the printer meet the requirements. Click OK to add the printer. The administrative client adds the printer to the database and returns to the main window.

Adding a storage set and storage node

About the example

Review the storage requirements for the telephone bill report that will be defined in [“Adding a report”](#) on page 64. Each report that is loaded into the system must be copied to cache storage. In addition, for long term storage, the system must maintain a copy of the report in archive storage.

- Cache storage. Content Manager OnDemand can automatically copy report data to cache storage on the object server identified by the primary storage node. However, you need to do two things to make sure that this happens. First, configure HFS storage. (The *Configuration Guide* describes how to configure HFS storage.) Second, specify the correct storage management information when you define the application group. For example, specify that the data stored in the application group is copied to cache storage and specify the number of days that you want the system to maintain the data in cache storage.
- Archive storage. This is the storage to which report data is copied for long term storage. Define one primary storage node in the storage set. The primary storage node will identify the object server on which the data is stored. In this example, the object server is on the same z/OS system as the library server. OAM is used to maintain data in archive storage. The primary storage node identifies a collection in OAM.

The basics

In general, here is how you work in the administrative client to add a storage set. See the online help for details.

1. Choose a library server and select **Storage Sets**.
2. Pick one of two ways to add a storage set.
3. Define the properties of the storage set by completing fields in the **Add a Storage Set dialog** box.
4. Add a primary storage node to the storage set by completing the fields in the **Add a Primary Node** dialog box.
5. When finished, add the storage set by clicking **OK** in the **Add a Storage Set** dialog box.

Note: To add a storage set, the logon userid must be a system administrator.

Choose a library server

1. On the left side of the main window, click the name of the library server to which you want to add the storage set.
2. Expand the areas of the library server. Double click the server name or click the + (plus) to the left of the server name.
3. Select Storage Sets.

Two ways to add a storage set

You can add a storage set by using the New Storage Set command. You can also add a storage set by copying an existing storage set definition.

New Storage Set command

From the File menu, select **New Storage Set** to open the **Add a Storage Set** dialog box.

Copy command

You can use the Copy command to add a storage set. In the Name list, point to the storage set that you want to copy and click the right mouse button. Select Copy from the menu to open the **Add a Storage Set** dialog box. The fields in the dialog box contain information copied from the storage set you selected. At a minimum, you need to change the Name (storage set names must be unique to Content Manager OnDemand).

Adding the storage set

Use the New Storage Set command to add the storage set.

1. First, point to Storage Sets and click the right mouse button. From the menu, select New Storage Set to open the Add a Storage Set Dialog box.
2. In the Name field, type the name of the storage set: Five Year Storage Set
3. In the Description field, type: Contact Leonard Little, x90059
4. Accept the default Load Type of Fixed. This means that Content Manager OnDemand will store data in the primary storage node that has the Load Data check box selected.

The following figure shows the Add a Storage Set dialog box with the basic information completed.

Add a Storage Set [?] [X]

Name

Description

Load Type

Storage Nodes

Primary Object Server	Primary Storage Node	Secondary Object Server	Secondary Storage Node

Primary Node

Figure 8: Add a Storage Set dialog box - basic information

5. Next, add a primary storage node to the storage set. In the Storage Nodes area, click Add to open the Add a Primary Node dialog box.
6. Accept the default Object Server of *ONDEMAND. This means that the object server (and primary storage node) reside on the same z/OS system as the library server.
7. In the Collection Name field, type: ARS. This is the identifier in OAM for the data that is stored in the primary node.
8. In the Description field, type: OnDemand Archive Storage.
9. Select the Load Data check box to identify this node as the primary storage node in which Content Manager OnDemand will load data for the application groups that are assigned to the storage set.

The following figure shows the completed Add a Primary Node dialog box.

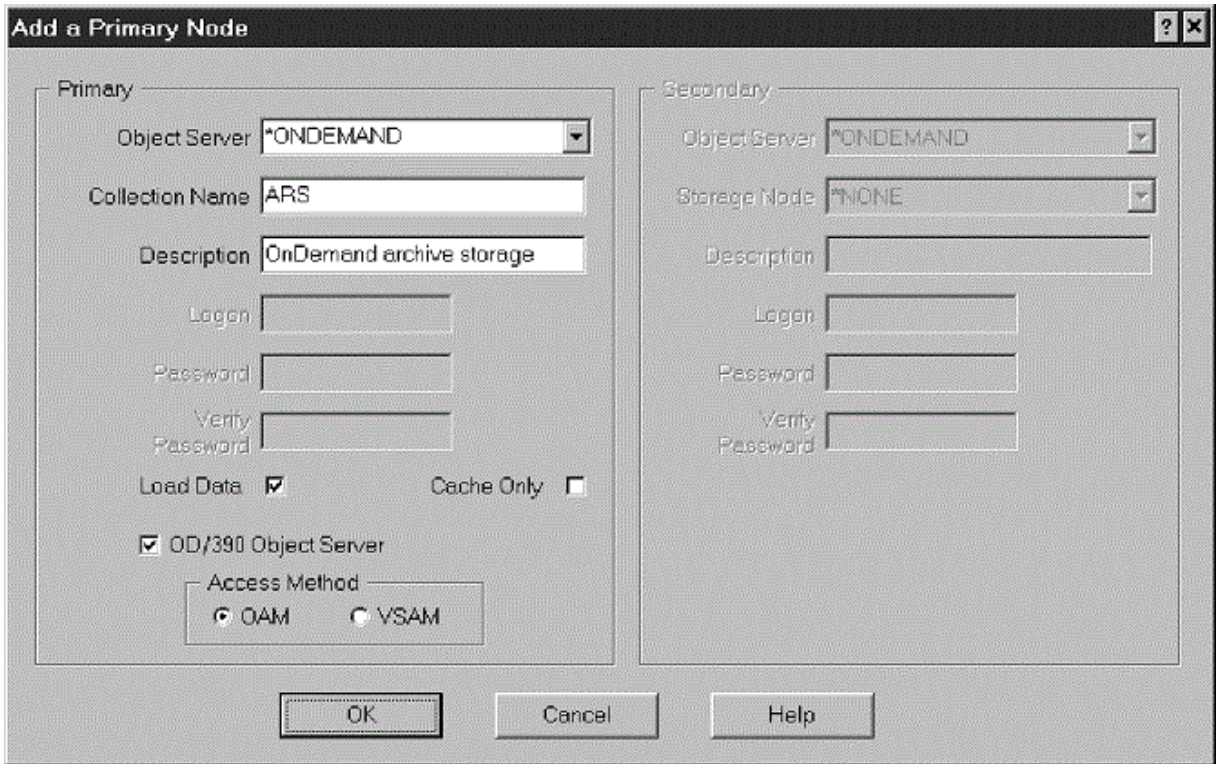


Figure 9: Add a Primary Node dialog box

10. At this point, the properties of the primary storage node meet the requirements. Click **OK** to add the primary storage node to the storage set and return to the Add a Storage Set window.
11. The following figure shows the completed Add a Storage Set dialog box.

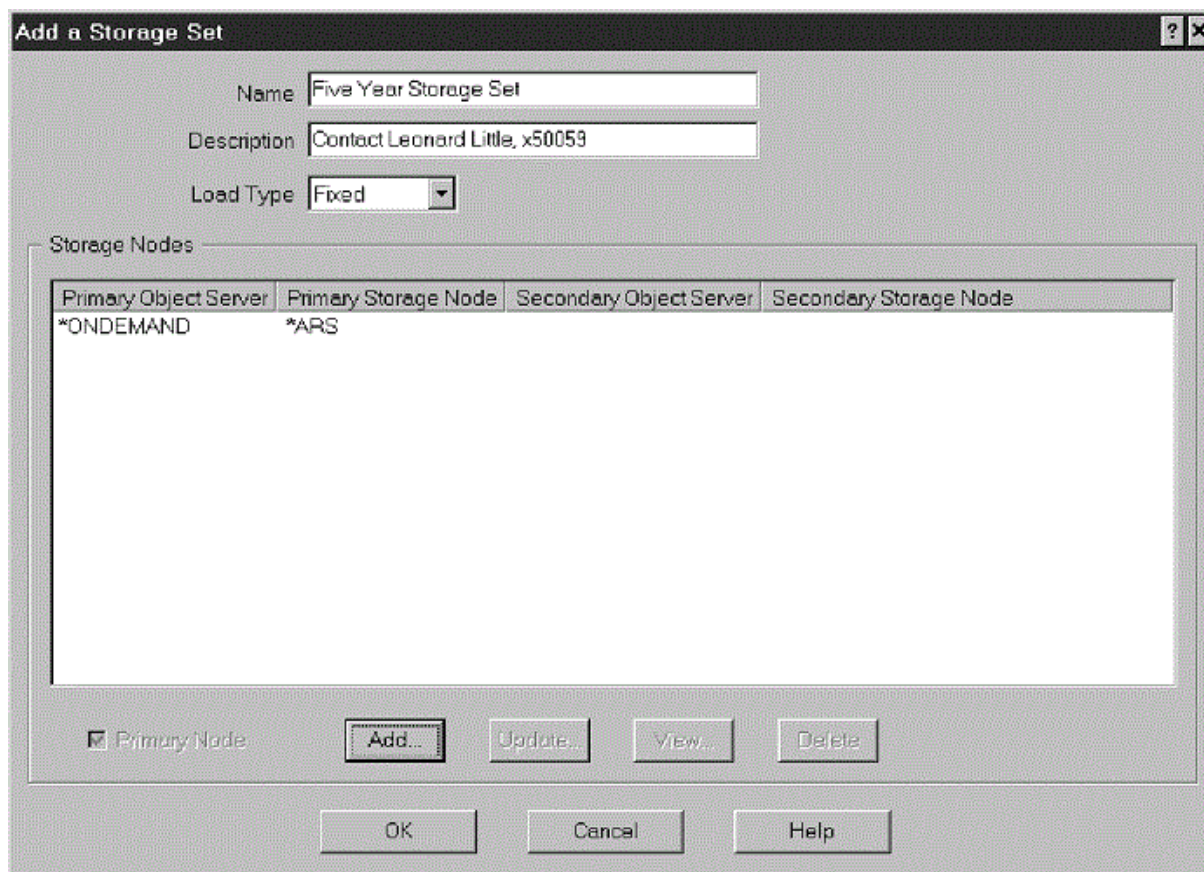


Figure 10: Add a Storage Set dialog box - completed

At this point, the properties of the storage set meet the requirements. Click **OK**. The administrative client adds the storage set to the database and returns to the main window.

Adding a report

About the example report

This section provides an overview of the sample telephone bill report that will be added to the system.

A telephone bill report typically contains hundreds of pages of line data. The report is logically segmented into statements. Users search for statements using a date and any combination of account number and customer name. The line data will be formatted into pages and the appearance of the output will be enhanced with images and fonts.

Most queries about a statement occur in the first sixty days after it is mailed to the customer. Little or no activity occurs a year after a statement is generated. For legal reasons, statements are kept for five years. The system must maintain index information in the most efficient way possible.

Two groups of users need to access the telephone bill reports. The customer service department is responsible for handling queries from customers. They answer questions about the statements, attach annotations to statements, and reprint statements. As part of a customer self-service initiative, Customer XYZ is permitted to access the system and retrieve and view their statements.

The basics

In general, here is how you work in the administrative client to add a report to the system:

1. Choose a library server. On the left side of the main window, click the name of the library server on which you want to define the report.

2. Select the area. First expand the areas of the library server. Double click the server name or click the + (plus) to the left of the server name. Then select the area.

When you define a report to the system, you typically add an application group, an application, and a folder.

- The application group identifies database and storage management information.
 - The application identifies viewing, indexing, loading, and printing information.
 - The folder provides users the ability to search for, retrieve, view, and print report data.
3. Pick one of two ways to add an application group, an application, and a folder. For example, use one of the following methods to add an application group:
 - Add a new application group. From the File menu, select New Application Group to open the Add an Application Group dialog box.
 - Copy an existing application group. In the Name list, point to the application group that you want to copy and click the right mouse button. Select Copy from the pop-up menu to open the Add an Application Group dialog box. The fields in the dialog box contain information copied from the application group you selected. At a minimum, you need to change the Name (application group names must be unique to the library server).
 4. Define the properties of the application group, application, and folder.
 5. When finished, add the application group, application, and folder by clicking OK in the add dialog box.

Note: To add an application group, the logon userid must be a system administrator, an application group/folder/cabinet administrator, or a user with create application groups authority.

To add an application, the logon userid must be a system administrator, an application group/folder/cabinet administrator, or a user with create application groups authority.

To add a folder, the logon userid must be a system administrator, an application group/folder/cabinet administrator, or a user with create folders authority.

About the application group

Before adding the application group, review the database and storage management requirements.

Database requirements

Database requirements can be grouped in two categories: database tables and the database fields.

Database tables

- Index data is stored in table spaces
- A database table contains index data from one or more reports
- Data is managed by OnDemand
- An annotation field is not required in the database
- The date field is the segment field for the application group

Database fields

- Three database fields: account number, customer name, and report date
- Account number is the index, data type string, 16 bytes. However, plan to store only 13 bytes in the database, removing the - (dash) characters from the account number string.
- Customer name is a filter, data type string, 30 bytes
- Report date is a filter, data type date

Storage management requirements

The storage management requirements determine where, how, and how long OnDemand maintains the report and index data.

- Maintain a report for five years
- Copy documents to cache storage and maintain them for sixty days
- Copy documents to archive storage when the report is loaded into the system
- Delete a table of index data at a time

Adding the application group

Use the New Application Group command to add the application group. First, point to Application Groups and click the right mouse button. From the pop-up menu, select New Application Group to open the Add an Application Group dialog box.

The pages of the Add an Application Group dialog box organize information about the application group into sections. The tabs show which page you are on: General, Message Logging, Storage Management, Permissions, Field Definition, Field Information, and Advanced Index Information.

General page

Start by completing the General page.

1. In the Name field, type the name of the application group.
2. In the Description field, type up to 120 characters of descriptive information about the application group.
3. Based on the database requirements, accept the recommended defaults for the rest of the fields on the General page. (You can click Advanced to see the other options.)

Message Logging page

1. Select the Message Logging tab to specify the types of application group messages that you want OnDemand to save in the system log. You can use the information in the system log messages to generate system usage reports.
2. Select Retrieval, Database Queries, and Server Printing. Clear all of the other check boxes.

Storage Management page

1. Select the Storage Management tab to provide information that OnDemand uses to manage data stored in the application group.
2. From the Storage Set Name list, select the name of the storage set that the system will use to maintain the report in cache storage and in archive storage. The storage set was added to the system in "Adding a storage set and storage node". The storage set can contain one or more primary storage nodes. A primary storage node identifies an object server and archive storage node in Tivoli Storage Manager. The system caches the report on the object server on which the primary storage node resides.
3. Under Cache Data, select Cache Data for __ Days and type 60 in the space provided. The system should maintain the report in cache storage for 60 days. After that time, when a user views or prints the report, the system retrieves the report from archive storage.
4. Under Life of Data and Indexes, select Expire in ____ Days and type 1825 in the space provided. This is the number of days (1825, or five years) that the system should maintain index data, documents, and resources related to the report. After that number of days, the system can delete the report from the system.

Permissions page

1. Select the Permissions tab to specify the types of report and application group functions that users can perform. For example, you can let users query report data, create logical views, print pages of the

report, and maintain the application group. You can specify default permissions and permissions for specific groups and users. By default, the person that creates the application group is given all application group permissions; no other users can access report data or maintain the application group. On the example system, other users obtain permissions from a group. The groups were added in [“Adding the customer service group”](#) on page 56 and [“Adding the Customer XYZ group”](#) on page 57.

2. First, add the customer service group. From the User/Groups list, select +CustomerService.
3. Select the Access check box.
4. Click Add.
5. Next, add the Customer XYZ group. From the User/Groups list, select +CustomerXYZ.
6. In the Annotation area, clear the View check box. The client also clears the Add check box. Customer XYZ users can view, print, and copy documents but do not have permission to use the annotation tools provided by the system.
7. Click Add.

Field Definition page

1. Select the Field Definition tab to define the database fields.
2. Define three database fields for the sample telephone bill report:
 - Account number
 - Customer name
 - Report date

To define a database field, type the name of the field in the Database Field Name field and click Add.

Field Information page

1. Select the Field Information tab to define the attributes of the database fields.
2. First, define field information for the account number field.
 - From the Type list, select Index
 - Type the string length (13)
3. Next, define field information for the customer name field.
 - From the Name list, select custname
 - From the Type list, select Variable
 - In the Length field, type 30
4. Next, define field information for the report date field.
 - From the Name list, select rdate
 - From the Data Type list, select Date
 - Select the Segment check box

For this example, leave the Advanced Index Information tab as default.

At this point, the properties of the application group meet the requirements. Click OK in the Add an Application Group window. The administrative client adds the application group to the database and returns to the main window.

About the application

Before adding the application, review the viewing, indexing, loading, and printing requirements.

Viewing requirements

- Source (line) data stored in OnDemand as AFP data

- Format data into pages and enhance the appearance with images and fonts
- Retrieve statements of one or more pages
- Define a default printer for the application. The default printer is where OnDemand sends documents when users select the server print command.

Indexing requirements

- Source data is EBCDIC, code page 500
- Segment report into groups of pages, one statement in each group
- Identify the beginning of a statement using:
 - Skip-to-channel one (X'F1')
 - PAGE 1 (X'D7C1C7C5404040F1')
- Generate three indexes for each statement: statement date, account number, and customer name
- Collect resources

Loading requirements

- Compress and store data in the most efficient method possible
- Application group database field names and index names match
- Date format is Mth d, yyyy
- Remove embedded - (dash) character from account number before storing value in the database

Adding the application

Use the New Application command to add the application. First, point to Applications and click the right mouse button. From the pop-up menu, select New Application to open the Add an Application dialog box.

The pages of the Add an Application dialog box organize information about the application into sections. The tabs show which page you are on: General, View Information, Indexer Information, Load Information, Logical View Fields, Logical Views, and Miscellaneous Options.

General page

Start by completing the General page.

1. In the Name field, type the name of the application.
2. In the Description field, type information about the application.
3. Click Select to open the Application Groups dialog box.
4. From the Names list, select Telephone Bill Reports.
5. Click OK.

View Information page

1. Select the View Information tab to specify information needed by OnDemand client programs to display the telephone bill report. This information is also used by the indexing program.
2. From the Data Type list, select Line.

Even though the report will be stored in the system as AFP data, initially set the Data Type to Line so that a sample of the source data can be processed with the graphical indexer. After generating the indexing parameters, you must reset the Data Type to AFP by selecting CONVERT=YES on the Data Format page of the Indexer Properties dialog box.

3. In the RECFM area, select Fixed. The report contains fixed length records, 133 bytes in length.

Indexer Information page

The Indexer Information page is where you can use a sample input report to define the indexing parameters and extract sample data values for use when processing reports in production.

Note: The example provides instructions for using the Content Manager OnDemand graphical indexer to process a sample report and create indexing information. The graphical indexer is part of the Content Manager OnDemand administrative client, a Windows application. To process a sample report, you typically create or extract a subset of a complete report. The report in this example was generated on a z/OS system, transferred to a Windows workstation (as a binary file) and then loaded into the graphical indexer.

It is important that the sample data that you use to create the indexing information matches the actual data to be indexed and loaded into the system. When you load a report into the system, Content Manager OnDemand uses the indexing parameters, options, and data values from the Content Manager OnDemand application to index the data. If the data being loaded does not match the data that you use to generate indexing parameters with the graphical indexer, then Content Manager OnDemand may not index the data properly. For example, Content Manager OnDemand may not be able to locate triggers, indexes, and fields or extract the correct index values.

1. Select the Indexer Information tab.
2. From the Indexer list, select ACIF.
3. Process a sample report using the graphical indexer. In the Parameters Source area, select Sample Data.
4. Click Modify to open the Open dialog box.
5. Select the name of the file that contains the sample report data.
6. Click Open. The client opens the Indexer Properties dialog box to the Data Format page. The Data Format page is where you can specify information that the client uses to read the report into the report window. For example, you can use the Maximum pages to display field to limit the number of pages that the client reads into the report window. Assume that the input report contains thousands of pages. You need only the first 20 pages of the report to define the indexing information. You can select the _____ Pages field and type 20 in the space provided.
7. No changes are required on the Data Format page for the example. However, when processing your own sample data, you must verify the information on the Data Format page; the report data will be displayed correctly only if all the information is correct for the type and format of the input data.
8. Click OK. The client reads the file into the report window.
9. Define the triggers for the report.
 - a. Define trigger number one. First, select any blank column in the first record. Click the Add a Trigger button to open the Add a Trigger dialog box.
 - b. In the Columns to Search area, select Carriage Control.
 - c. Click OK to add the trigger.
 - d. Define trigger number two. First, select the string PAGE 1. Click the right mouse button. From the pop-up menu, select Trigger to open the Add a Trigger dialog box.
 - e. Click OK to add the trigger.
10. Define the fields for the report.
 - a. Define field number one. First, select the string Customer XYZ and enough blank characters to the right of the string to reserve enough space to hold the largest index value (30 characters) the field can contain. (The selected string length guide, which appears above the field, displays the number of selected characters.) Then click the right mouse button. From the pop-up menu, select Field to open the Add a Field dialog box.
 - b. From the Trigger list, select Trigger2.
 - c. Click OK to add the field.

- d. Define field number two. First, select the string May 11, 1996. Then click the right mouse button. From the pop-up menu, select Field to open the Add a Field dialog box.
 - e. From the Trigger list, select Trigger2.
 - f. Click OK to add the field.
 - g. Define field number three. First, select the string 303-555-1212-95B. Then click the right mouse button. From the pop-up menu, select Field to open the Add a Field dialog box.
 - h. From the Trigger list, select Trigger2.
 - i. Click OK to add the field.
11. Define the indexes for the report.
 - a. First, clear any selected areas of the report. Right-click and select **Index** to open the Add an Index dialog box.
 - b. From the Attribute list, select custname.
 - c. In the Break area, select No.
 - d. In the Fields list, double-click Field1.
 - e. Click Add to add the index.
 - f. From the Attribute list, select rdate
 - g. In the Break area, select No.
 - h. In the Fields list, double-click Field2.
 - i. Click Add to add the index.
 - j. From the Attribute list, select acct.
 - k. In the Fields list, double-click Field3.
 - l. Click Add to add the index.
 - m. Click Done to close the Add an Index dialog box.
 12. In the toolbar, click the Indexer Properties button to open the Indexer Properties dialog box.
 13. On the Data Format page, change the Data Conversion field to Yes so that ACIF converts the input line data to AFP.

When you change Data Conversion to Yes, the administrative client automatically changes the Data Type on the View Information page to AFP.
 14. Define the resources for the report.
 - a. Click the Resource Information tab to specify the resources required to view the report.
 - b. Type the name of the form definition. This is the AFP object that determines how data is placed on a page.
 - c. Type the name of the page definition. This is the AFP object that determines other resources required to view a page.
 - d. In the Resource File Contents area, select the Form Definitions, Page Segments, and Overlays check boxes. These are the resources that ACIF should collect.
 - e. In the Search Paths area, type the full path names of the directories in which the indexing program can locate the resources.
 15. Click OK to close the Indexer Properties dialog box.
 16. Close the report window, saving the changes.

Load Information page

1. Select the Load Information tab to specify information that OnDemand uses to process the index data before storing it in the database.
2. In the Application Group Database Name list, select rdate to specify the format of the date as it appears in the report.

3. From the format list, select %b %d, %Y.
4. In the Application Group Database Name list, select acct. To conserve space in the database, OnDemand will remove the - (dash) character from index values before storing the values in the database.
5. In the Embedded field, type the - (dash) character.

Miscellaneous Options page

1. Select the Miscellaneous Options tab to specify client and server processing options for the application.
2. Under Client Options, in the Default Server Printer list, select Customer Service Printer. This is the printer that was defined in [“Adding a printer”](#) on page 58.

At this point, the properties of the application meet the requirements. Click OK in the Add an Application window. The administrative client adds the application to the database and returns to the main window.

About the folder

Before adding the folder, review the data access requirements, the types of permissions that should be specified, and the search and display fields that should be defined.

Data access requirements

The folder allows users to access the telephone bill report application group and the telephone bill report application.

Permissions

Who needs access to the folder and what types of permissions do the users need?

- Users in the customer service department can open the folder to search for and retrieve statements.
- Users at Customer XYZ can open the folder to search for and retrieve statements that contain their account number and customer name.
- Define a set of folder fields for the Customer XYZ users. The folder fields will limit access to specific statements.

Search and display fields

Define two sets of folder fields:

- One set at the folder level. These folder fields allow users in the customer service department to access any statement in the database.
- One set for the CustomerXYZ group. These folder fields allow users at Customer XYZ to access specific statements.

Adding the folder

Use the New Folder command to add the folder. First, point to Folders and click the right mouse button. From the pop-up menu, select New Folder to open the Add a Folder dialog box.

The pages of the Add a Folder dialog box organize information about the folder into sections. The tabs show which page you are on: General, Permissions, Field Definition, Field Information, and Field Mapping.

General page

Start by completing the General page.

1. In the Name field, type the name of the folder.
2. In the Description field, type up to 120 characters of descriptive information about the folder.

3. Select the Display Document Location check box. This provides users with a visual clue about the type of media on which a statement is stored.
4. In the Application Groups list, select Telephone Bill Reports.

Permissions page

1. Select the Permissions tab to specify the types of folder functions that users can perform. For example, you can let users open the folder, create private named queries, and maintain folder fields. You can specify default permissions and permissions for specific groups and users. By default, the person that creates the folder is given all folder permissions; no other users can open or maintain the folder. On our system, other users obtain permissions from a group. You need to add two groups to the folder. The groups were added in [“Adding the customer service group” on page 56](#) and [“Adding the Customer XYZ group” on page 57](#).
2. From the Users and Group list, select +CustomerService.
3. Select the Access check box.
4. Click Add, to add the Customer Service group to the folder.
5. From the Users and Group list, select +CustomerXYZ.
6. Click Add, to add the CustomerXYZ group to the folder.

Field Definition page

1. Select the Field Definition tab to define the folder fields. Define four folder fields to allow users to search for statements:
 - Account Number, a string field
 - Customer Name, a string field
 - Report Date, a date field
 - Other Information, a text search field
2. Complete the following steps to define a folder field:
 - a. In the Name field, type the name of the folder field.
 - b. In the Description field, type up to 120 characters of descriptive information about the folder field.
 - c. From the Field Type list, select the data type of the field.
 - d. Select the Mapping Type. All of the fields in this example use the Single mapping type.
 - e. Click Add.

Field Information page

1. Select the Field Information tab to specify the properties of the folder fields. Using the *PUBLIC identifier, you can specify field information that is used by all users that can open the folder. You can also specify field information for specific users and groups. The public field information will be used unless it is overridden by field information for a specific user or group. For this folder, do the following:
 - Specify public field information. For the Account Number, Customer Name, and Other Information fields, accept the default values. For the Report Date field, you must specify field information.
 - Specify field information for the CustomerXYZ group. For the Other Information and Report Date fields, accept the default values. For the Account Number and Customer Name fields, specify field information.
2. First, specify the public field information for the Report Date field.
 - a. From the Name list, select Report Date
 - b. From the ID list, select *PUBLIC
 - c. Select the Default check box
 - d. From the Display Format list, select %b %e, %Y

- e. From the Display Formats list, select %b %e, %Y
- f. In the Interval area, select Last, type a 3 (three) in the entry field, and select Months
3. Next, specify the field information for the CustomerXYZ group. First, make a copy of the folder fields. Click the Permissions tab.
4. From the Selected List, select +CustomerXYZ.
5. In the User/Group Fields area, click Yes.
6. Click the Field Information Tab.
7. Specify the Customer XYZ field information for the Account Number field.
 - a. From the Name list, select Account Number
 - b. From the ID list, select +CustomerXYZ
 - c. From the Default list, select Equal
 - d. Clear the Like check box
 - e. Select the Default check box
 - f. Select the Fixed check box
 - g. In the first Defaults entry field, type 1234567890123
 - h. Clear the Append check box
8. Next, specify the Customer XYZ field information for the Customer Name field.
 - a. From the Name list, select Customer Name
 - b. From the Default list, select Equal
 - c. Clear the Like check box
 - d. Select the Default check box
 - e. Select the Fixed check box
 - f. In the first Defaults entry field, type Customer XYZ
 - g. Clear the Append check box

Field Mapping page

1. Select the Field Mapping tab to map the folder fields to application group database fields. OnDemand uses the values that users type in folder fields to construct queries against the Content Manager OnDemand database.
2. Map the following folder fields to their corresponding application group fields:
 - Account Number to acct
 - Customer Name to custname
 - Report Date to rdate

At this point, the properties of the folder meet the requirements. Click OK in the Add a Folder window. The administrative client adds the folder to the database and returns to the main window.

Summary

The example shows the basic requirements for adding a report to the system. Hopefully the scenario that was described and developed is similar to how you plan to use OnDemand at your company. The example should have enough variations to show the flexibility of OnDemand to meet a range of business and operational requirements. Of course, there are several tasks that were not shown. For example, logical views of the report were not created. The system log user exit was not explored. And all of the ways to complete a given task or implement a specific requirement were not shown. As with most administrative software, there is more than one way to accomplish a task. Hopefully, the example showed you the most straightforward way to get things done with the administrative client.

You can use reference information provided with the product and the online help to find out more about how to use OnDemand. If you have questions and can't find the answers, please contact the IBM support center. IBM also offers classes that further explore how to administer the system. Finally, you can let IBM know how well the information in this book was presented and if you found the book helpful. The section titled How to Send Your Comments explains how to let IBM know.

Chapter 10. Loading AFP™ and line data

About this task

This section provides an overview of the data loading process.

Overview

You can archive the print output of your existing application programs in Content Manager OnDemand without changing the print data stream or writing programs to process the data. In Content Manager OnDemand, the print output of an application program is called a report. Content Manager OnDemand provides programs that can automatically index the report, add the index data to the database, divide the input data into indexed groups of pages (documents), compress the documents, and copy the compressed documents to storage volumes. After you load a report into the system, your users can query, retrieve, and view or print pages of the report using one of the Content Manager OnDemand client programs.

The Content Manager OnDemand data indexing and loading programs process input files that reside on a Content Manager OnDemand server. When you index a report that contains AFP data (or if you plan to convert line data to AFP data), you must make sure that the data indexing program can access the resources required by the report. Resources include page segments and fonts.

You can create up to 128 index fields for each type of report that you define to Content Manager OnDemand, providing many ways for users to query information contained in a report. The number of index 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 fields for the 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 fields for the date and transaction number. After you determine what index fields you need and define them to the system, Content Manager OnDemand extracts the index values from a report during the load process and stores them in records that are added to the database.

Content Manager OnDemand compresses report data into storage objects, using information that you specify in the application. Depending on how you configure storage management for your application groups, Content Manager OnDemand can automatically copy the report to cache storage and archive storage.

The load process saves messages in the system log each time that you load an input file into the system. You can open the System Log folder and view the messages for information such as the name of the input file, the indexing information, and the number of rows that were added to the database.

The load process provides checkpoints so that you can restart a previously interrupted load process. You can recover or restart the steps that add the index data to the database and copy the storage objects to storage volumes.

Before you begin loading input data

Estimating storage space

When you initially configure a Content Manager OnDemand system, you calculate the total amount of disk, optical, and tape storage required to hold the Content Manager OnDemand database, database log files, and reports. You also need to plan for the temporary space needed by Content Manager OnDemand programs. The amount of storage space that you need on your system is usually a factor of how much data that you plan to store in Content Manager OnDemand, how long you need Content Manager OnDemand to maintain the data, the compression ratio that you can expect to achieve on the report data,

and the number of copies of reports that you need the system to maintain. If you plan to index your reports on the Content Manager OnDemand server, then you must allocate temporary space for the data indexing program. Temporary space is also required for the data loading program.

After you determine how much storage space you need, you should organize the physical storage devices for the various components of the system, such as the database, and cache storage. If you plan to maintain data in archive storage, then you also need to prepare storage devices for use by the archive storage manager (VSAM or OAM). Physically separating program directories, the database, cache storage, and temporary work spaces can improve performance.

Defining the application group

About this task

When you load a report into the system, the ARSLOAD program adds index data to the database and compresses report data into storage objects and copies the storage objects to storage volumes. The index data and the storage objects are associated with an application group. As part of defining a report to Content Manager OnDemand, you must define or specify the application group that Content Manager OnDemand uses to maintain the data. The application group specifies the database fields that hold the index data and the storage management information that determines where Content Manager OnDemand maintains the report data and how long it maintains the index data and the report data. You can use the Content Manager OnDemand administrative client to define an application group. See the online help for the administrative client for more information about defining database fields and specifying storage management information for application groups.

If you plan to maintain a copy of your reports in archive storage, the application group must use a storage set that includes a primary node that identifies a collection name (OAM) or the high level qualifier of the data sets (VSAM) that will be used for storage.

Note:

1. OAM and VSAM are documented under DFSMS/MVS system management software. DFSMS/MVS product information is available from the z/OS and OS/390® Web sites. . For OAM, see *z/OS DFSMS Object Access Method Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426.
2. VSAM datasets to be used for archive storage of Content Manager OnDemand documents must be managed by SMS.

Defining the application

About this task

Most customers define an application for each different report (or source of data) that they plan to load into the system. When you create an application, you specify information about the report, such as:

- The application group in which you want to store the report
- Physical information about the report, including the type of data found in the report and the code page of the data
- The indexing parameters, if you plan to index the report on the server
- The processing that Content Manager OnDemand should do to the index data before adding it to the database

You can use the Content Manager OnDemand administrative client to define an application.

Configuring the ARSLOAD program

About this task

The ARSLOAD program is the primary Content Manager OnDemand data indexing and loading program. The ARSLOAD program determines if the input data needs to be indexed, and if it does, calls the indexing program. The ARSLOAD program then processes the index data, adding it to the database, optionally compresses the report data into storage objects, and copies the storage objects to storage volumes.

You run the ARSLOAD program each time that you want to load a report or set of reports into the system. You can run the ARSLOAD program either as a started task or from a job to periodically check for input data to process.

Note: The OnDemand server program (ARSSOCKD or ARSOBJD) must be running, otherwise the ARSLOAD program will fail.

See [ARSLOAD](#) for more information about the ARSLOAD program, including the parameters that you can specify, examples, and sample JCL.

Specifying input data to process

About this task

The ARSLOAD program can use the following sources for input files to process:

- one or more classes specified with the -C parameter,
- one or more file systems specified with one or more -d parameters,
- a destination that is specified with the -D parameter,
- a FORM name that is specified with the -F parameter,
- a WRITER name that is specified with the -W parameter, or
- one or more load file names.

If you omit the load file name, the ARSLOAD program will run in daemon mode and attempt to load input data from the JES spool (if any of the JES spool options -C, -D, -F, or -W were specified) or poll the directories that are specified by the -d parameter for input data to process. If you omit the load file name and you do not specify at least one of the daemon mode parameters (-C, -D, -F, -W, or -d), the ARSLOAD program will issue a usage note and exit without processing.

Important: For JES3, only SPOOL datasets in the "hold for XWTR" queue are selected. It is recommended that all SYSOUT datasets to be loaded through the ARSLOAD daemon processing be written to a SYSOUT class defined as "HOLD=EXTWTR".

See *JES3 Initialization and Tuning Reference* for instructions on how to define a SYSOUT class as "HOLD=EXTWTR".

Input file names

About this task

The ARSLOAD program can derive the name of the application group to load from the input file name. To do so, the input file name must conform to the following format:

```
WRITER.MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSS.ARD
```

Important: The .ARD file name extension is required to initiate a load process.

By default, the ARSLOAD program uses the FORMS part of the file name to identify the application group to load. However, you can use the -G parameter to specify a different part of the file name (WRITER, MVS™, JOBNAME, or DATASET) to identify the application group to load.

If the application group contains more than one application, you must identify the application to load; otherwise the load will fail. You can run the ARSLOAD program with the -A parameter to specify the part of the input file name (WRITER, MVS, JOBNAME, DATASET, or FORMS) that identifies the application.

Userid and password

About this task

When you run the ARSLOAD program, you must specify a userid with administrator authority for the application group. You must also specify a password for the user. There are several ways that you can provide the userid and password:

- You can specify the -u and -p parameters each time that you run the ARSLOAD program. The *IBM Content Manager OnDemand for z/OS: Configuration Guide* describes these parameters and how to specify them to the ARSLOAD program.
- Assign a userid to the ARSLOAD started task via the RACF® STARTED class and use the ARS.PTGN exit to provide a PassTicket. See *z/OS SecureWay Security Server RACF System Programmer's Guide, SA22-7683* for information about the RACF STARTED class. See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for information about the ARS.PTGN exit and creating PassTickets.

Indexing input data

About this task

You must generate index data for a report before you load the report into the system. Content Manager OnDemand supports several programs that you can use to generate index data for your reports:

- AFP Conversion and Indexing Facility (ACIF), to index AFP data and line data reports. ACIF can also collect AFP resources. If the report contains AFP data (or you are converting line data to AFP data), you need to store the AFP resources in OnDemand. The resources are required to display and reprint pages of a report that contains AFP data.
- Content Manager OnDemand OS/390 Indexer, to index line data reports and fully resolved AFP data streams.
- Content Manager OnDemand IBM Content Manager OnDemand PDF Indexer for Multiplatforms, to index Adobe PDF input files
- Content Manager OnDemand Generic Indexer, to index almost any other type of input data.

If the report contains AFP data (or you are converting line data to AFP data), then you need to store the AFP resources in OnDemand. The resources are required to display and reprint pages of a report that contains AFP data.

Important: Some users find that they are able to pass large files through ACIF successfully, and then the job fails during the load. This problem is typically related to ulimits. RSS and file size usually have default settings from installation. Increasing these values usually fixes this issue.

For details about indexing data, see the *IBM Content Manager OnDemand: Indexing Reference*.

Processing the input data

About this task

After a report has been indexed, the ARSLOAD program can process the index file to prepare the index data for loading into the database and prepare the report data and resource group files for the storage manager to load on storage volumes.

Processing index data

About this task

The index file that ACIF creates is processed by the ARSLOAD program before the index data is added to the database. The ARSLOAD program extracts information from the application group and the application and performs the following processing:

- Extracts the database field information from the application group.
- Extracts the preprocessing information from the application. This step is required if the field names in the index file are not the same as the database field names. Then you must map the field names so that Content Manager OnDemand stores index values in the correct database fields.
- After preprocessing the index file, the ARSLOAD program creates the database rows:
 - One row for every group of indexed pages in a report that contains a sorted transaction value
 - One row for every indexed item in a report that contains logical items, such as policies and statements
- Extracts the postprocessing information from the application. You specify exactly what processing you want the ARSLOAD program to do 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 Content Manager OnDemand. An index row contains:
 - One column for each field defined in the application group
 - One or more columns of OnDemand control information

Processing reports and resources

About this task

The ARSLOAD program divides the input data into indexed groups of pages (documents) and compresses the documents into storage objects. Dividing a report into groups of pages 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 program:

- Extracts the compression information from the application.
- Extracts the storage management information from the application group. The storage management information determines the storage locations, such as cache storage and archive storage.
- Compresses the documents into storage objects. Content Manager OnDemand compresses report data into approximately 100 KB blocks (a default value) and places the blocks into a storage object. Content Manager OnDemand uses a 10 MB storage object (a default value) to improve storage efficiency and performance. Compressed data does not span storage objects. Content Manager OnDemand assigns unique file names to the storage objects and sequentially numbers them within an application group.

Loading index data

About this task

The ARSLOAD program works with the database manager to add the index data to the database. Depending on the database organization that you specified when you defined the application group, Content Manager OnDemand either creates a new database table each time that you load a report or adds the index data to an existing database table.

Content Manager OnDemand uses a *segment table* as a high level index to the index data for an application group. Each row in the segment table identifies a specific table of application group index data. The fields in the segment table identify the application group and the dates found in a table. The

dates represent the earliest and latest date that can be found in that segment of application group index data. Content Manager OnDemand can use the segment table to limit a query to a specific table of application group index data.

The database manager updates the segment table if the beginning date in the report is earlier than a date already stored in the table.

Content Manager OnDemand limits the size of a table to improve performance and storage management. The number of rows in a table can be specified when you define the application group. The default size of a table is ten million rows. The database manager automatically closes a table and opens a new table when this threshold is reached. When closing a table, the database manager updates the segment table with the latest ending date found in the table.

Loading storage objects

About this task

Cache storage

The primary purpose of cache storage is for short-term, high-speed retrieval of report data. Cache storage is located in HFS storage. Reports are always stored in cache storage on the object server identified by the active storage node.

If you configure your application groups to copy data to cache storage, then the storage manager copies the storage object to cache storage. The `Cache Data for xx Days` setting on the Storage Management page determines whether Content Manager OnDemand copies documents to cache storage.

Archive storage

A storage node can identify a client node in OAM or VSAM. Content Manager OnDemand uses archive storage to maintain storage objects for long-term storage and for backup copies of reports.

The storage manager can copy the storage object to archive storage when the report is initially loaded into the system or at a later time, depending on how you configure your application groups. Most customers configure the system to copy report data to cache storage and archive storage at the same time.

Resources

If caching of data is enabled, Content Manager OnDemand stores resources in the cache, and the resources are not expired from the cache to provide fast retrieval when a user selects an item for viewing.

The ARSLOAD program saves one copy of a resource on each node for each application group. The resource can be stored multiple times depending on how the ARSLOAD program compares the data. ARSLOAD compares the last 50 resources against the resource generated by the load. If a match is not located, then a new resource is stored.

When processing a resource group file, the ARSLOAD program checks the resource identifier to determine if the resource is already present on the system.

If the storage node identifies a client node in OAM or VSAM, then the storage manager copies the resources to archive storage.

Verifying processing

About this task

When you load an input file into the system, the ARSLOAD program saves a copy of the messages generated during the load process in the system log. (Search for message number 87 in the system log.) After a load process completes, you can open the System Log folder and view the messages. The

information in the messages includes the date and time that the load process started and completed, the name of the input file, and the number of rows that were added to 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
        Content Manager 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 rows that Content Manager OnDemand added to the database:

- For a report that contains transaction data that you have divided into 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 a report contains 150,010 pages and there are 100 pages in an indexed group of pages, then Content Manager OnDemand should have added 1,501 rows to the database.
- For a report that contains logical items, such as statements and policies, the number of rows added to the database should equal the number of indexed items in the report. For example, if a report contains 1,000 statements, OnDemand should have added 1,000 rows to the database.

The Content Manager OnDemand Load ID represents the data that the ARSLOAD program stored in the system during a load process. The Load ID can be used to identify a specific load process. For example, you can run the ARSADMIN program and specify the Load ID to delete the index data and documents that were created when the ARSLOAD program processed an input file. For an explanation of the format of the load ID, see the description of the `-L` parameter in [“Parameters” on page 121](#).

Backing up databases

After you load reports into Content Manager OnDemand, IBM recommends that you create a backup copy of the Content Manager OnDemand database. When you backup the Content Manager OnDemand database, you protect control information and index data that Content Manager OnDemand and the database manager need to support the system. Also, if you maintain DB2[®] archived log files on disk, then after the backup completes, the database manager can reclaim the disk space used by the log files.

Utilities provided with DB2 must be used to take backups of the database tables. See the appropriate DB2 reference material for information about DB2 table maintenance; also see your installation's database administrator.

Chapter 11. Loading image files

About this task

Overview

Content Manager OnDemand provides support for storing and retrieving letters and other types of correspondence. The most straight forward way to store this type of information is to transform a printed copy of a letter into an image file using 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 the ARSLOAD program to load the index data into the database and store the letter file on the system. Your users can then query, retrieve, and view, print or send copies of the letter using one of the Content Manager OnDemand client programs.

Content Manager OnDemand provides the Generic indexer so that you can index input files that contain data other than AFP data, line data, and PDF. You specify the index data for the input files that you want to index with the Generic indexer in a parameter file. The parameter file contains the index field names and values and identifies the input files that you want to process. You can create up to 128 index fields for each input file that you want to load into the system, providing many ways for users to query and retrieve the letters. The number of index fields that you define usually depends on how your users retrieve documents from the system. In the example that follows, index fields were specified for the date on the letter, the name of the person that sent the letter, the company name, and the subject of the letter.

This section describes how to use a scanner and image capture software to create the image files. Before you can load the image files into the system, you must define an application group to manage the storage of the letters, an application to specify the physical and logical properties of the letters, and a folder to let users search for and retrieve the letters.

Defining the application group

When you load an input file into the system, Content Manager OnDemand updates the database with the index data that you provide and stores the indexed groups of pages as documents in cache storage and archive storage. The application group contains the information that Content Manager OnDemand uses to store and maintain the index data and the documents on the system. This section provides information about some of the key properties of the application group. You can use the Content Manager Content Manager OnDemand administrative client to define an application group.

Database Organization

The Database Organization determines how Content Manager OnDemand organizes the index data that is stored in the application group.

Accept the default Database Organization of Multiple Loads per Database Table and Table Size of 10 million rows. Each time that a letter (input file) is loaded into the system, Content Manager OnDemand adds one row to a database table. When a table reaches 10 million rows, Content Manager OnDemand closes the table and creates a new table. Content Manager OnDemand always adds index records to the open table; the closed tables can be queried.

Expiration Type

The Expiration Type determines how Content Manager OnDemand deletes index data and documents from the application group.

Accept the default Expiration Type of Load. This means that Content Manager OnDemand deletes the index data and documents from one load process (one or more input files) at a time from the application group. Depending on the number of input files that are processed during a load process, one or more letters (input files) may be deleted at a time. For example, if you create a parameter file for the Generic Indexer that contains index data for several input files, then Content Manager OnDemand will delete all of the index data and documents from the application group at the same time.

Permissions

Application group permissions determine the users that can access information stored in the application group and determine the users that can do other types of tasks related to the application group.

Under the *PUBLIC identifier, specify the Access permission so that all of the users defined to the library server can access data stored in the application group.

Field Definition

The Field Definition page is where you define the database fields for the application group. When you load an input file into the system, Content Manager OnDemand stores the index data that is specified in the parameter file into fields in records that are added to the database. When a user searches for letters, Content Manager OnDemand compares the search criteria entered by the user with index data in the application group.

Define the following database fields. The fields allow 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.

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.

Defining the application

Most customers define an application for each different source of input data that they plan to load into Content Manager OnDemand. This section provides information about some of the key properties of the application. You can use the Content Manager Content Manager OnDemand administrative client to define an application.

Application Group

You must assign an application to an application group. Assign the application to the application group that was created in [“Defining the application group” on page 83](#).

Data Format

The Data Format determines the kind of data that is found in the documents that are stored on the system. If the type of the data that is found in the original input file is different than the type of the data that is found in the documents that are stored on the system, then you should specify the type of the data that will be stored in Content Manager OnDemand. (In this example, the type of the data is the same – a TIFF image.) Select TIFF as the Data Type. When saving the scanned image of a letter, save the data as a TIFF image.

Indexer

The Indexer determines the indexing program that Content Manager OnDemand uses to index and convert input data. Select Generic as the Indexer. To store TIFF images in the system, you must index them with the Generic indexer program that is provided with Content Manager OnDemand.

Note: some users may get the following error when trying to load TIFF data to Content Manager OnDemand:

```
"arsload: Processing file >D:\SourceDBHK\SOURCE\120040511.txt<
arsload: 05/20/04 18:13:06 -- Loading started, --UNKNOWN-- bytes to
process
Unable to allocate enough memory. File=arslacif.c, Line=402
OnDemand Load Id =
>5242-3-0-20040101000000-20040101000000-5243<
An error occurred. Contact your system administrator and/or consult
the System Log. File=arsadmin.c, Line=1651
Unable to store the object >27FAAA<. Object size 8
Loaded 0 rows into the database".
```

The failure is because of the 0 byte file. It does not work to try to reallocate a buffer to the 0 byte file. OnDemand issues a generic allocation message "Unable to allocate enough memory. File=arslacif.c, Line=402". The 0 byte file loading is not supported by OnDemand.

Data Compression

The Data Compression determines whether Content Manager OnDemand compresses the input files.

Select Disable, so that Content Manager OnDemand does not attempt to compress the input files. In this example, the scanned image files are saved as compressed TIFF images, Therefore, it is not necessary for the system to compress them.

Defining the folder

You must define a folder so that users can search for and retrieve the input files that you load into Content Manager OnDemand. This section provides information about some of the key properties of the folder. You can use the Content Manager Content Manager OnDemand administrative client to define a folder.

Application Group

A folder can be used to search one or more application groups. Select the application group that was created in [“Defining the application group”](#) on page 83. When users open the folder, they can search for and retrieve the input files that were loaded into the application group.

Permissions

Folder permissions determine the users that can open the folder and determine the users that can do other types of tasks related to the folder.

Under the *PUBLIC identifier, specify the Access permission so that all users defined to the library server can open the folder.

Field Definition

The Field Definition page is where you define the search fields for the folder. The search fields contain the search criteria entered by the user. For most folders, you probably want to define a search field for each field that you defined for the application group.

Define the following fields. The fields allow 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.

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.

Field Mapping

The Field Mapping page is where you *map*, or associate, the folder fields to the application group fields. This is how you specify that the search criteria that a user enters in a particular folder field should be used to search a specific application group field. Map each of the folder fields to their corresponding application group fields. For example, map the folder field named Letter Date to the application group field named ldate.

Accessing the input files

About this task

The ARSLOAD program runs on the Content Manager OnDemand server. Therefore, the files that you want to load into the system must be stored on the server or you must provide network access to files that reside on some other system.

In the example, the files were copied from a PC to the server as binary files with a file type of TIF.

Store the input files on the server in the location from which you plan to run the ARSLOAD program. Otherwise, specify the full path name of the input files in the Generic index file.

Creating index data

About this task

You must create index data for your input files before you can load them into Content Manager OnDemand. If the input data is other than AFP, line data, or PDF, then you must use the Generic indexer to load it into the system. You specify index data for the Generic indexer using a parameter file. You should use a standard text editor to create the parameter file. [Figure 11 on page 87](#) shows an example of a parameter file that can be used by the Generic indexer to process image files.

See the *IBM Content Manager OnDemand Indexing Reference* for more information about the Generic indexer and the parameter file.

```
CODEPAGE:819
COMMENT:      input file number 1
GROUP_FIELD_NAME:ldate
GROUP_FIELD_VALUE:09/01/95
GROUP_FIELD_NAME:name
GROUP_FIELD_VALUE:Mr. Earl Hawkins
GROUP_FIELD_NAME:company
GROUP_FIELD_VALUE:Soft Products
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:optical storage devices
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter1.tif
COMMENT:      input file number 2
GROUP_FIELD_NAME:ldate
GROUP_FIELD_VALUE:09/01/95
GROUP_FIELD_NAME:name
GROUP_FIELD_VALUE:Hans G. Piker
GROUP_FIELD_NAME:company
GROUP_FIELD_VALUE:MBI Company
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:optical storage devices
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter2.tif
COMMENT:      input file number 3
GROUP_FIELD_NAME:ldate
GROUP_FIELD_VALUE:09/16/95
GROUP_FIELD_NAME:name
GROUP_FIELD_VALUE:Laurie Unicolos
GROUP_FIELD_NAME:company
GROUP_FIELD_VALUE:Dove Properties
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:account balance due
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter3.tif
COMMENT:      input file number 4
GROUP_FIELD_NAME:ldate
GROUP_FIELD_VALUE:10/01/95
GROUP_FIELD_NAME:name
GROUP_FIELD_VALUE:George VanLocal
GROUP_FIELD_NAME:company
GROUP_FIELD_VALUE:Express American
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:airline fares
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter4.tif
```

Figure 11: Example of a Generic indexer parameter file

Save the parameter file as LETTERS.IND on the server in the directory from which you plan to run the ARSLOAD program.

Configuring the ARSLOAD program

About this task

Note: The Content Manager OnDemand server program must be running, otherwise the ARSLOAD program will fail.

The ARSLOAD program is the primary Content Manager OnDemand data indexing and loading program. The ARSLOAD program determines if the input data needs to be indexed, and if so, it calls the indexing program. The ARSLOAD program then processes the index data, loading it into the database, optionally compresses the input data into storage objects, and copies the storage objects to storage volumes. See Chapter 24, “ARSLOAD,” on page 185 for more information about the ARSLOAD program, including the parameters that you can specify to process input files.

You typically run the ARSLOAD program each time that you want to load files into the system. You can either run the ARSLOAD program as a started task or from a job to periodically check for input data to process. To support a low volume scanning operation, most customers choose to run the ARSLOAD program from the TSO OMVS command line. For example:

```
/usr/lpp/ars/V10R1M0/bin/arsload -u admin -p "" -n -g Letters letters
```

Specify the following parameters to the ARSLOAD program:

-u admin

The name of a Content Manager 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 Content Manager OnDemand user.

-n

Do not delete the input files.

-g Letters

The name of the application group to load.

letters

The name of the input files to process.

In the example, the ARSLOAD program can locate the input file LETTERS.IND in the directory from which the program was started. The input file contains the index information that was created in “[Creating index data](#)” on page 86 (which is a parameter file for the Generic indexer). The image files to be loaded are in the same directory.

See [Chapter 16, “Commands overview,”](#) on page 113 for information about running the Content Manager OnDemand programs as commands (under the TSO OMVS command), about setting up JCL to run the Content Manager OnDemand programs from jobs and started tasks, and examples of how to specify parameters to the Content Manager OnDemand programs.

Processing the input data

The ARSLOAD program processes the parameter file, loads the index data into the database, and loads the image files on to storage volumes.

Processing index data

The ARSLOAD program processes the parameters that were specified on the command line and the indexing parameter file before loading the index data into the database. The ARSLOAD program extracts information from the application group and the application and performs the following processing:

- Extracts the database field information from the application group

- Extracts the data type and other information from the application
- Creates one database row for each group that was specified in the parameter file. Specified one group for each image file.
- Passes the index rows to the database manager. The rows contain the index values and other fields generated by Content Manager OnDemand. An index row contains:
 - One column for each field that was defined for the application group
 - One or more columns of Content Manager OnDemand control information

Processing the image files

The ARSLOAD program processes each image file that was specified in the parameter file. The ARSLOAD program extracts information from the application group and the application and performs the following processing:

- Extracts the compression information from the application
- Extracts the storage management information from the application group. The storage management information determines the storage locations, such as cache storage and archive storage.
- Stores the image files in storage objects. Content Manager OnDemand uses a 10 MB storage object (the default value) to improve storage efficiency and performance. Content Manager OnDemand assigns unique names to the storage objects and sequentially numbers them within an application group.

Verifying processing

The ARSLOAD program saves a copy of the messages that were generated during a load process in the system log. (Search for message number 87 in the system log.) After a load process completes, you can open the System Log folder and review the messages. The information in the messages includes the date and time that the load process started and completed, the name of the input file(s), and the number of rows that were added to the database. For example:

```

arsload: Processing file >letters<
arsload: Mon Dec 18 13:04:37 MST 1995 Indexing started,
        116017 bytes to process
        Content Manager OnDemand Load ID =
>6927-1-0-19950101000000-19950101000000-7606<
        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 rows that Content Manager OnDemand added to the database. In the example, the number of rows added to the database should equal the number of groups (and image files) that were specified in the parameter file.

The Content Manager OnDemand Load ID represents the data that the ARSLOAD program stored into the system during a load process. The Load ID can be used to identify a specific load process. For example, you can run the ARSADMIN program and specify the Load ID to delete the index data and documents that were created when the ARSLOAD program processed a Generic indexer parameter file.

Chapter 12. Loading user-defined data

About this task

Overview

Content Manager OnDemand provides support for storing and retrieving almost any type of data. For example, OnDemand provides support for AFP data, line data, PDF files, and image data (GIF, JFIF (JPEG), PCX, and TIFF). However, Content Manager OnDemand is not limited to maintaining these types of data. Content Manager OnDemand provides the User-Defined data type to support almost any other type of data that you want to store in the system. For example, you can configure the system to process Lotus® WordPro documents, so that when a user retrieves one of the documents from the system, Content Manager OnDemand automatically starts Lotus WordPro to open the document.

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 Content Manager OnDemand. The file type determines the program that is started on the client PC to open a file when a user retrieves one of the files from the system. The file type must also be registered with the client operating system. If your Content Manager OnDemand system supports client programs that run under different operating systems, then you must register the specified file type on all of the client operating systems.

Content Manager OnDemand provides the Generic indexer so that you can index user-defined data. You specify the index data for the input files that you want to index with the Generic indexer in a parameter file. The parameter file contains the index field names and values and identifies the input files that you want to process. You can create up to 128 index fields for each input file that you want to load into the system, providing many ways for users to query and retrieve documents. The number of index fields that you define usually depends on how your users retrieve documents from the system. For example, you might want to define index fields for the date, author, and subject or purpose of the user-defined data.

Before you can load user-defined data into the system, you must define an application group to manage the storage of the data, an application to specify the physical and logical attributes of the input files, and a folder to let users search for and retrieve the documents. This section contains an example that shows how to define Lotus WordPro files to Content Manager OnDemand. It provides an overview of defining the application group, application, and folder.

Defining the application group

When you load an input file into the system, Content Manager OnDemand updates the database with the index data that you provide and stores the indexed groups of pages as documents in cache storage and archive storage. The application group contains the information that Content Manager OnDemand uses to store and maintain the index data and the documents on the system. This section provides information about some of the key properties of the application group. You can use the Content Manager Content Manager OnDemand administrative client to define an application group.

Database organization

The Database Organization determines how Content Manager OnDemand organizes the index data that is stored in the application group.

Accept the default Database Organization of Multiple Loads per Database Table and Table Size of 10 million rows. Each time that a Lotus WordPro file is loaded into the system, Content Manager OnDemand adds one row to a database table. When a table reaches 10 million rows, Content Manager OnDemand closes the table and creates a new table. Content Manager OnDemand always adds index records to the open table; closed tables can be queried.

Expiration type

The Expiration Type determines how Content Manager OnDemand deletes index data and documents from the application group.

Accept the default Expiration Type of Load. This means that Content Manager OnDemand deletes the index data and documents from one load process (one or more input files) at a time from the application group. Depending on the number of input files processed during a load process, one or more Lotus WordPro files may be deleted at a time. For example, if you create a parameter file for the Generic indexer that contains index data for several input files, then Content Manager OnDemand will delete all of the index data and documents from the application group at the same time.

Permissions

Application group permissions determine the users that can access information stored in the application group and determine the users that can do other types of tasks related to the application group.

Under the *PUBLIC identifier, specify the Access permission so that all of the users defined to the library server can access data stored in the application group.

Field definition

The Field Definition page is where you define the database fields for the application group. When you load an input file into the system, Content Manager OnDemand stores the index data that is specified in the parameter file into fields in records that are added to the database. When a user queries the system, Content Manager OnDemand compares the search criteria entered by the user with index data in the application group.

Define the following database fields. The fields 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 input file. For example, the date that the file was created or the date that 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.

Defining the application

Most customers define an application for each different source of input data that they plan to load into the system. This section provides information about some of the key properties of the application. You can use the Content Manager Content Manager OnDemand administrative client to define an application.

Application Group

You must assign the application to an application group. Assign the application to the application group that was created in [“Defining the application group”](#) on page 91.

Data Format

The Data Format determines the kind of data that is found in the documents that are stored on the system. If the type of the data that is found in the original input file is different than the type of the data that is found in the documents that are stored on the system, then you should specify the type of the data that will be stored in Content Manager OnDemand.

Because there is not a supplied Data Format for Lotus WordPro files, select User Defined from the Data Type list.

File Extension

When you select User Defined from the Data Type list, you must also enter a value in the File Extension. The File Extension determines the program that is started by the client to open a user-defined document when it is retrieved from the system. For the example, enter the characters LWP, for Lotus WordPro.

The File Extension that you specify must also be registered on the client operating system. See your operating system information for help with registering file extensions.

Indexer

The Indexer determines the indexing program that Content Manager OnDemand uses to index and convert input data. Select Generic as the Indexer. To store user-defined files on the system, you must index them with the Generic indexer program that is provided with Content Manager OnDemand.

Data Compression

The Data Compression determines whether Content Manager OnDemand compresses the input files.

On the Load Information page, accept the default Data Compression of OD77, to compress the input files before storing them on the server.

Note: Resources are not supported for user-defined input data.

Defining the folder

You must define a folder so that users can search for and retrieve the input files that you load into Content Manager OnDemand. This section provides information about some of the key properties of the folder. You can use the Content Manager Content Manager OnDemand administrative client to define a folder.

Application Group

A folder can be used to search one or more application groups. Select the application group that was created in [“Defining the application group”](#) on page 91. When users open the folder, they can search for and retrieve the Lotus WordPro files that were loaded into the application group.

Permissions

Folder permissions determine the users that can open the folder and determine the users that can do other types of tasks related to the folder.

Under the *PUBLIC identifier, specify the Access permission so that all users defined to the library server can open the folder.

Folder fields

The Field Definition page is where you define the search fields for the folder. The search fields contain the search criteria entered by the user. For most folders, you probably want to define a search field for each field that you defined for the application group.

Define the following folder fields. The fields allow users to locate files based on different criteria, such as the date, the author, and the subject.

File Date

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

Author

The person that created the file.

Subject

The subject or purpose of the file.

Field Mapping

The Field Mapping page is where you *map*, or associate, the folder fields to the application group fields. This is how you specify that the search criteria that a user enters in a particular folder field should be used to search a specific application group field. Map each of the folder fields to their corresponding application group fields. For example, map the folder field named File Date to the application group field named fdate.

Accessing the input files

About this task

The ARSLOAD program runs on the Content Manager OnDemand server. Therefore, the files that you want to load into the system must be stored on the server.

You can transfer the files to the server using one of a number of methods. For example, use FTP to copy the files from the PC on which the files were created to the server on which the ARSLOAD program will run. Copy the files from the PC to the server in binary format with a file type of LWP.

Store the input files on the server in the location from which you plan to run the ARSLOAD program. Otherwise, specify the full path name of the input files in the Generic index file.

Creating the index data

About this task

You must create index data for your input files before you can load them into the system. If the input data is other than AFP, line, or PDF, then you must use the Generic indexer to load it into the system. You must specify index data for the Generic indexer using a parameter file. You should use a standard text editor to create the parameter file. [Figure 12 on page 95](#) shows an example of a parameter file that can be used by the Generic indexer to process the Lotus WordPro files.

See the *IBM Content Manager OnDemand Indexing Reference* for more information about the Generic indexer and the parameter file.

```
CODEPAGE:819
COMMENT:      input file number 1
GROUP_FIELD_NAME:fdate
GROUP_FIELD_VALUE:12/18/95
GROUP_FIELD_NAME:author
GROUP_FIELD_VALUE:Jessica Hawkins
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:optical storage devices
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter1.lwp
COMMENT:      input file number 2
GROUP_FIELD_NAME:fdate
GROUP_FIELD_VALUE:12/18/95
GROUP_FIELD_NAME:author
GROUP_FIELD_VALUE:Paul Garveys
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:optical storage devices
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter2.lwp
COMMENT:      input file number 3
GROUP_FIELD_NAME:fdate
GROUP_FIELD_VALUE:12/18/95
GROUP_FIELD_NAME:author
GROUP_FIELD_VALUE:Randy Perkinsen
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:account balance due
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter3.lwp
COMMENT:      input file number 4
GROUP_FIELD_NAME:fdate
GROUP_FIELD_VALUE:12/18/95
GROUP_FIELD_NAME:author
GROUP_FIELD_VALUE:Georgia July
GROUP_FIELD_NAME:subject
GROUP_FIELD_VALUE:airline fairs
GROUP_OFFSET:0
GROUP_LENGTH:0
GROUP_FILENAME:letter4.lwp
```

Figure 12: Example of a Generic indexer parameter file

Save the parameter file as LWP . IND on the server in the directory from which you plan to run the ARSLOAD program.

Configuring the ARSLOAD program

About this task

Note: The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD) must be running, otherwise the ARSLOAD program will fail.

The ARSLOAD program is the primary Content Manager OnDemand data indexing and loading program. The ARSLOAD program determines if the input data needs to be indexed, and if so, it calls the indexing program. The ARSLOAD program then processes the index data, loading it into the database, optionally compresses the input data into storage objects, and copies the storage objects to storage volumes. See

Chapter 24, “ARSLOAD,” on page 185 for more information about the ARSLOAD program, including the parameters that you can specify to process input data.

You typically run the ARSLOAD program each time that you want to load files into the system. You can either run the ARSLOAD program as a started task or from a job to periodically check for input data to process. To store a few WordPro files at a time, most customers choose to run the ARSLOAD program from the TSO OMVS command line. For example:

```
/opt/IBM/ondemand/V10.1/bin/arsload -u admin -p "" -n -g 'Lotus WordPro Documents'  
lwp
```

The following parameters were specified to the ARSLOAD program:

-u admin

The name of a Content Manager 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 Content Manager OnDemand user.

-n

Do not delete the input files.

-g 'Lotus WordPro Documents'

The name of the application group to load.

lwp

The name of the input file to process.

In the example, the ARSLOAD program can locate the input file LWP . IND in the directory from which the program was started. The input file contains the index information that was created in “[Creating index data](#)” on page 86 (which is a parameter file for the Generic indexer). Because the full path name of the input files was not specified, they must reside in the same directory.

See [Chapter 16, “Commands overview,”](#) on page 113 for information on running ARS programs as commands (under the TSO OMVS command) and on setting up JCL to run the ARS programs from jobs and started tasks. This chapter also has examples of how to input parameters to the ARS programs.

Processing the input data

The ARSLOAD program processes the parameter file, loads the index data into the database, and loads the Lotus WordPro files on to storage volumes.

Processing index data

The ARSLOAD program processes the parameters that were specified on the command line and the indexing parameter file before loading the index data into the database. The ARSLOAD program extracts information from the application group and the application and performs the following processing:

- Extracts the database field information from the application group
- Extracts the data type and other information from the application
- Creates one database row for each group that was specified in the parameter file. One group was specified for each Lotus WordPro file.
- Passes the index rows to the database manager. The rows contain the index values and other fields generated by Content Manager OnDemand. An index row contains:
 - One column for each field that was defined for the application group
 - One or more columns of Content Manager OnDemand control information

Processing the Lotus WordPro files

The ARSLOAD program processes each Lotus WordPro file that was specified in the parameter file. The ARSLOAD program extracts information from the application group and the application and performs the following processing:

- Extracts the compression information from the application
- Extracts the storage management information from the application group. The storage management information determines the storage locations, such as cache storage and archive storage.
- Stores the Lotus WordPro files in storage objects. Content Manager OnDemand uses a 10 MB storage object (the default value) to improve storage efficiency and performance. Content Manager OnDemand assigns unique names to the storage objects and sequentially numbers them within an application group.

Verifying processing

The ARSLOAD program saves a copy of the messages that were generated during a load process in the system log. (Search for message number 87 in the system log.) After a load process completes, you can open the System Log folder and review the messages. The information in the messages includes the date and time that the load process started and completed, the name of the input file, and the number of rows that were added to the database. For example:

```
arsload: Processing file >lwp<
arsload: Mon Dec 18 14:14:47 MST 1995 Indexing started,
        5116017 bytes to process
        Content Manager 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 >lwp<
```

You can verify the number of rows that Content Manager OnDemand added to the database. In the example, the number of rows added to the database should equal the number of groups (and Lotus WordPro files) that were specified in the parameter file.

The Content Manager OnDemand Load ID represents the data that the ARSLOAD program stored into the system during a load process. The Load ID can be used to identify a specific load process. For example, you can run the ARSADMIN program and specify the Load ID to delete the index data and documents that were created when the ARSLOAD program processed a Generic indexer parameter file.

Chapter 13. Restarting a load process

About this task

The ARSLOAD program will terminate if an unrecoverable error occurs during index, database, or storage manager processing. Termination processing includes setting a return code and saving error messages in the system log.

To start problem determination, open the System Log folder and view the messages that the ARSLOAD program generated during the load process. (Search for message number 87 in the system log.) The message log will contain normal processing messages, return codes, and error messages. See *IBM Content Manager OnDemand: Messages and Codes* for more information about messages.

If the ARSLOAD program failed during indexing, correct the problem and then restart the load process from the beginning. Common causes of problems during indexing include invalid input files or indexing parameter files and insufficient temporary space.

If the ARSLOAD program failed during database processing or storage manager processing:

- Determine and correct the problem.
- If a Load ID is listed in the message log that the ARSLOAD program saved in the system log, then you can use the ARSADMIN program to unload the data. See "Deleting a report" for information about unloading data from Content Manager OnDemand.
- Restart the load process from the beginning.

Chapter 14. Deleting a report

You can use the UNLOAD function of the ARSADMIN program to delete the index data and documents that the ARSLOAD program stored in the system during a load process. To use the UNLOAD function, you must specify the Load ID that the ARSLOAD program generated during the load process. The Load ID represents the index data that was added to the database and the storage objects that were copied to cache storage and archive storage. The Load ID can be found in the message log that the ARSLOAD program saved in the System Log.

When you use the UNLOAD function, you must specify the name of the application group and the Load ID. You can obtain the Load ID from the message log that the ARSLOAD program generated during the load process. If there is more than one Load ID listed in the message log, then you should use the last Load ID listed in the log. Storage locations include cache storage, the primary storage node, and the secondary storage node. The UNLOAD function deletes the storage objects associated with the Load ID. The UNLOAD function also deletes the rows of index data that were added to the database during the original load process.

The following shows an example of the program, path, and required parameter values to run the ARSADMIN program from the TSO OMVS command line on a z/OS server to delete a load process from all of the storage locations:

```
/opt/IBM/ondemand/V10.1/bin/arsadmin unload -g 'Credit' -L  
1220-1-0-8FAA-5538-5673
```

When the ARSADMIN program completes, you should open the System Log folder to view the messages that were generated during the UNLOAD process.

Note:

1. The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD) must be running, otherwise the ARSADMIN program will fail.
2. See [Chapter 17, “ARSADMIN,” on page 119](#) for more information about the ARSADMIN program and the parameters that you can specify to UNLOAD data from the system.
3. See [Chapter 16, “Commands overview,” on page 113](#) for information about running the Content Manager OnDemand programs as commands (under the TSO OMVS command), specifying JCL to run the Content Manager OnDemand programs from jobs and started tasks, and examples of how to specify parameters to the Content Manager OnDemand programs.

Chapter 15. Importing and exporting administrative objects through an XML interface

Overview

You can use an XML interface to import and export administrative objects from and into a Content Manager OnDemand system. The administrative objects that can be imported and exported include:

- systemParameters
- user
- group
- printer
- storageSet
- applicationGroup
- application
- folder
- cabinet
- hold
- odfRecipient
- odfRecipientList
- odfReportId
- odfDistribution with odfReportBundles

This XML interface expands the functionality and enables you to export all administrative objects into a single XML file, and later import them into the same OnDemand system or another system. Also, you can create an XML file from scratch through a user application or Web interface according to the defined specifications, and import it into the system.

When you create an XML import file, ensure that you include the administrative objects in this order:

1. systemParameters
2. user
3. group
4. printer
5. storageSet
6. applicationGroup
7. application
8. folder
9. cabinet
10. hold
11. odfRecipient
12. odfRecipientList
13. odfReportId
14. odfDistribution with odfReportBundles

For instructions about installing the administration interface, see "Installing the OnDemand administration interface" in the *Content Manager OnDemand for z/OS Configuration Guide*.

Files provided

The following files are provided in the `-PathPrefix- /opt/IBM/ondemand/V10.1/` directory:

- `arsxml`
- `xml/ODAdmin.jar`
- `xml/libarsxapi.so`
- `xml/ondemand.xsd` - OnDemand schema definition in UTF-8
- `xml/ondemand.ebcdic.xsd` - OnDemand schema definition in IBM-1047
- `xml/samples/addgroups.xml` - Sample to add groups in UTF-8
- `xml/samples/addgroups.ebcdic.xml` - Sample to add groups in IBM-1047
- `xml/samples/addusers.xml` - Sample to add users in UTF-8
- `xml/samples/addusers.ebcdic.xml` - Sample to add groups in IBM-1047
- `xml/samples/deletegroups.xml` - Sample to delete groups in UTF-8
- `xml/samples/deletegroups.ebcdic.xml` - Sample to delete groups in IBM-1047
- `xml/samples/deleteusers.xml` - Sample to delete users in UTF-8
- `xml/samples/deleteusers.ebcdic.xml` - Sample to delete users in IBM-1047
- `xml/samples/exportgroups.xml` - Sample to export groups in UTF-8
- ¹ - Sample to export groups in IBM-1047
- `xml/samples/exportusers.xml` - Sample to export users in UTF-8
- `xml/samples/exportusers.ebcdic.xml` - Sample to delete groups in IBM-1047
- `xml/samples/updategroups.xml` - Sample to update groups in UTF-8
- `xml/samples/updategroups.ebcdic.xml` - Sample to update groups in IBM-1047
- `xml/samples/updateusers.xml` - Sample to update users in UTF-8
- `xml/samples/updateusers.ebcdic.xml` - Sample to update users in IBM-1047

The XML parser included with IBM 64-Bit SDK for z/OS Java 2 can process both UTF-8 and EBCDIC (for example, IBM-1047).

However, be careful when you edit the UTF-8 files on z/OS to make sure that the files do not get saved in EBCDIC.

The samples reference the Content Manager OnDemand schema through `noNamespaceSchemaLocation=" ../ondemand.xsd"`. If you copy the samples to another directory, you might need to alter this specification to point to the location of `ondemand.xsd`.

Importing an XML file into a Content Manager OnDemand system

About this task

Importing an XML file that contains administrative objects into a Content Manager OnDemand system is a two-step process:

Procedure

1. Preparing an XML file for the import process

¹ `/xml/samples/exportgroups.ebcdic.xml`

2. Importing the XML file by using the ARSXML command

Preparing an XML file for the import process

About this task

Before importing data into a Content Manager OnDemand system, you need to have an XML file that contains all the data that needs to be imported. The XML file can be either created during a previous XML export process or written new. If you decide to develop your own XML file, you must follow the format of the Content Manager OnDemand XML schema file.

The Content Manager OnDemand XML schema file defines the syntactical format for all Content Manager OnDemand XML files, and is used during the import process to validate the contents of the import XML file. For different objects, the schema file specifies which fields are required and which fields are optional. Also, the schema file can establish a list of valid values for certain fields.

Creating an XML file

Occasionally, you need to create a Content Manager OnDemand XML file, and this section discusses the overall syntax and layout of the XML file. The XML syntax discussed in this section helps you understand XML to be able to construct the XML objects which are used by Content Manager OnDemand. For more detailed information on XML and specific XML syntax information, reference XML documentation.

Overall file structure

A Content Manager OnDemand XML file is a basic text file that can be created by the user.

Every Content Manager OnDemand XML file will contain the following:

- An XML identifier tag
- A Content Manager OnDemand identifier tag
- One or more Content Manager OnDemand objects
- A Content Manager OnDemand ending tag

XML identifier tag

The XML identifier tag specifies the version of XML that is used and the encoding that is used. The standard XML identifier is:

```
<?xml version="1.0" encoding="UTF-8"?>
```

Content Manager OnDemand identifier tag

The Content Manager OnDemand identifier tag specifies what the XML file is used for and what schema file to use. The standard Content Manager OnDemand identifier is:

```
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ondemand.xsd">
```

The name of the `ondemand.xsd` file, shown here as `"ondemand.xsd"`, should be changed to reference the Content Manager OnDemand XML schema file which is located in the Content Manager OnDemand `bin/xml` directory.

OnDemand objects

See [“Objects and data model used in the Content Manager OnDemand XML file”](#) on page 322 for objects and data model used in the XML file.

OnDemand ending tag

The Content Manager OnDemand ending tag indicates the end of the Content Manager OnDemand XML file. The ending tag is:

```
</onDemand>
```

Every standard Content Manager OnDemand XML file looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>

<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
    ... Content Manager OnDemand Objects ...
</onDemand>
```

Content Manager OnDemand objects

The basic building blocks for the XML file are referred to as objects.

The following objects can be included within the Content Manager OnDemand XML file:

- systemParameters
- user
- group
- printer
- storageSet
- applicationGroup
- application
- folder
- cabinet
- hold
- odfRecipient
- odfRecipientList
- odfReportId
- odfDistribution with odfReportBundles

These objects as well as all of the child objects are shown in detail in the XML data tables in [“XML objects in the Content Manager OnDemand XML file”](#) on page 322.

Ensure that you include the objects in this order:

1. systemParameters
2. user
3. group
4. printer
5. storageSet
6. applicationGroup
7. application
8. folder
9. cabinet
10. hold
11. odfRecipient
12. odfRecipientList
13. odfReportId
14. odfDistribution with odfReportBundles

Each object which is to be put into the XML file contains the following information:

1. A Start of Object tag.

2. A list of object attributes.
3. An optional list of child objects.
4. An End of Object tag.

Start of Object

An object is specified in the XML file by placing the object name after a < symbol. For example,

```
<user
```

starts a user object.

Important: The capitalization of the object names is important and should be used exactly as shown in the data tables in [“Objects and data model used in the Content Manager OnDemand XML file” on page 322](#).

Object attributes

All of the information about the object is contained within the object attributes. To add an attribute, specify the attribute name followed by an "=" symbol followed by the value of the attribute in double quotes. (Note: all attribute values need to be enclosed in a set of double quotes even if the value is a numeric value or a single character.) After all of the attribute values the '>' character is used to indicate the end of the object tag. For example,

```
<user name="SAMPLEUSER" phone="(212) 555-1212" timeOut="4" >
```

Indicates a user with a name of SAMPLEUSER, a phone number of (212) 555-1212, and a timeout value of 4 minutes. All of the attributes that can be specified for each object as well as the possible values and default values are shown in the XML data tables.

Important:

1. The capitalization of the object names is important and should be used exactly as shown in the data tables in [“Objects and data model used in the Content Manager OnDemand XML file” on page 322](#).
2. Unless you set Case Sensitivity as one of the system parameters, when you add a user, Content Manager OnDemand converts lowercase letters in the user ID to uppercase. You can type the user ID in uppercase, lowercase, or mixed case letters. In the above example, whether you enter the user ID as SAMPLEUSER, sampleuser, or SampleUser, Content Manager OnDemand automatically converts it to SAMPLEUSER.

Child objects

Child objects are constructed the same way as the nine main Content Manager OnDemand objects. All child objects must occur after the object to which they are associated with and before the end of the object tag. In this example, there are two user permission children for the SampleUser user.

```
<user name="SampleUser" phone="(212) 555-1212" timeOut="4" >  
<permission user="APP1" adminAuthority="Yes" />  
<permission user="APP2" adminAuthority="Yes" />
```

Important: If an object cannot contain any children, such as the permission objects shown above, it must be ended with a slash / and the > symbol.

See [“XML objects in the Content Manager OnDemand XML file” on page 322](#) for detailed information about the types of child objects that each object can have and, in some cases, the maximum number of child objects that can be created.

Some of the child objects themselves can have children. These are defined in the same way as above. See the second example in the Examples section below.

End of object

The last item the user object needs is the end of object tag. This is indicated by placing the name of the object between </ and >. For example,

```
</user>
```

indicates the end of the user object.

As with any object if the object does not contain child objects, the object can be ended by placing a /> at the end of the object definition. So the following,

```
<group name="Sample" gid="1084000">
</group>
```

is equivalent to:

```
<group name="Sample" gid="1084000"/>
```

Examples

The following is a complete Content Manager OnDemand XML file that contains two users and a user group.

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ondemand.xsd">

  <user
    name="SampleUser1" email="Sample@us.ibm.com" printer="Sample"
    userType="User Admin"
    createFoldersAuth="Yes">
    <permission user="SampleUser2" adminAuthority="Yes" />
    <permission user="SampleUser3" adminAuthority="No" />
    <permission group="SampleGroup1"/>
  </user>

  <user
    name="SampleUser2" password="xxxxxxx" timeOut="No Limit"
    description="This is a description of Sample User Two">
  </user>

  <group
    name="SampleGroup1"
    description="A Sample Group Number One"
    ownerUser="Admin">
    <user name="SampleUser2"/>
    <user name="SampleUser3"/>
  </group>

</onDemand>
```

The following example shows a folder with a field child and the field child has a fieldInfo child.

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <folder name="FolderOne" searchType="Hit List">
    <field name="cost" fieldType="Decimal" >
      <fieldInfo user="SampleUser" sortOrder="2" greaterThan="Default"
lessThan="Yes"/>
    </field>
```

```
</folder>
</onDemand>
```

Creating Content Manager OnDemand XML files for update

When creating an XML file to be used for updating information, the syntax and layout of the file are the same as discussed in the previous section. However, there are two major differences.

- Only the fields which are intended to be updated need to be specified.
- Child objects might contain an additional attribute, `task`. The `task` attribute indicates the task that is performed by the child object.

Each object that you update must contain the name of the object. Any other attributes that are specified are updated to the specified value. If an object that is updated requires a change to another object, that object must be included in the XML file before the object that references it. For example, to change the user `SampleUser` to timeout after 10 minutes, you need to put the following XML code into an XML file.

```
<user name="SampleUser" timeOut="10"/>
```

When you specify some special characters in XML, you need to use the XML specification. For example:

<i>Table 6: XML specification for special characters</i>	
XML code	Character
&	&
'	'
"	"
<	<
>	>

Your input file might look like this:

```
...
<user
  name="DBRYANTDEU"
  acctInfo="Engraving & Printing"
```

To rename some objects, you should use the `newName` attribute. For example, to change the name of a printer from Boston to New York, use the following XML code during an update:

```
<printer name="Boston" newName="NewYork"/>
```

Most objects might contain a set of similar child objects. To update these objects, you should use the `task` attribute. The `task` attribute can have three values:

add

Indicates that the child object should be added to the parent object. The default value for the `task` attribute is `add`. However, some child objects do not contain a `task` attribute. For those objects without a `task` attribute, the action taken for the object is the same action as the parent object.

update

This indicates that the child object already exists and should be updated with the attributes provided.

delete

This indicates that the child object already exists and should be removed from the parent object.

The `task` attribute is only examined during an update process. To add, update or delete a child object, the parent object must be specified, and must contain the child object to be updated. See the examples below.

The default value for the task attribute is "add". However, some child objects do not contain a task attribute. For those objects without a task attribute, the action taken for the object will be the same action as the parent object.

If an object that is updated requires a change to another object, that object must be included in the XML file before the object that references it.

Examples

To add a permission for the user SampleUser to the folder FolderOne, the following code could be used during an update:

```
<folder name="FolderOne">
<permission user="SampleUser" adminAuthority="Yes" maxHits="No Limit" />
</folder>
```

To remove the admin authority permission from SampleUser on the folder FolderOne, you need to update the permission child. The following code is an example:

```
<folder name="FolderOne">
<permission task="update" user="SampleUser" adminAuthority="No" />
</folder>
```

To remove the SampleUser permission from the folder, you need to delete the permission child.

```
<folder name="FolderOne">
<permission task="delete" user="SampleUser"/>
</folder>
```

Creating Content Manager OnDemand XML files for delete and export

When creating a Content Manager OnDemand XML file to be used to export or delete objects, the syntax and layout of the file is the same as discussed above, however there are two major differences:

- The only attribute field which is examined is the name field.
- All child objects are ignored.

When you export or delete objects, the only information that is required is the name of the object(s). Other attributes might be present in the XML file, but they are ignored. Child objects cannot be deleted through the delete process, they can be deleted only during an update process by using the attribute task="delete".

When you specify an application for delete or export, you must also specify the name of the application group in which the application is contained.

To export the users, SampleOne, SampleTwo and SampleThree, the following Content Manager OnDemand XML file can be used during an export:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ondemand.xsd">

  <user name="SampleOne"/>
  <user name="SampleTwo"/>
  <user name="SampleThree"/>
</onDemand>
```

To delete the printer, OldPrinter, and the folder, MyFolder, use the following XML code:

```
<printer name="OldPrinter"/>
<folder name="MyFolder"/>
```

To delete all defined printers, export all the printers by using the following XML code and command:

```
<printer name="_ALL"/>

arsxml export -h <host> -u <uid> -p <password> -i input.xml -o
allprinters.xml
```

Then use the output from this command to feed the delete command:

```
arsxml delete -h <host> -u <uid> -p <password> -i allprinters.xml
```

When you specify an application for delete or export, the name of the application group in which it is contained must also be specified.

Application Index Parameter Object

The `indexParm` child object of the application object has a slightly different syntax than all other objects. Due to the nature of the data associated with the Index Parameters it is not appropriate to use an object attribute to specify this information. Therefore, for this object (and only this object), the data associated with the object is simply the character data which appears between the `<indexParm>` and the `</indexParm>` flags.

For example, the following code can be used to set the index parameters associated with the SampleApp application:

```
<application name="SampleApp" .... >
<indexParm>
*dummy index parameters
parm1=value;
parm2=value;
parm3=value;
</indexParm>
</application>
```

Importing the XML file by using the ARSXML command

About this task

Use the ARSXML command to import the XML file into the Content Manager OnDemand system. For syntax, description, and parameters of this command, see [“Purpose” on page 231](#).

Exporting Content Manager OnDemand administrative objects to an XML file

About this task

You can use the ARSXML command to export objects from Content Manager OnDemand into an XML file in several ways:

- Export a single object, such as a single user
- All of the defined objects of a particular type, for example, all user groups
- Any combination of objects, for example, a user group and all users in that group

This feature can be used to backup part of a Content Manager OnDemand system, copy objects from one system to another, or store objects into an XML file so that they can be processed by another application.

Use the ARSXML command to export administrative objects into an XML file. For syntax, description, and parameters of this command, see [“Examples” on page 237](#).

If you are exporting objects that have dependencies on other objects: Some objects in Content Manager OnDemand might have dependencies on other objects, for example, a group object has a dependency on all of the users that are defined within the group. When you export these objects, you can use the `—r` parameter with a value of `d` to include in the XML file all of the dependent objects that the exported objects might have.

See [“Examples” on page 237](#) for more example XML files that are generated in different export scenarios.

Chapter 16. Commands overview

The commands contained in this reference are Content Manager OnDemand server commands and commands that you can use to control the archive storage manager and the database manager. The commands are performed by a set of utility programs which may be executed in a variety of ways:

- Under USS (UNIX System Services)
 - Run from TSO command line using OMVS command
 - Invoked by another program
 - Run as a shell script
- Under native z/OS
 - Run as a started task
 - Invoke by a batch job JCL EXEC statement as the entry program in a job step
 - Run from a batch job as a called program in a job step

IBM recommends that all commands be run under a started task.

The ARS utility programs require that the procedure ARSSOCKD be running as a started task before they are invoked; ARSSOCKD executes the server. This procedure was customized to run in your operating environment when OnDemand was installed.

In the following chapters the utility programs are presented in alphabetical order. Each chapter contains a description of a utility program's purpose, syntax diagrams for each function that program performs, and descriptions of the parameters which can be used to enter commands to that program. Examples and general information about using the commands have also been included.

Running Content Manager OnDemand server commands

About this task

The examples below show some of the ways that the ARSLOAD utility program can be executed. They illustrate how the same path, required parameter values and optional parameter values are entered to the z/OS system using the different methods of execution. The examples in the chapters which follow will present only the syntax for the path, required parameter values and optional parameter for the purposes of illustrating their use for each utility program. Variation for method of execution as illustrated below is not factored into further examples.

When running commands by using JCL, it may be necessary to set environment variables to control program processing. For jobs that use BPXBATCH, the STDENV DD statement provides a means of setting environment variables. See *z/OS UNIX System Services Command Reference*, SA22-7802 for more information.

For JCL that does not use BPXBATCH, the Language Environment® ENVAR runtime option provides a means of setting environment variables. See *z/OS Language Environment Programming Reference*, SA22-7562 for more information. Additional environment variables can be set from an MVS data set or HFS file by using the `_CEE_ENVFILE` environment variable. See *z/OS V1R1.0 C/C++ Programming Guide*, SC09-4765 for details about the `_CEE_ENVFILE` environment variable. Note that for an MVS data set, the data set must be allocated with RECFM=V.

Important: Examples of commands may appear in one of the following two formats:

```
/usr/lpp/ars/V10R1M0/bin/arsmaint -cdmsv >> /tmp/arsmaint.log 2>&1
```

```
/usr/lpp/ars/V10R1M0/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.

Procedure

1. To run under USS from TSO using the OMVS command:
 - a) In a TSO session, enter OMVS into the command line and hit enter.
 - b) In the next screen, enter the parameter values into the command line: `"/usr/lpp/ars/V10R1M0/bin/arsload -u USERID -p PASSWORD -d d1 -h systemid -C Q"`
2. To run as a started task, enter the following command on the SDSF command line: `/S ARSLOAD`

The following job stream will be submitted to the z/OS system:

```
//ARSLOAD JOB MSGLEVEL=1  
//STARTING EXEC ARSLOAD
```

and the following procedure will be executed:

```
//ARSLOAD PROC  
//ARSLOAD EXEC PGM=ARSLOAD,REGION=0M,TIME=NOLIMIT,  
// PARM=('/-h ARCHIVE -C Q')  
//STEPLIB DD DISP=SHR,DSN=ARS.V10R1M0.SARSLOAD  
// DD DISP=SHR,DSN=DB2.V10R1M0.SDSNEXIT  
// DD DISP=SHR,DSN=DB2.V10R1M0.SDSNLOAD  
// DD DISP=SHR,DSN=APK.ACIF231.SAPKMOD1  
//*****  
//SYSPRINT DD SYSOUT=*,RECFM=FBA,LRECL=121,BLKSIZE=6050  
//SYSOUT DD SYSOUT=*  
//*****  
//* The following 3 DD statements should be uncommented and  
//* customized if the PDF indexer is used.  
//*****  
//*ADOBERES DD DSN=ADOBE.PDFLIB.RESOURCE.INDEX(ADOBERES),DISP=SHR  
//*ADOBEFNT DD DSN=ADOBE.PDF405.PLUSP1C.ADOBEFNT.LST,DISP=SHR  
//*TEMPATTR DD DSN=ADOBE.PDF405.PLUSP1C.TEMPATTR,DISP=SHR
```

3. To invoke a batch job JCL EXEC statement, use the sample JCL.

```
//STEP1 EXEC PGM=ARSLOAD,REGION=0M,  
// PARM=('ENVAR("_CEE_ENVFILE=DD:MYVARS")/-u user -p password -h ARCHI+  
// VE -C R ')  
//STEPLIB DD DISP=SHR,DSN=ARS.V10R1M0.SARSLOAD  
// DD DISP=SHR,DSN=DB2.V10R1M0.SDSNEXIT  
// DD DISP=SHR,DSN=DB2.V10R1M0.SDSNLOAD  
// DD DISP=SHR,DSN=apk.acif231.SAPKMOD1  
//SYSPRINT DD SYSOUT=*  
//CEEDUMP DD SYSOUT=T  
//MYVARS DD DUMMY
```

Reading a syntax diagram

About this task

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) non italicized. Keywords are case sensitive, for example: -G is a different keyword with different effects than -g.
- Variables represent names or values you supply and are shown (in syntax) in italics. An example of a variable is *filename*. You would substitute an actual file name for filename. The values you supply might be case sensitive. A file name of /u/user1/foo is a different file name than /u/user1/F00.

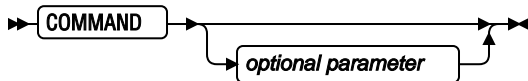
The command names, when specified in a UNIX System Services (USS) shell, are case sensitive and must be lowercase. For commands that can be specified in JCL, the command must be in uppercase. For example: // EXEC PGM=ARSL0AD.

A parameter can be a combination of a keyword and a variable.

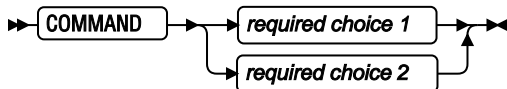
Required parameters are displayed on the main path.



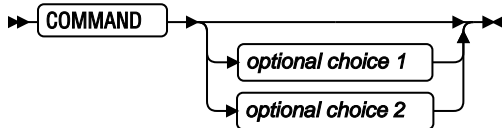
Optional parameters are displayed below the main path.



A stack of parameters, with the first parameter displayed on the main path, shows that you must choose one of the parameters.

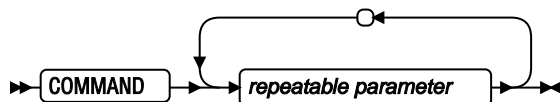


A stack of parameters, with the first parameter displayed below the main path, shows that you can choose one of the parameters.



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.

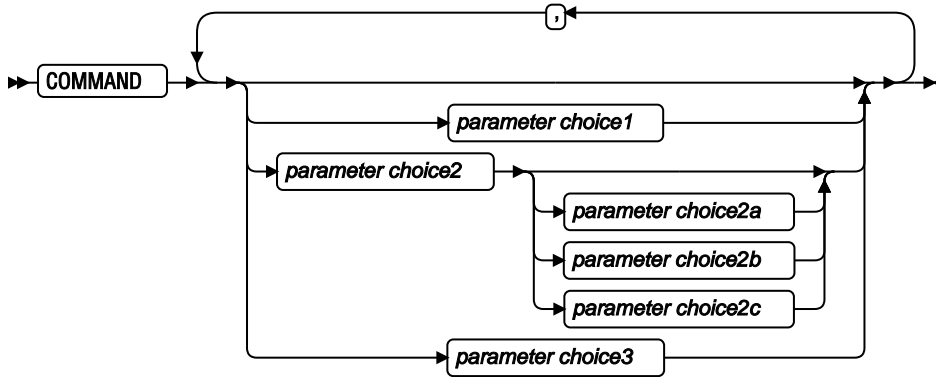


- If the repeat arrow contains a comma, the item can be repeated in a list with the items separated by commas.

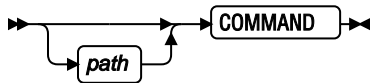


You can repeat items from parameter stacks following the stack conventions for required and optional parameters described previously.

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.

Using quotes in commands

About this task

When you run a Content Manager OnDemand program from the command line and you specify parameter values that contain a null (blank) character or some other special character (such as the parenthesis), if the command line interpreter of the operating system from which you are running the command requires the quoting of special characters, then you must delimit the parameter value with double quotation mark characters.

For example, when running the ARSDOC GET program from the command line under a USS shell and you specify the `-o` parameter to specify one or more database field names, then in addition to delimiting each field name with parenthesis, you must also delimit the entire parameter value with double quotation mark characters. For example:

```
-o "(sdate)(student)"
```

See your operating system documentation for more information about quoting and running commands.

Note: There are some exceptions to this rule, where quotes might always be required. See the specific commands in this reference to identify specific instances where parameters and values may always require quoting.

Omitting a userid and password

If the installation has enabled unified login with the ARS.PTGN dynamic exit and you omit the parameters specifying a userid and password, you may not be prompted for a userid and password. Instead, your userid (as returned by UNIX System Services) will be used to log on to Content Manager OnDemand. If that userid cannot be used, you will then be prompted for a userid and password.

For information about enabling unified login and the ARS.PTGN dynamic exit, see *IBM Content Manager OnDemand for z/OS: Configuration Guide*.

Removing trailing blank characters

About this task

Trailing blank characters are trimmed from the name when adding or updating the following objects:

- application
- application group
- folder
- hold
- printer
- storage set
- cabinet

As a reminder, for existing objects, the name specified to a command line program must include trailing blank characters that are part of the name.

Using the h and I parameters

About this task

The Content Manager OnDemand programs use the `-h` and `-I` parameters to determine the name of the Content Manager OnDemand instance to process. You must specify the parameter and name the instance if any of the following are true:

- The name of the default instance is `archive` and you are using a port number other than the default port number (1445).
- The name of the default instance is not `archive`.
- You are running more than one instance on the same workstation and you want to process an instance other than the default instance.
- You are running the program from an object server that resides on a different system or node than the library server.

The Content Manager OnDemand programs locate the specified instance name in the ARS.INI file to determine the TCP/IP address, host name alias, or fully-qualified host name of the system on which the Content Manager OnDemand library server is running and other information about the instance. The ARSADMIN, ARSDOC, and ARSLOAD programs support the `-h` parameter. The ARSDB, ARSLOAD, ARSMAINT, and ARSTBLSP programs support the `-I` parameter. For the ARSLOAD program, if both the `-h` and `-I` parameters are specified, the value of the last parameter specified is used.

Chapter 17. ARSADMIN

Purpose

The ARSADMIN program provides functions that can help you manage storage objects, index data, and specific documents in storage objects.

Description and syntax

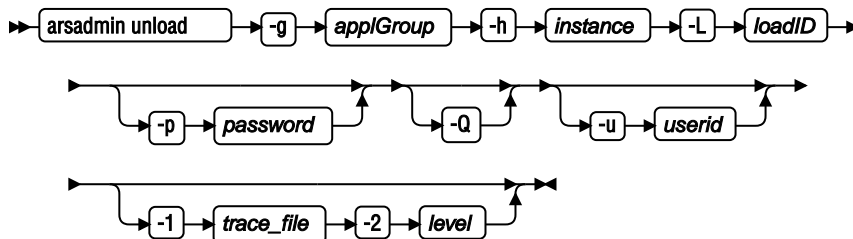
The ARSADMIN program provides the following functions:

Unload

The Unload function can be used to delete the storage objects and index data that were loaded into the system during input processing. To use the Unload function, you must specify a load ID. The Load ID represents the database rows and storage objects that were created when an input file was loaded into the system.

To get a load ID, in the Windows client, right-click a report and get a partial load ID, then search in the system log for an 87 message to find the full load ID.

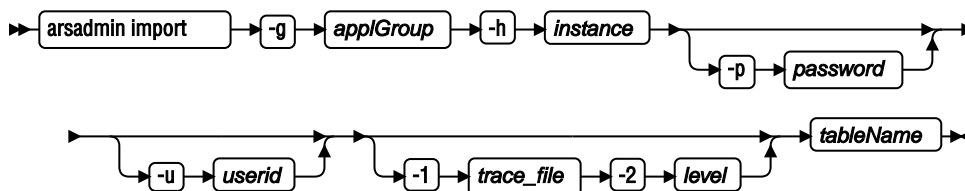
You would typically use the Unload function to manually remove an input file (report) from the system. You can also use the Unload function to delete data that was created during testing. You must specify the user ID of a user that is permitted to delete documents from the application group. The ARSADMIN program searches the system log for the specified Load ID. If the Load ID is not found in the system log, the unload will fail unless you specify the -Q parameter.



Import

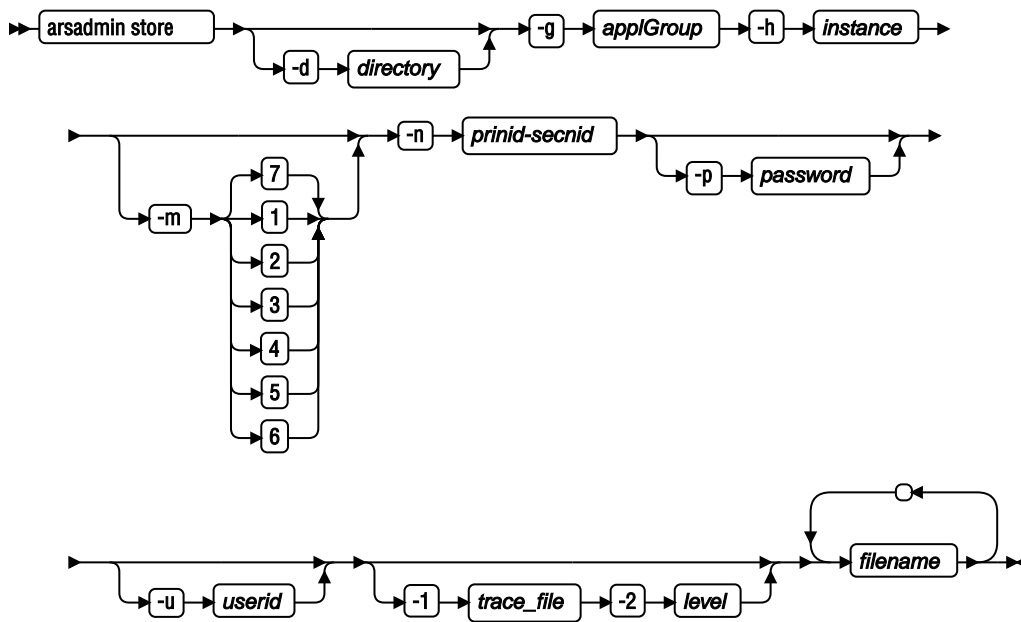
The Import function can be used to import migrated index data from archive storage to the database. When Content Manager OnDemand determines that a query requires index data that has been migrated to archive storage, then a message is sent to STDOUT and saved in the system log. An administrator must use the ARSADMIN program to import the required table or tables into the database. After the index data has been imported, the user can try the query again.

Note: Importing migrated index data requires additional space in the database, additional log file storage, and additional temporary space.



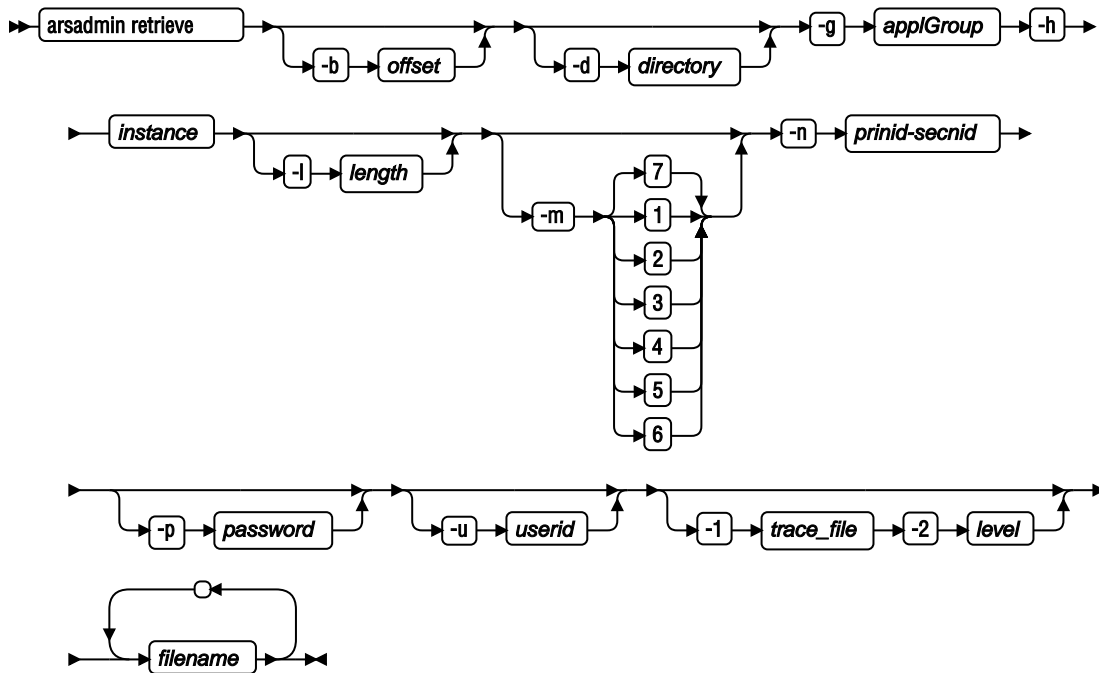
Store

The Store function can be used to copy storage objects from one storage location to another. For example, you could copy a storage object from cache storage to a primary storage node.



Retrieve

The Retrieve function can be used to retrieve a storage object from a specified location. For example, you could retrieve a storage object from cache storage or a primary storage node. After retrieving a storage object, you could use the Decompress function to extract a specific document from the storage object file.



Compress

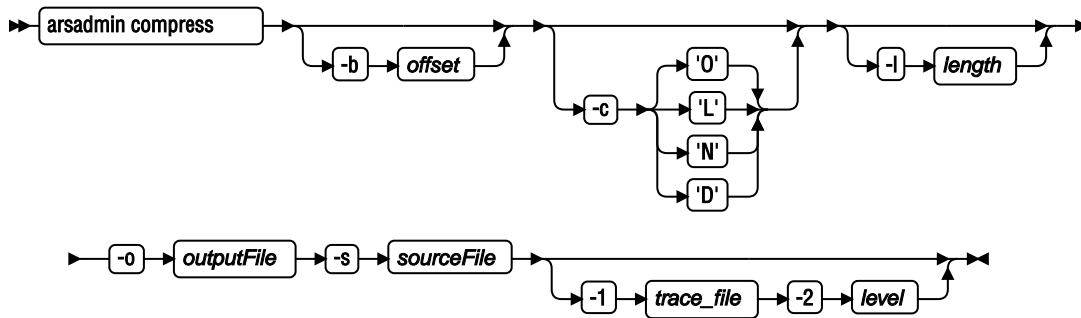
The Compress function is used to compress input files in preparation for creating storage objects.

The Compress function can also be used to estimate the amount of storage space required to store an input file in the system:

- To estimate the amount of storage required to load an input file that contains logical items, such as bills, invoices, or statements, first extract a representative sample of data from an input file. Then compress the sample with the -1 parameter. Divide the size of the compressed output file by the number of bytes of input data that you compressed. The result is the compression ratio achieved.

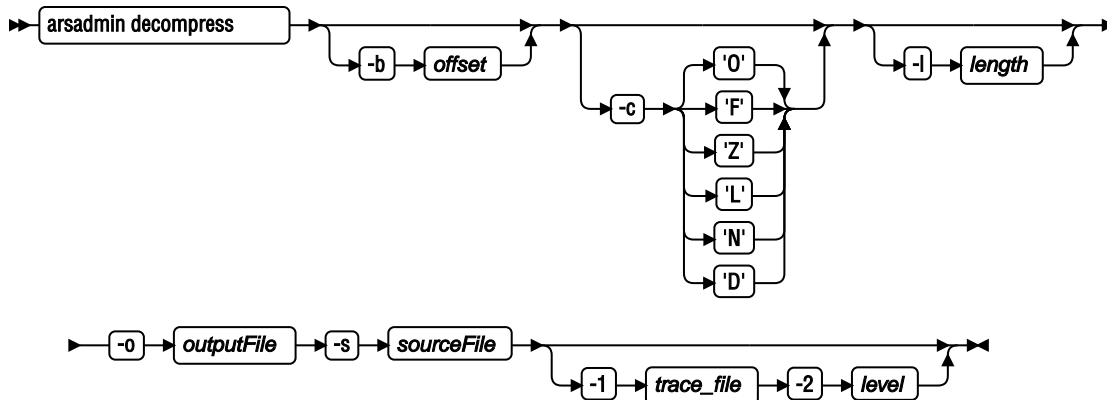
- To estimate the amount of storage required to store an input file that contains transaction data, first extract a representative sample of data from an input file. Then compress the sample with the `-l` parameter. Divide the size of the compressed output file by the size of the input file. The result is the compression ratio achieved.

To determine the best compression method for a particular type of input data, IBM recommends that you compress a sample of the data using each of the four compression methods provided by Content Manager OnDemand (LZW12, LZW16, OD77, OD77Lite, OD77HW, and OD77LiteHW). Compare the compressed file sizes and the time required to compress the data to determine the best compression method for the input data.



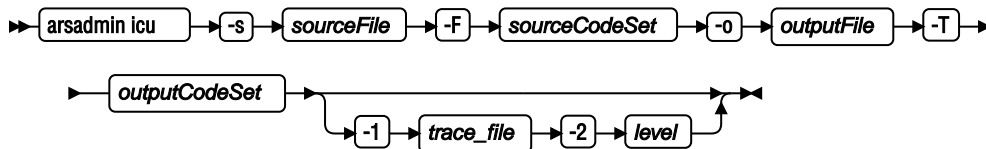
Decompress

The Decompress function can be used to extract a specific document from a storage object file. The document can then be sent to a printer or viewed with the appropriate viewing program. You must first create the storage object file with the Retrieve function. Storage objects whose applications are defined to use Large Object cannot be decompressed by using this command.



ICU

The ICU function converts a document from one code page to another code page. The `icu` function would be used if a document is stored in OnDemand in one code page and then is retrieved and uncompressed on a system that uses a different code page than what the document was stored in; the file would then need to be converted.



Parameters

-b offset

For the RETRIEVE function, the offset (in bytes) into the specified storage object where the ARSADMIN program begins a partial storage object retrieval. The default is the first byte in the storage object.

For the COMPRESS function, the offset (in bytes) from the beginning of the file where the ARSADMIN program begins data compression. The default is 0 (zero).

For the DECOMPRESS function, the offset (in bytes) from the beginning of the file where the ARSADMIN program begins data decompression. The default is 0 (zero).

-c type

For the COMPRESS and DECOMPRESS functions, the compression type for source files. Options are 'F' for OD77Lite, 'O' for OD77 compression (the default), 'L' for LZW12 compression, 'Z' for LZW16 compression, 'N' for no compression, and 'D' for disable compression. Storage objects whose applications are defined to use Large Object cannot be decompressed by using the DECOMPRESS function.

-d directory

For the STORE function, the name of the directory that contains the storage object file to load. The default is the directory from which the ARSADMIN program was invoked.

For the RETRIEVE function, the directory in which the ARSADMIN program writes the file that contains the storage object retrieved from Content Manager OnDemand. The default is the directory from which the ARSADMIN program was invoked.

-F

For the ICU function, specifies the code set for the document specified by the -s parameter.

filename

For the STORE function, the name of the storage object file (or files) to load. To specify more than one storage object file, separate the file names with the blank character.

For the RETRIEVE function, the name of the storage object or objects to retrieve. To specify more than one storage object file, separate the file names with the blank character. Also determines the name of the file that the ARSADMIN program creates in the directory specified with the -d parameter.

-g applGroup

The name of the application group from which Content Manager OnDemand extracts processing information and stores the data.

-h instance

The name of the Content Manager OnDemand instance to process. The ARSADMIN program will attempt to locate the specified instance name in the ARS.INI file to obtain the TCP/IP address, host name or host name alias of the workstation or node on which the instance is running. If the ARSADMIN program cannot locate the instance name in the ARS.INI file, the specified value is treated as a host name.

The -h parameter is required for all functions, except COMPRESS and DECOMPRESS.

If you are running multiple instances of Content Manager OnDemand on the same workstation or node, then you should always specify the -h parameter to identify the name of the instance that you want to process. You should also verify that the ARS.INI file is configured with the correct information for all instances of Content Manager OnDemand. See the *IBM Content Manager OnDemand for z/OS: Configuration Guide* for more information about working with multiple instances of Content Manager OnDemand.

-l length

For the RETRIEVE function, the number of bytes that the ARSADMIN program retrieves from the specified storage object. The -l parameter and the -b parameter provides support for partial storage object retrieval.

For the COMPRESS function, the number of bytes (from the offset) of data to compress. The default is zero bytes, meaning all the remaining bytes in the file.

For the DECOMPRESS function, the number of bytes (from the offset) of data to decompress. The default is zero bytes, meaning all the remaining bytes in the file.

-L loadID

Identifies the value that Content Manager OnDemand generates to uniquely identify an input file that was loaded into the system. The Load ID contains the following parts:

```
agid-pri_nid-sec_nid-docid-start_date-stop_date-aid
```

The following list describes each part:

agid

Application group identifier.

docid

Document identifier.

pri_nid

Primary storage node.

sec_nid

Secondary storage node.

start_date

First date in the report. The format of the date is YYYYMMDDHHMMSS. Loads created with versions of Content Manager OnDemand prior to version 9.0 use an older, internal date format for this part.

The value of this part can also be 0.

stop_date

Last date in the report. The format of the date is YYYYMMDDHHMMSS. Loads created with versions of Content Manager OnDemand prior to version 9.0 use an older, internal date format for this part.

The value of this part can also be 0.

aid

Application identifier. This part is optional.

If you redirected the output of the ARSLOAD program or the Load function to a log file, see the log file for the Load ID that was created during load processing.

For the UNLOAD function, the Load ID that you want to delete. When recovering from a system failure or some other problem that occurred during input processing, you should delete the last (or only) Load ID that is listed in the log file.

In the Windows client, you can right-click and get a partial load ID, and then check the system log for an '87' message to find the full load ID.

-m location

Determines the location(s) of the storage objects that are processed by the storage manager. Possible locations are cache storage, the primary storage node, and the secondary storage node. The default value is 7, which specifies that the storage manager should process the storage objects for all of the locations that are specified in the application group. Specify 1 for cache storage, 2 for the primary storage node, or 4 for the secondary storage node. The values are additive; that is, if you specify a value of 3, then the storage manager processes the storage objects in cache storage and the primary storage node.

-N

For the UNLOAD function, allows you to attempt to unload data even if the Load ID is not present in the system log.

-n *prinid-secnid*

The primary and secondary storage node identifiers for the application group. Separate the identifiers with the - (dash) character.

Content Manager OnDemand stores the primary and secondary storage node identifiers in the database when you load data into an application group. Content Manager OnDemand includes the identifiers in the Load ID.

The values specified with the -n and -g parameters enable the ARSADMIN program to store data on or retrieve data from a specific Content Manager OnDemand object server.

-o *outputFile*

For the COMPRESS function, the name of the compressed output file.

For the DECOMPRESS function, the name of the file that contains the uncompressed document.

For the ICU function, specifies the name of the file that will contain the document after it has been converted to the code set specified by the -T parameter.

For the Trace function, specifies the name of the output XML file.

-p *password*

Specify one of the following options for *password*:

- The name of the stash (encrypted password) file that contains the password for the user ID specified with the -u parameter.
- The password for the Content Manager OnDemand user ID specified with the -u parameter. If the user ID is not assigned a password, then enter a null password (that is, specify -p ""). If you do not specify the -p parameter while you specify the -u parameter, then the program retrieves the password for the user ID from the stash file specified in ARS.INI file for that instance. If the user ID is not assigned a password, then press the Enter key when prompted.

-Q

For the UNLOAD function, allows you to force the ARSADMIN program 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 program does not find the Load ID in the system log and you do not specify the -Q parameter, then the Unload function will fail.

-s *sourceFile*

For the COMPRESS function, the name of the uncompressed input file.

For the DECOMPRESS function, the name of the storage object file that contains the document or resource to be uncompressed. You cannot decompress Large Object data with the Decompress function.

For the ICU function, specifies the name of the file that contains the document that is in the code set specified by the -F parameter.

For the Trace function, specifies the name of the binary file that is generated by any Content Manager OnDemand command.

-T

For the ICU function, specifies the code set for the document specified by the -o parameter.

-u *userid*

A Content Manager OnDemand user ID with administrator authority for the application group. If you do not specify this parameter, then the ARSADMIN program prompts you to enter the user ID.

For the UNLOAD function, the user ID that you specify must have permission to delete documents from the application group.

If you do not specify the -p parameter with the -u parameter, the ARSADMIN program retrieves the password for the user ID from the ARS.INI file for that instance.

tableName

For the IMPORT function, the name of the migrated index table to import into the database.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

1. The following example shows how to use the UNLOAD function to delete an input file from the system.

```
arsadmin unload -g 'Credit' -L 1220-1-0-1FAA-9339-9345
```

2. The following example shows how to use the COMPRESS function to estimate the amount of storage required for an input file that contains logical items, such as bills, invoices, or statements. The ARSADMIN program 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
```

3. The following example shows how to use the COMPRESS function to estimate the amount of storage required for an input file that contains transaction data, such as a general ledger. The input file is a portion of a larger input file.

```
arsadmin compress -s report_file_segment -o output_file
```

4. The following example shows how to use the IMPORT function to import a migrated index table from archive storage into the database. The name of the application group and the index table to import can be obtained from the message that OnDemand saves in the system log.

```
arsadmin import -g 'Credit' AAA27
```

5. The following example shows how to use the ICU function to convert the input .utf8 file from Unicode to EBCDIC and save the output in the output .ibm500 file.

```
arsadmin icu -s input.utf8 -F UTF-8 -o output.ibm500 -T IBM-500
```

Notes

The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD) must be running, otherwise the ARSADMIN program will fail.

Files

/usr/lpp/ars/V10R1M0/bin/arsadmin

The z/OS executable program.

Chapter 18. ARSAFPD

Purpose

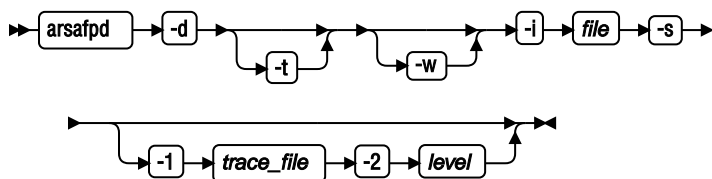
The ARSAFPD command displays information about input files that are loaded into Content Manager OnDemand.

The command also displays any Advanced Function Presentation (AFP) readable file. The AFP file can be a document, index or resource file.

ARSAFPD command interprets a subset of valid structured fields only. Structured fields that are not interpreted are displayed as

```
nnn      ??? Unknown Structured Field
```

Syntax



Remember: The -d option and the -s option cannot be used at the same time.

Description

The ARSAFPD command displays information about the format of an input file, or displays an Advanced Function Presentation (AFP) readable file.

Parameters

- d**
Displays an Advanced Function Presentation (AFP) readable file.
- i file**
The input file name.
- s**
Display information about the format of the file.
- t**
Display information about AFP triplets. This option displays more detail about the AFP file. This parameter can be used with the -d parameter only.
- w**
Disables warning messages. The arsa.fpd command issues warning messages if the AFP file is not in the correct format for loading into Content Manager OnDemand. This parameter can be used with the -d parameter only. The arsa.fpd command displays only up to 10 warning messages at one time.
- 1 trace_file**

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

Display file format information

For example, if you use the `arsafpd -s -i \acif\input\loan10k.report` command to display information, the output could be:

```
File is ASCII line data.  
File contains hex 0D0A delimiters.  
Carriage control type is ANSI (ASCII).
```

Display AFP file format information

For example, if you use the `arsafpd -s -i \acif\input\test.afp` command to display AFP format information, the output could be:

```
File is AFP.  
File contains group TLE structured fields.
```

AFP output

For example, if you use the `arsafpd -d -i \acif\input\com.afp` command to display AFP information, the output could be:

```
1 BDT Begin Document                                0070 D3A8A8  
2   BNG Begin Named Page Group 00000001           002A D3A8AD  
3     TLE Tag Logical Element                     0026 D3A090  
4     TLE Tag Logical Element                     002B D3A090  
5     TLE Tag Logical Element                     0029 D3A090  
6     TLE Tag Logical Element                     0039 D3A090  
7     TLE Tag Logical Element                     002C D3A090  
8     TLE Tag Logical Element                     002C D3A090  
9     TLE Tag Logical Element                     0043 D3A090  
10    BPG Begin Page 00000001                     0029 D3A8AF  
11      BAG Begin Active Environment Group         0008 D3A8C9  
12        MCF2 Map Coded Font2                    0033 D3AB8A  
13        NOP No Operation                        0008 D3EEEE  
14        PGD Page Descriptor                     0016 D3A6AF  
15        PTD2 Presentation Text Desc2            0016 D3B19B  
16      EAG End Active Environment Group           0008 D3A9C9  
17      BCT Begin Composed-Text Block              0008 D3A89B  
18      PTX Presentation Text Data                 12F7 D3EE9B  
19      ECT End Composed-Text Block                0008 D3A99B  
20      EPG End Page                              0010 D3A9AF  
21      ENG End Named Group                       0010 D3A9AD
```

AFP with triplet display output

For example, if you use the `arsafpd -d -t -i \acif\input\com.afp` command to display AFP information with the triplet display, the output could be:

```
1 BDT Begin Document                                0070 D3A8A8  
  BDT Coded Graphic Character Set Triplet (01)  
  BDT Code Page = 500 (1F4)
```

```

BDT Comment Triplet (65)
BDT Fully Qualified Name Triplet (02)
BDT 01 Replace First GID Name
BDT Name = 'TEST.#331027.ACIF.OUTPUT'
BDT Fully Qualified Name Triplet (02)
BDT 0A Begin Resource Group Reference
BDT Name = 'TEST.#331027.ACIF.RESOBJ'
BDT Fully Qualified Name Triplet (02)
BDT 98 Begin Document Index Reference
BDT Name = 'TEST.#331027.ACIF.INDEX'
2 BNG Begin Named Page Group 00000001 002A D3A8AD
  BNG Fully Qualified Name Triplet (02)
  BNG 01 Replace First GID Name
  BNG Name = 'AB 00000001'
  BNG Fully Qualified Name Triplet (02)
  BNG 8D Begin Medium Map Reference
  BNG Name = 'F2F515'
3 TLE Tag Logical Element 0026 D3A090
  TLE Fully Qualified Name Triplet (02)
  TLE 0B Attribute Name
  TLE Name = 'ADMIN CD'
  TLE Attribute Value Triplet (36)
  TLE Value = 'AB '
  TLE Attribute Qualifier Triplet (80)

```

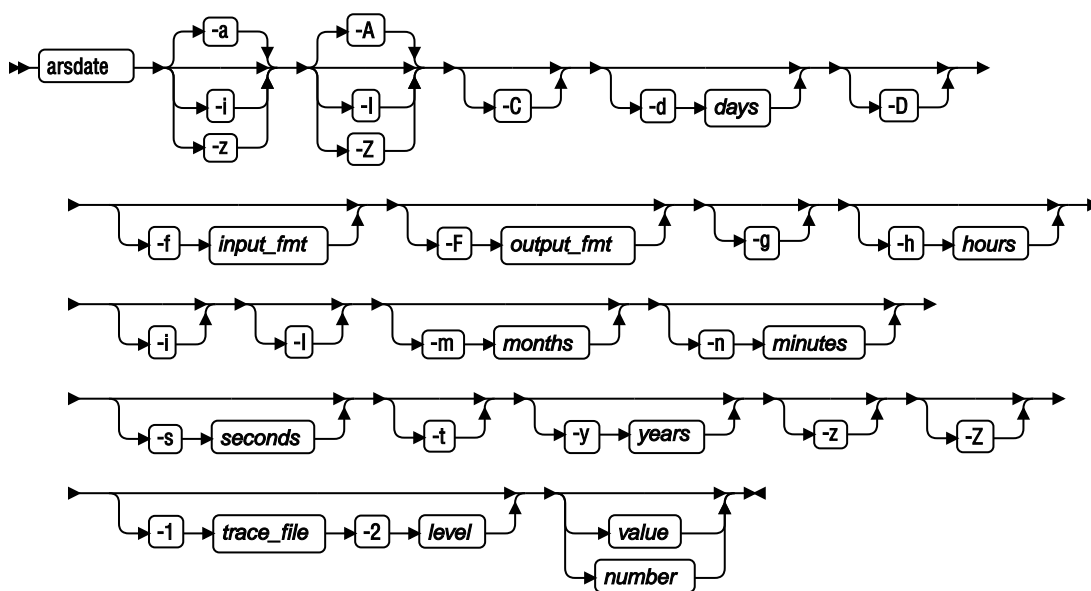

Chapter 19. ARSDATE

Purpose

For Date (old style), Date/Time (old style), Date/Time (TZ) (old style), and Time (old style) fields, do one of the following tasks:

- Content Manager OnDemand displays the internal database value for the specified date and time string.
- Content Manager OnDemand displays the date and time string for the specified internal database value.

Syntax



Description

The ARSDATE program displays the IBM Content Manager OnDemand internal database value for the specified date and time string or displays the date and time string for the specified Content Manager OnDemand internal database value.

Values whose data types are any of the following are represented in an internal form that is not apparent to the casual user of Content Manager OnDemand:

- Date (old style)
- Date/Time (old style)
- Date/Time (TZ) (old style)
- Time (old style)

Casual users enter date and time values the same way that they are displayed in a report. However, to search the database with an SQL string, a user must enter the internal form of the value. The ARSDATE program lists the internal value of a date or time string.

Parameters

-a

Use to display database values and date strings for date fields. For example, to display the database value for the date 9/1/99, enter:

```
arsdate -a 9/1/99
```

The ARSDATE program returns:

```
9/1/99 -> 10836
```

To display the date string for the database value 10836 in the default date format, enter:

```
arsdate -a 10836
```

The ARSDATE program returns:

```
10836 -> 9/1/99
```

-A

Output date type (defaults to input type)

-i

Use to display database values and date and time strings for Date (old style), Date/Time (old style), Date/Time (TZ) (old style), and Time (old style) fields. The time part of the Date/Time (old style) and Date/Time (TZ) (old style) fields is not adjusted for the local time zone. You typically use the -i parameter to find out one of two things:

- Given a date and time printed on a report, what value did Content Manager OnDemand store in Date (old style), Date/Time (old style), Date/Time (TZ) (old style), or Time (old style) database field? You can use the result to search a date or time field with an SQL string.
- Given a value stored in a Date (old style), Date/Time (old style), Date/Time (TZ) (old style), or Time (old style) database field, what would be the date and time printed on a report?

-I

Output date/time type (without timezone)

-z

Use to display database values and date and time strings for Date/Time (TZ) (old style) fields. The time part of a Date/Time (TZ) (old style) field is adjusted for the local time zone. If you run the ARSDATE program with the -z parameter on systems in different time zones and you specify the same date and time value, the result will be different. For example, suppose that you need to determine the value stored in a Date/Time (TZ) (old style) field for "09/01/00 04:00:00". The command:

```
arsdate -z "09/01/00 04:00:00"
```

When run on a server in the Eastern time zone will return:

```
09/01/00 04:00:00 -> 936187200
```

If you run the same command on a server in the Mountain time zone, then result will be:

```
09/01/00 04:00:00 -> 936180000
```

A typical use of the -z parameter is to determine a database value with which to search the system log. You can use the result to search the Date/Time (TZ) (old style) field of the system log with an SQL string. For example, suppose a user in New York logs on to a server in Denver. To retrieve the log on messages with an SQL string, you must specify the date and time part of the query using the local time of the client that is running the query. If the user logged on to the server at 4 a.m. Eastern time, then a query that is run in Denver must specify 2 a.m. to retrieve the message.

-Z

Output date/time type (with timezone)

-C

Convert to/from Content Manager OnDemand internal type and database type

-d days

Add the specified number of days to the specified internal value.

-D

Database type (default is Content Manager OnDemand internal type)

-f input_fmt

Determines the format of the date and time string that Content Manager OnDemand displays.

Table 7 on page 133 lists the standard date and time formats that are supported by Content Manager OnDemand. If the input data contains a date or time format that is not listed in the table, you can specify the format to Content Manager OnDemand. However, when specifying a format, you can only use values and separators from the standard formats that are listed in the table. Date and time formats may also be specified on the Load Information page in the application and the Field Information page in the folder.

The format consists of a set of values (for example, %m) and separators (such as the slash, dash, colon, or dot characters). A specification for a date format that uses any of these three chars (/ - .) can be used interchangeably; the program allows any of the three, regardless of which one you specified in the date format. A specification for a time format that uses either of these two chars (: .) can be used interchangeably; the program allows either of the two, regardless of which one you specified in the time format.

Table 7: Date and time format specifications

Date Format Specifier	Date Format	Example
%m/%d/%y	mm/dd/yy	01/31/17
%d/%m/%y	dd/mm/yy	31/01/17
%f/%e/%y	m/d/yy	1/31/17
%e/%f/%y	d/m/yy	31/1/17
%y.%m.%d	yy.mm.dd	17.01.31
%m.%d.%Y	mm.dd.yyyy	01.31.2017
%m-%d-%y	mm-dd-yy	01-31-17
%d-%m-%y	dd-mm-yy	31-01-17
%m%d%y	mmddy	013117
%m%d%Y	mmddyyyy	01312017
%y%m%d	yymmdd	170131
%Y%m%d	yyyymmdd	20170131

Table 7: Date and time format specifications (continued)

Date Format Specifier	Date Format	Example
%b %e, %Y	Mth d, yyyy	Jan 31, 2017
%B %e, %Y	Month d, yyyy	January 31, 2017
%m/%d/%y %H:%M	mm/dd/yy hh:mm	01/31/17 10:50
%H:%M	hh:mm	10:50
%T	hh:mm:ss	10:50:59
%H.%M	hh.mm	10.50
%T	hh.mm.ss	10.50.59

-F output_fmt

Output format string

-g

Display the current system date.

-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.

-t

Use to display database values and time strings for Time fields. The time is not adjusted for the local time zone. For example, to display the database value for the time 04:00:00, enter:

```
arsdate -t 04:00:00
```

The ARSDATE program returns:

```
04:00:00 -> 4800
```

To display the time string for the database value 4800, enter:

```
arsdate -t 4800
```

The ARSDATE program returns:

```
4800 -> 04:00:00
```

-y years

Add the specified number of years to the specified internal value.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

1. The following example shows how to determine the database value for the specified date and time string. The data type of the database field is **Date/Time (old style)**.

```
arsdate -i "09/01/99 04:00:00"  
  
09/01/99 04:00:00 -> 936158400
```

In the example, you could use the database value to search a **Date/Time (old style)** field in the database with an SQL string. For example:

```
arsdoc get -i "WHERE somedate=936158400" ...
```

2. The following example shows how to determine the date and time string for the specified database value. The data type of the database field is **Date/Time (old style)**. The result is shown using the default display format.

```
arsdate -i 936158400  
  
936158400 -> 09/01/99 04:00:00
```

3. The following example shows how to determine the database value for the specified date and time string. The data type of the database field is **Date/Time (TZ) (old style)**. The ARSDATE program adjusts the time part of the result for the local time zone.

```
arsdate -z "09/01/99 04:00:00"  
  
09/01/99 04:00:00 -> 936180000
```

If you were to run the same command on a server in the Eastern time zone, the result would be:

```
arsdate -z "09/01/99 04:00:00"  
  
09/01/99 04:00:00 -> 936187200
```

Notes

1. The ARSDATE program works with the following data types to display their internal values in a date or time format that you can understand:

- Date (old style)
- Date/Time (old style)
- Date/Time (TZ) (old style)
- Time (old style)

The following data types are stored in a date and time format that you can understand; therefore, you do not need to use the ARSDATE program to interpret the values:

- Date
- Date/Time
- Date/Time (TZ)

2. When displaying the date string for a given internal value, by default, the ARSDATE program displays the date string using the mm/dd/yy format. If you want the ARSDATE program to display the date

string using a different format, then you must specify the date format with the -f parameter. For example:

```
arsdate 10907
10907 -> 11/11/99
arsdate -f "%m/%d/%Y" 10907
10907 -> 11/11/1999
```

3. When displaying the internal value for a given date string, by default, the ARSDATE program expects you to specify the date string using the mm/dd/yy format. If you want to specify the date string using a different format, then you must specify the date format with the -f parameter. For example:

```
arsdate 11/12/99
11/12/99 -> 10908
arsdate 11/12/1999
11/12/1999 -> -1 (Error)
arsdate -f "%m/%d/%Y" 11/12/1999
11/12/1999 -> 10908
```

Files

/usr/lpp/ars/V10R1M0/bin/arsdate

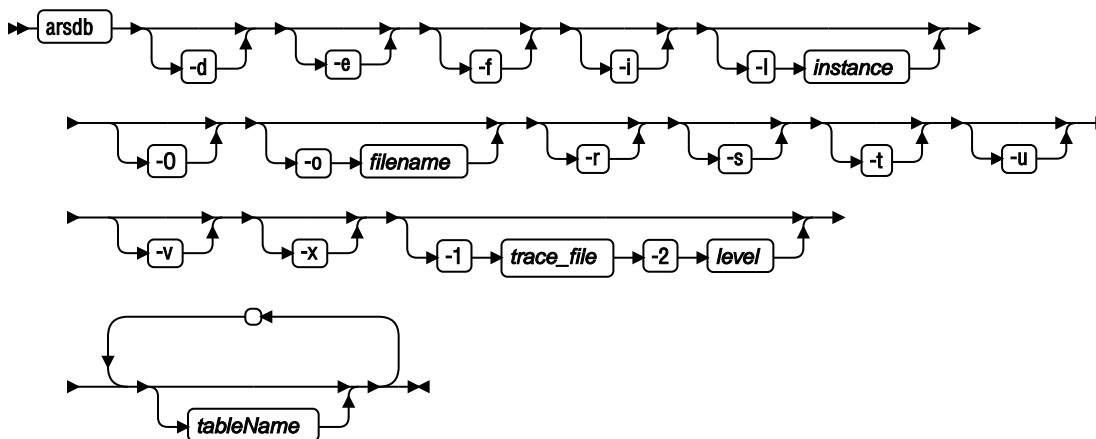
The z/OS executable program.

Chapter 20. ARSDB

Purpose

Provide an interface to the database manager for creating and initializing the Content Manager OnDemand database. The ARSDB program may also be used to drop indexes and tables and import and export tables. The Content Manager OnDemand database resides on the library server.

Syntax



Description

The ARSDB program provides an interface to the database manager to create and initialize the Content Manager OnDemand database, drop indexes and tables, and import and export tables.

The ARSDB program reads the ARS.CFG file to obtain the database configuration information. Before creating the database, you should verify the database configuration information. The *IBM Content Manager OnDemand for z/OS: Configuration Guide* provides details.

Parameters

- d**
Drop tables from the database. The default is all tables. To drop a specific table, use the `-d tableName` form of the command, where `tableName` is the name of the OnDemand system table.
- e**
Drops all the indexes for the specified Content Manager OnDemand system table (using `-e tableName`). The default, if no `tableName` is specified, is to drop all indexes for all Content Manager OnDemand system tables.
- f**
Ignore database manager failures.
- i**

Import tables. The default is all Content Manager OnDemand system tables. To import a specific table, use the `-i tableName` form of the command, where `tableName` is the name of the Content Manager OnDemand system table (and the name of the file that contains the table to be imported). Content Manager OnDemand reads the file from the directory in which the ARSDB program was invoked.

Important: The `-i` parameter should only be used to import tables from files that are created with the `-x` parameter when the system tables exist in multiple tablespaces.

-I instance

The name of the Content Manager OnDemand instance to process. The ARSDB program will attempt to locate the specified instance name in the ARS.INI file, to obtain the TCP/IP address, host name or host name alias of the system on which the instance is running. If the ARSDB program cannot locate the instance name in the ARS.INI file, then the specified value is treated as a host name.

You must specify the `-I` parameter and name the instance if:

- The name of the default instance is not ARCHIVE
- You are running more than one instance on the same system and you want to process an instance other than the default instance

Note: If you are running multiple instances of Content Manager OnDemand on the same system, then you should always specify the `-I` parameter to identify the name of the instance that you want to process. You should also verify that the ARS.INI file is configured with the correct information for all instances of Content Manager OnDemand. See the *IBM Content Manager OnDemand for z/OS: Configuration Guide*, for more information about working with multiple instances of OnDemand.

-m

Performs maintenance on system tables.

-o <filename>

Create the output file name.

-O

Parameter to identify the Content Manager OnDemand Distribution Facility. With the exception of the `-d` parameter, parameters specified with the ARSDB program will only be performed on the Content Manager OnDemand Distribution Facility tables if the `-O` parameter is specified.

-r

Create configuration indexes. The default is all indexes. To configure a specific index, use the `-r tableName` form of the command, where `tableName` is the name of the OnDemand system table.

-s

Run database statistics. Use this parameter to optimize indexes and tables and make access to information as efficient as possible. The default is all Content Manager 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 Content Manager OnDemand system table.

-t

Create configuration tables. The default is all Content Manager OnDemand system tables. To configure a specific table, use the `-t tableName` form of the command, where `tableName` is the name of the Content Manager OnDemand system table.

-u

Updates the Content Manager OnDemand system tables with new changes whenever you upgrade Content Manager OnDemand. Use this option only when you are explicitly directed by IBM.

-v

Enables verbose mode, to display all messages (informational and error). By default, the ARSDB program displays only error messages.

-x

Export tables. The default is all Content Manager OnDemand system tables. To export a specific table, use the `-x tableName` form of the command, where `tableName` is the name of the Content Manager OnDemand system table (and also the name of the file that Content Manager OnDemand creates). Content Manager OnDemand writes the file in the directory from which the ARSDB program was invoked.

tableName

Optionally specify the name of one or more Content Manager OnDemand system tables for the drop (`-d` and `-e`) and create configuration (`-r` and `-t`) parameters. If you do not specify the **tableName** parameter, then Content Manager OnDemand runs the command for all of the system tables.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

1. The following example shows how to use the ARSDB program to export the `arsprt` table from the database to a file. The output file is named `arsprt` and will be written to the directory in which the ARSDB program was invoked.

```
arsdb -x arsprt
```

2. The following example shows how to use the ARSDB program to import the `arsprt` table from a file into the database. The input file is named `arsprt` and resides in the directory in which the ARSDB program was invoked.

```
arsdb -i arsprt
```

Files

/usr/lpp/ars/V10R1M0/bin/arsdb

The z/OS executable program.

Chapter 21. ARSDOC

Purpose

The ARSDOC program is a multi-purpose document processing program. You can use the ARSDOC program to do the following tasks:

- Query the database and generate a list of items that match a query
- Retrieve documents from the system
- Add, delete, and update documents
- Send documents to a server printer
- Add and release holds on documents
- Add or remove documents from full text indexing
- Federate documents to IBM FileNet P8 through Content Federation Services for Content Manager OnDemand

Description and syntax

The ARSDOC program provides the following functions:

- ADD
- DELETE
- GET
- PRINT
- QUERY
- UPDATE
- HOLD_ADD
- HOLD_RELEASE
- CFSOD_FED
- FTI_ADD
- FTI_RELEASE

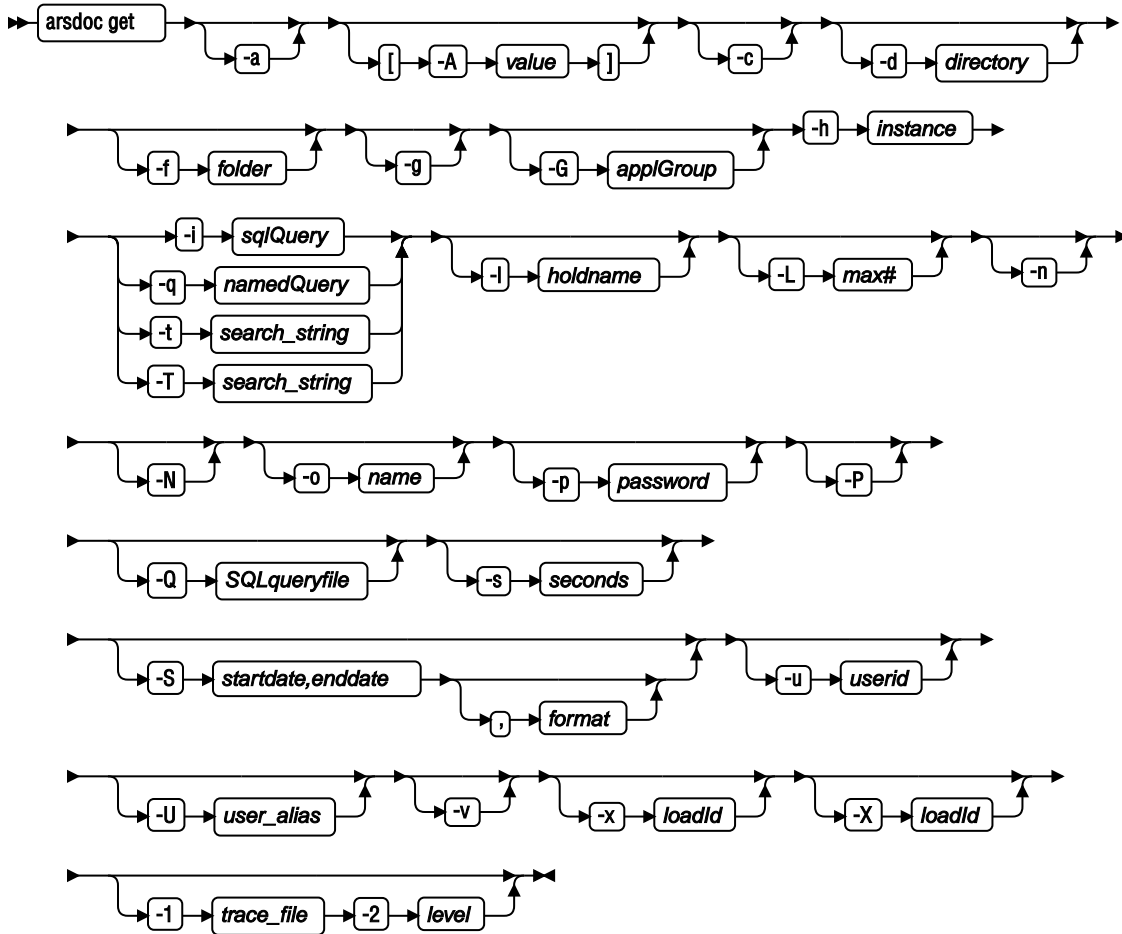
For each function, you can specify all of the required options on the command line or you can specify the name of a parameter file that contains the options. The syntax of each function is listed twice: first, when you specify the options on the command line; second, when you specify a parameter file.

GET function

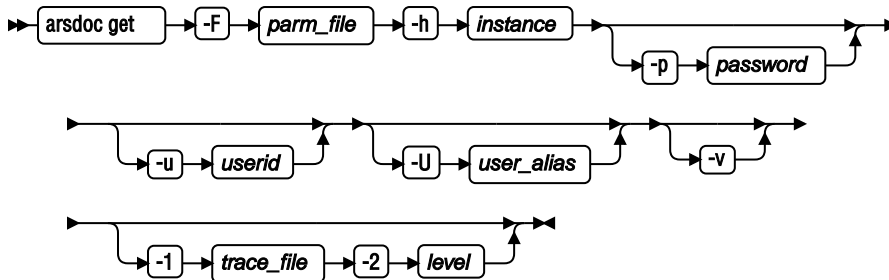
Use to retrieve documents and resources from the system. The GET function can also generate and save generic index data for the documents that match the query. You must identify the name of the IBM Content Manager OnDemand library server. You specify the application group or folder that you want to search. You can use the following parameters to query the database:

- SQL query (-i)
- hold name (-l)
- public named query (-n)
- text search string (-t or -T)
- load ID (-x or -X)

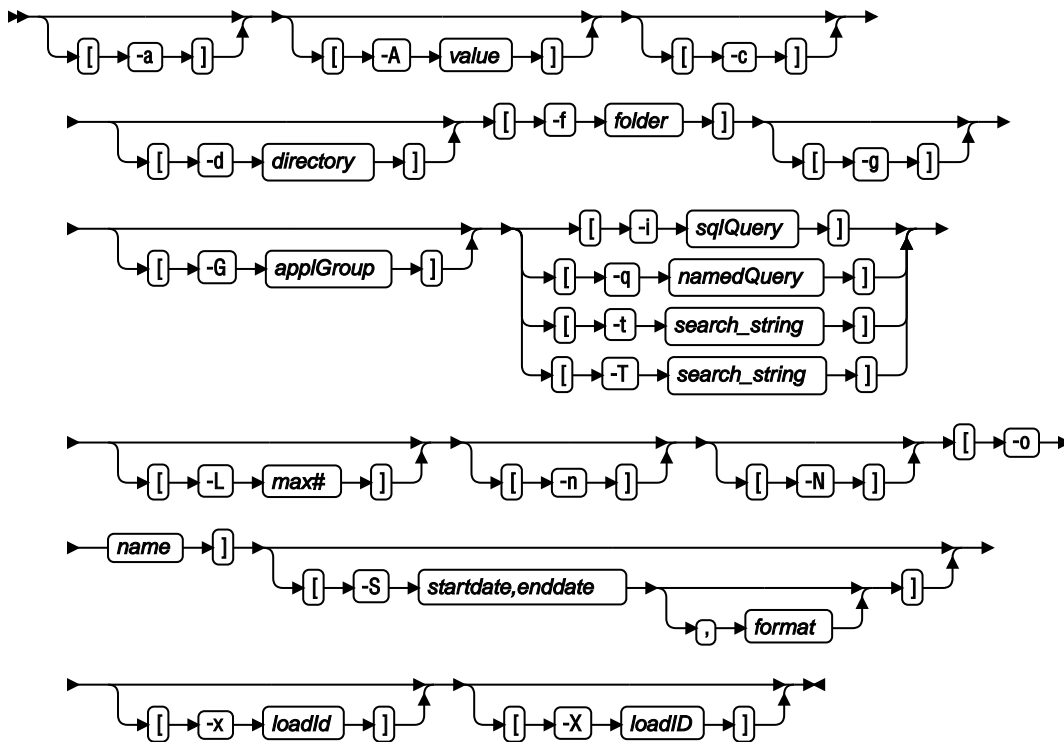
If you use an SQL query, it must contain clauses, database field names, index values, and operators. By default, the ARSDOC program sends a copy of the documents that match the query to stdout. You can also choose to write the output to a file. To retrieve documents in a sorted order, you must specify the -n parameter.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the GET function when you use the parameter file option:



The syntax of the parameter file for the GET function is:

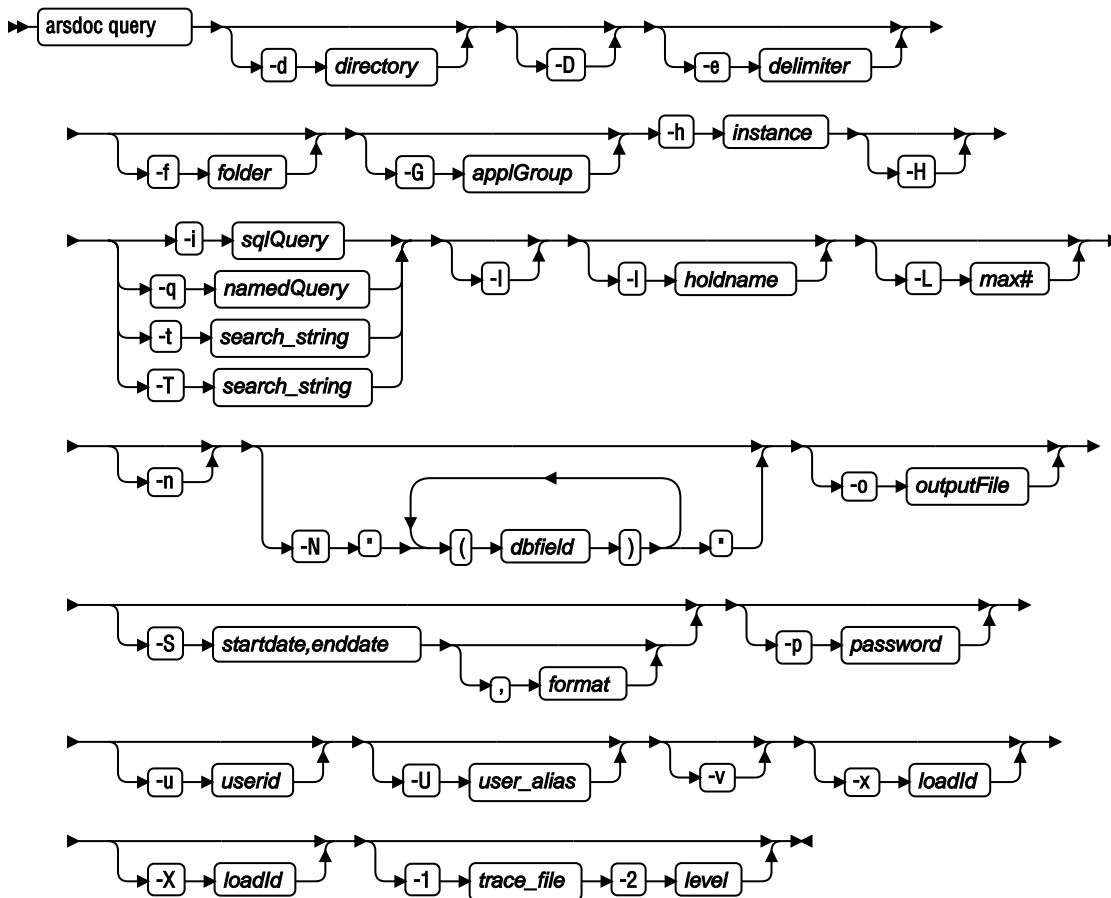


QUERY function

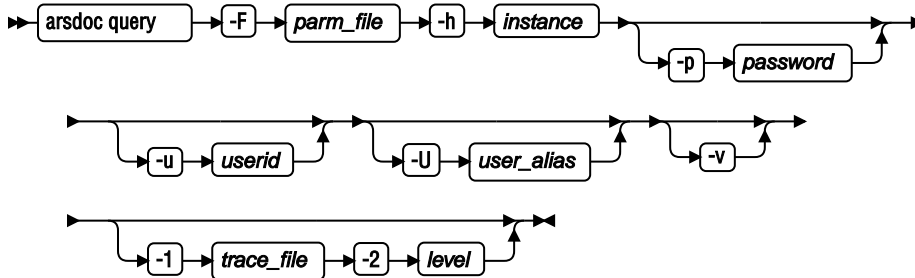
Use to generate a list of items that match a query. You must specify the name of the IBM Content Manager OnDemand library server. You specify the application group or folder that you want to search. You can use the following parameters to query the database:

- SQL query (-i)
- hold name (-l)
- public named query (-n)
- text search string (-t or -T)
- load ID (-x or -X)

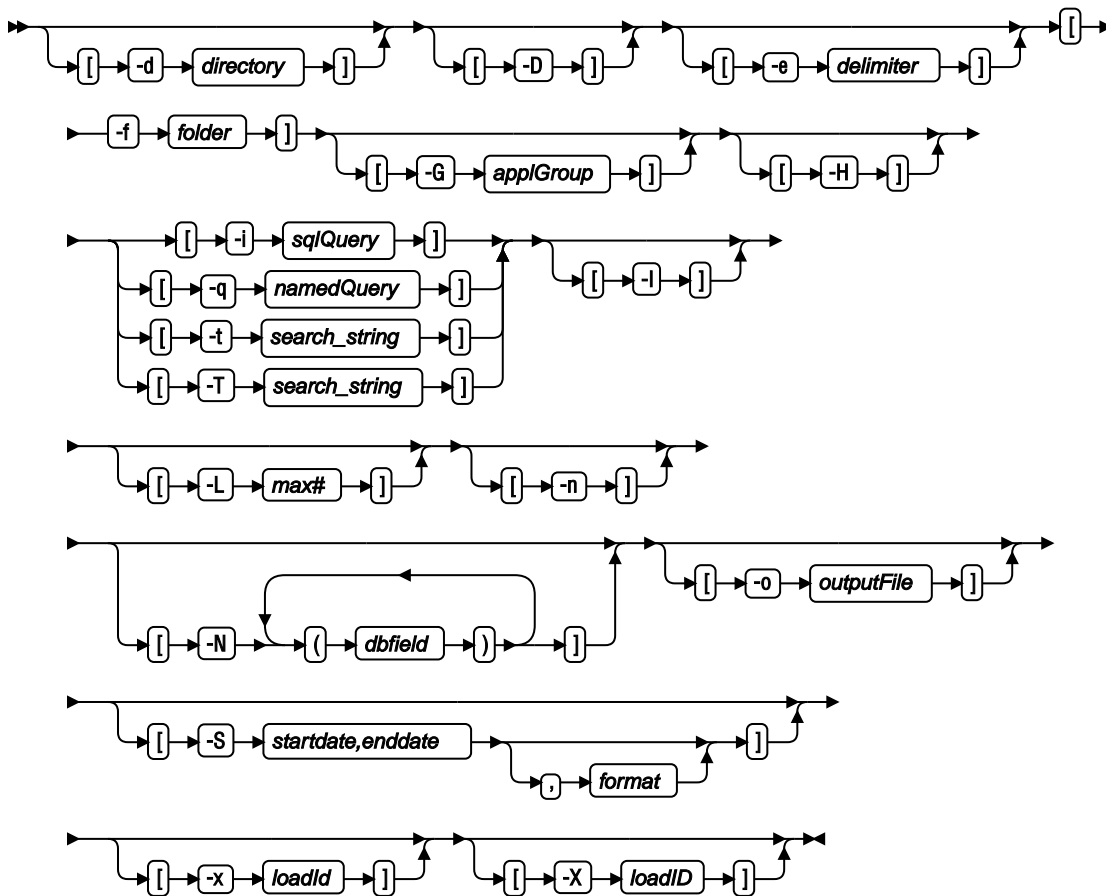
If you use an SQL query, it must contain clauses, database field names, index values, and operators. The data that is retrieved from the server is sorted by using the **Sort** field from the **Field Information** page for the folder.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following shows the syntax of the QUERY function when you use the parameter file option:



The syntax of the parameter file for the QUERY function is:



ADD function

Use to store data into the system by specifying the folder, application group, application, and database fields and values. If you specify the -O parameter, then you do not have to specify all of the database fields; however, you must always specify fields related to date or time fields. When you specify the -O parameter, the ARSDOC program stores a default value in any database field that you omit. The default value for string fields is an empty (null) 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 the following ways:

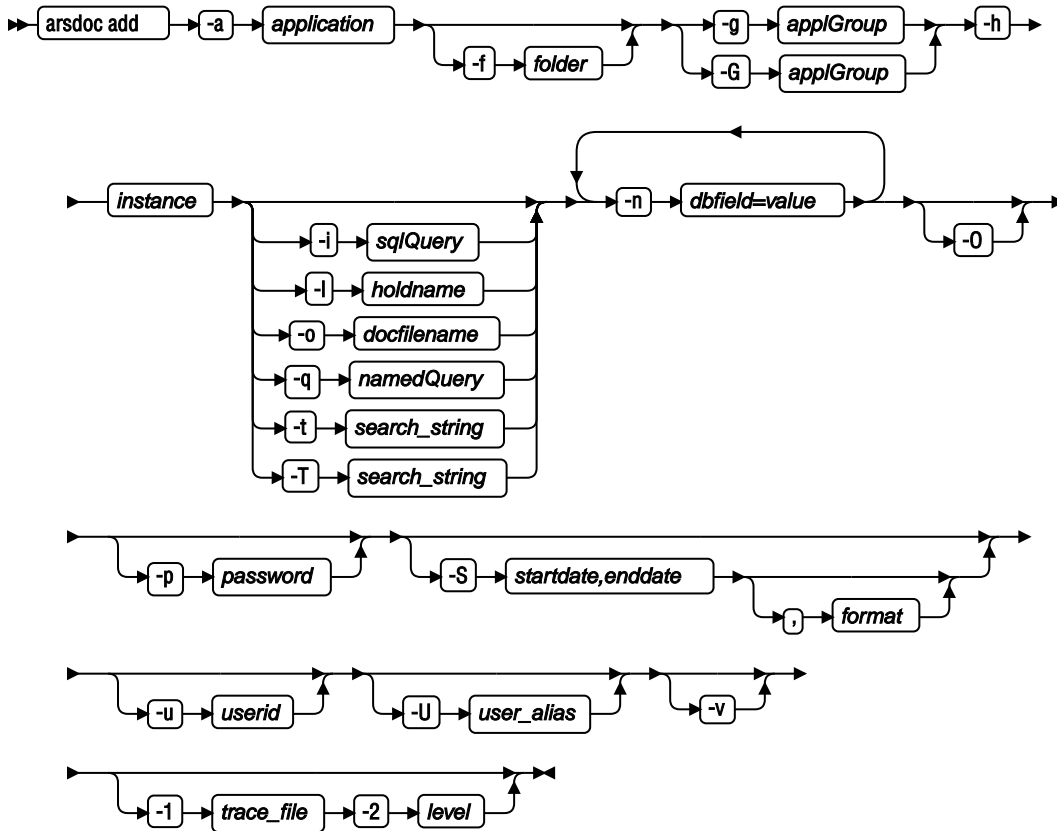
- A document file
- An SQL query that contains clauses, database field names, index values, and operators (-i)
- A public named query (-n)
- A hold name (-l)
- A text search string (-t or -T)

When you specify an SQL query, public named query, hold name, or text search string, you are creating a database row that points to an existing document that has been identified by the query.

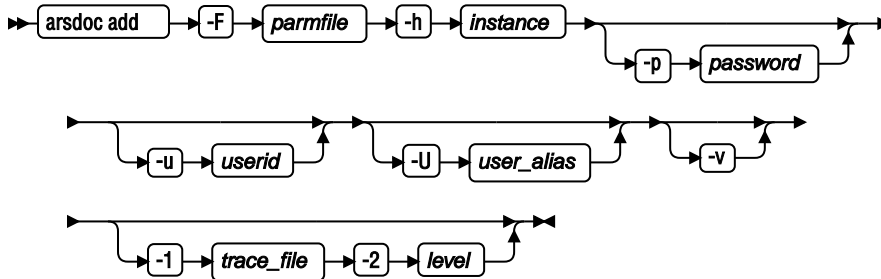
When you use the ADD function to add a row for an existing document, the row must be added to the application group and the application that contain the document. The application must be specified with the -a parameter. The application group must be specified with the -g parameter and must be one of the application groups referenced by the folder named with the -f parameter.

Important: The Database Organization of the application group must be set to Multiple loads per database table to add a document. If a new document is added, the Expiration Type must be set to Segment or Document. If you are adding index data for an existing document, Load can also be specified for Expiration Type if the existing document is identified by an SQL string, public named query, text string, or hold name.

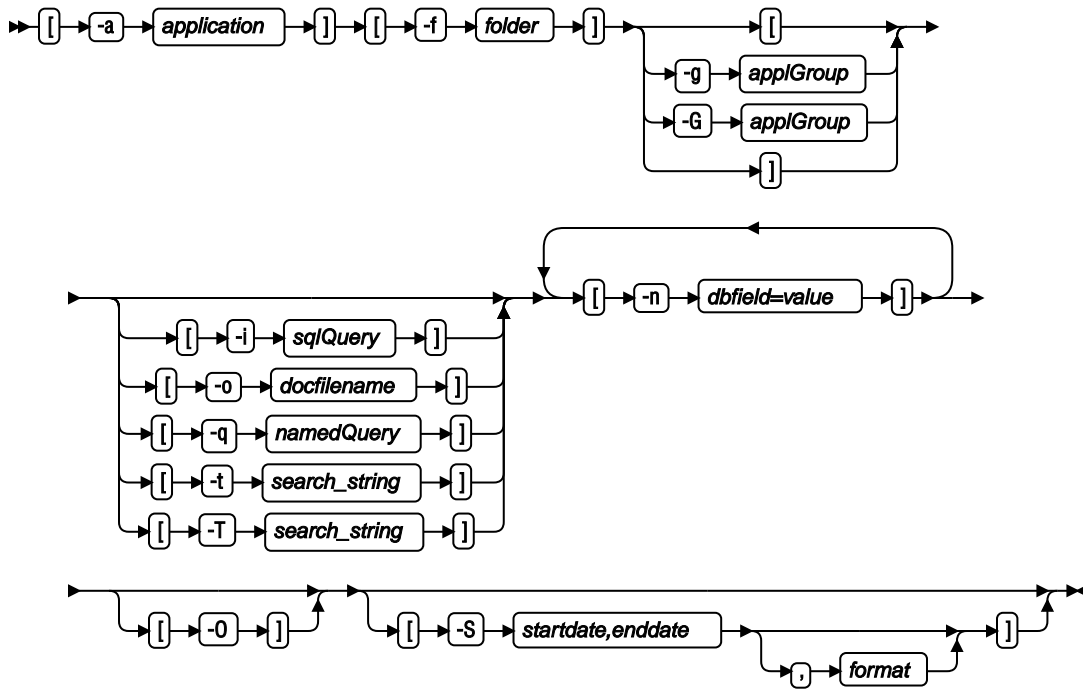
Storage management for the application group needs to be configured to use cache. Cache is required to allow the Content Manager OnDemand server to build a larger storage object and append to the data. Then this storage object can be migrated to TSM. Storing data directly into TSM is not recommended and can cause TSM performance issues.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the ADD function when you use the parameter file option:



The syntax of the parameter file for the ADD function is:



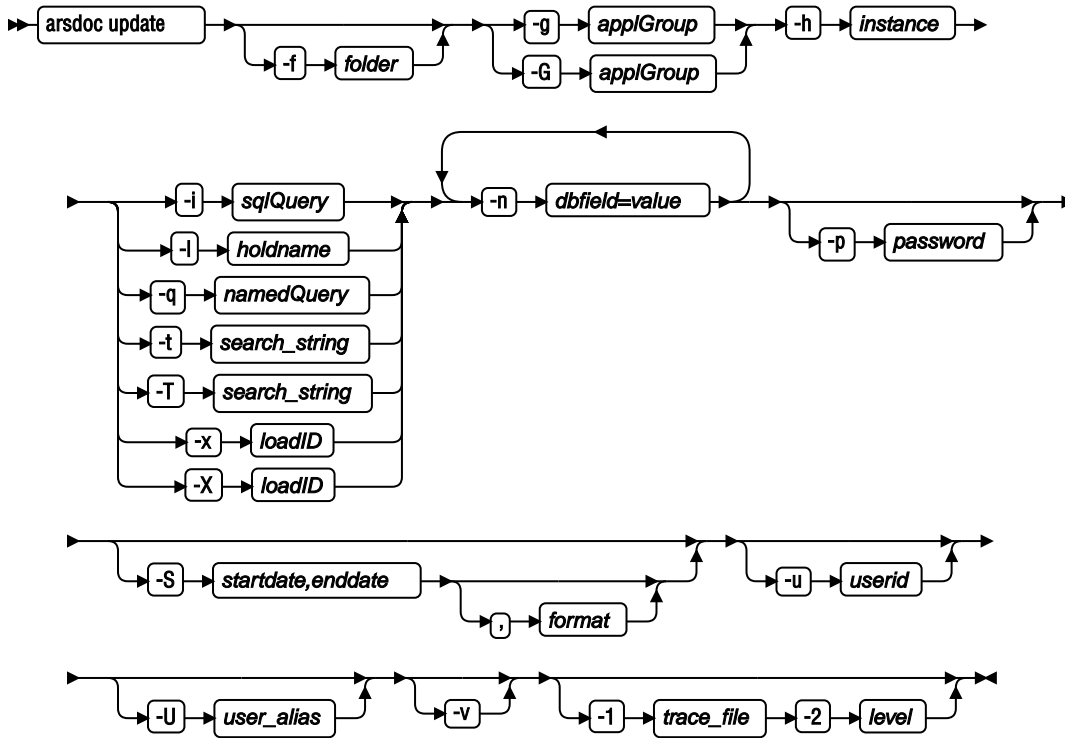
Important: To use the ADD function, every database field in the application group must be mapped to a folder field. The database field values are validated based on the information that is provided in the folder field.

UPDATE function

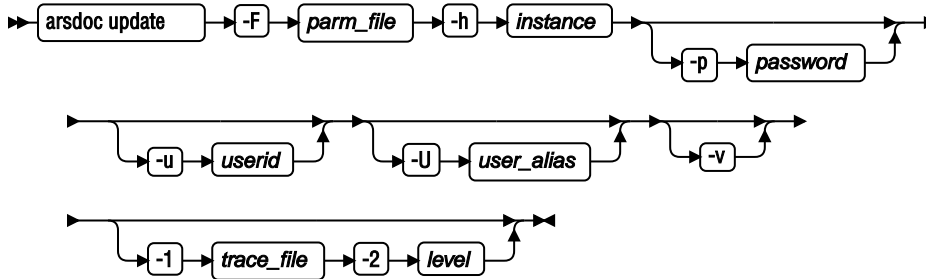
Use to update documents. You must name the application group to update and specify one or more application group fields and their values. To identify the documents that you want to update, you can use the following parameters to query the database:

- SQL query (-i)
- hold name (-l)
- public named query (-n)
- text search string (-t or -T)
- load ID (-x or -X)

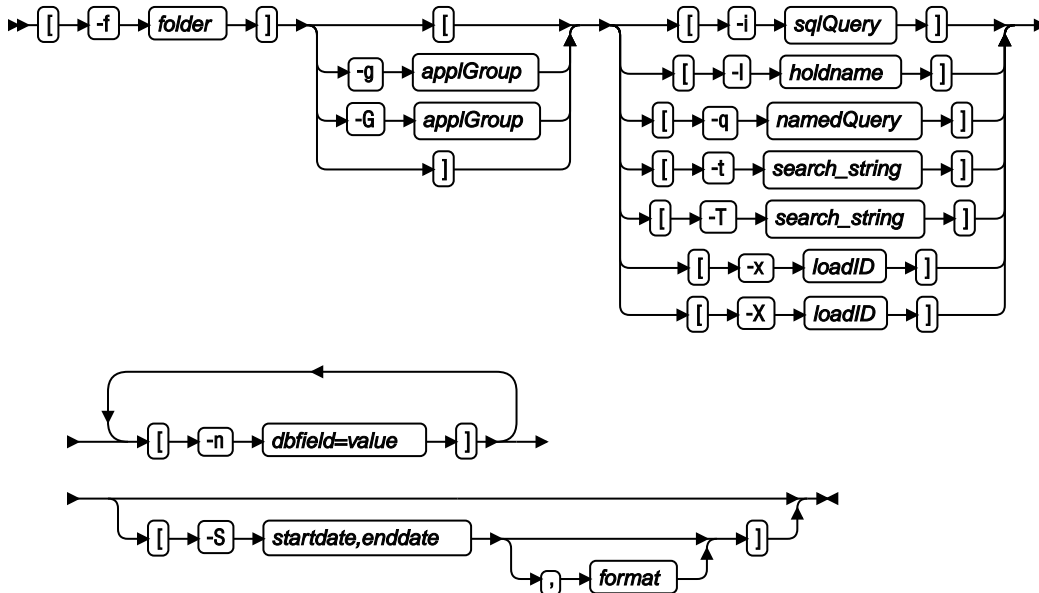
If you use an SQL query, it must contain clauses, database field names, index values, and operators. The UPDATE function updates all of the rows that match the query. It does not change the actual document data.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following shows the syntax of the UPDATE function when you use the parameter file option:



The syntax of the parameter file for the UPDATE function is:



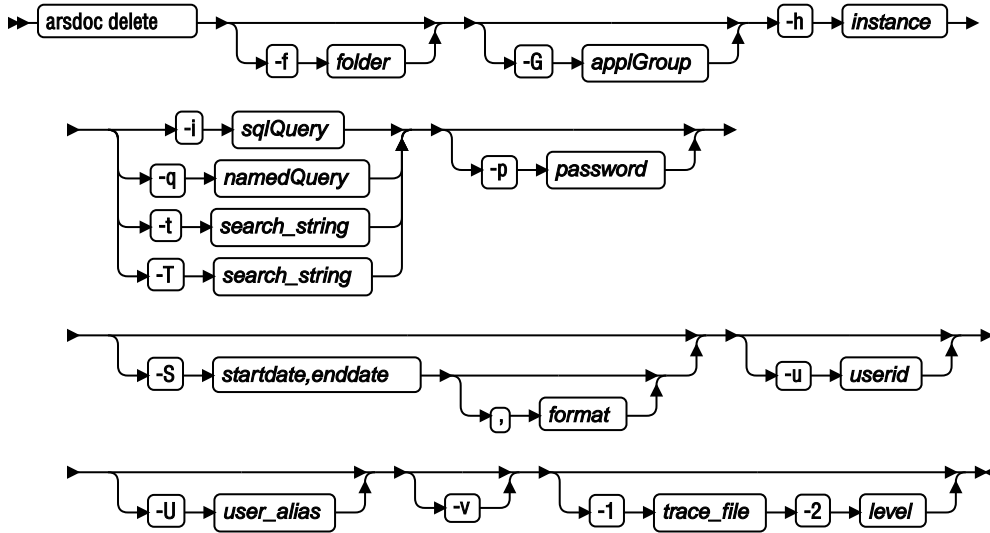
DELETE function

Use to delete documents from the system.

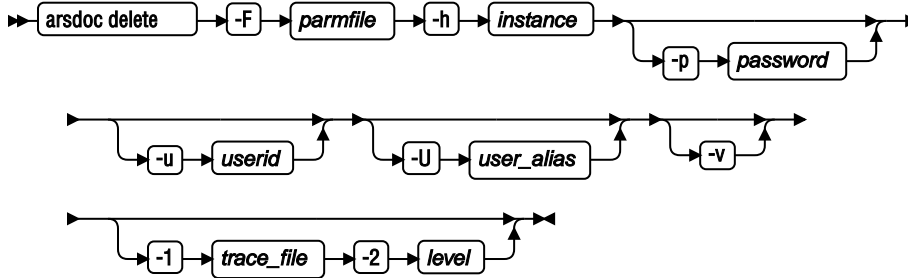
To identify the documents that you want to delete, you can use the following parameters to query the database:

- SQL query (-i)
- text search string (-t or -T)

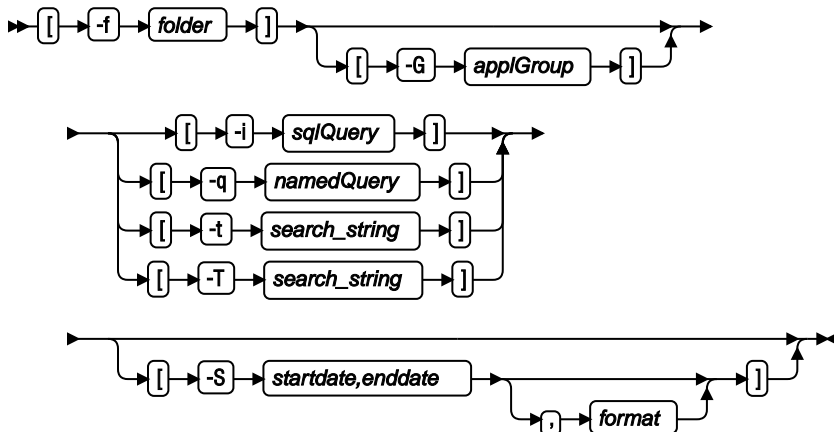
If you use an SQL query, it must contain clauses, database field names, index values, and operators. The DELETE function deletes all database rows that match the query.



When you use the parameter file option, you must specify the `-F`, `-h`, `-p`, `-u`, `-U`, `-v`, `-1`, and `-2` parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the DELETE function when you use the parameter file option:



The syntax of the parameter file for the DELETE function is:

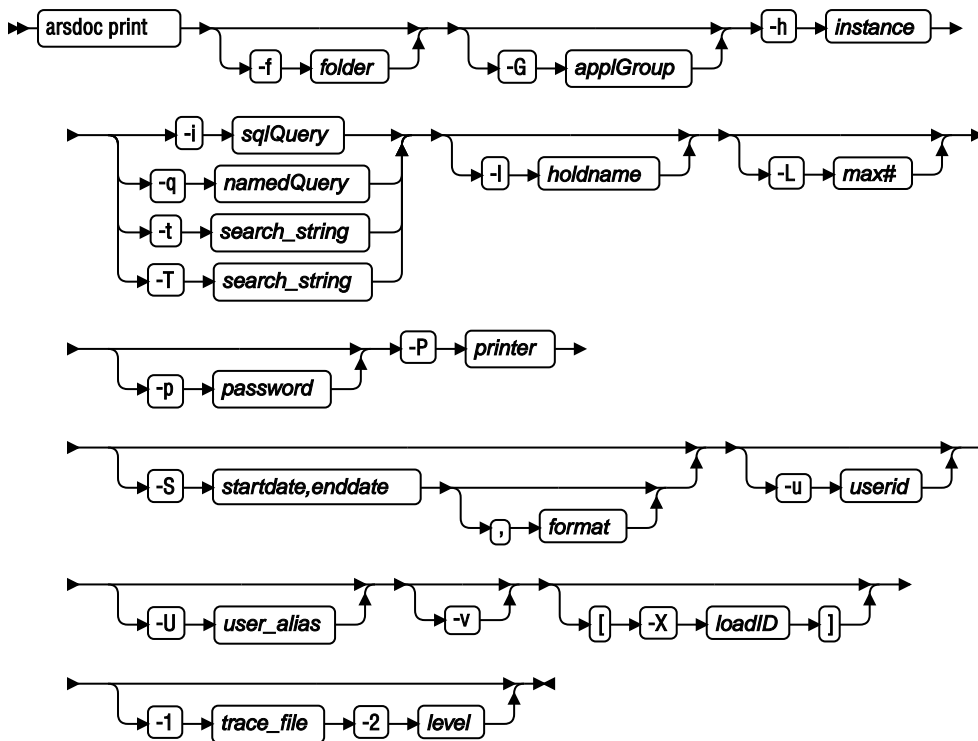


PRINT function

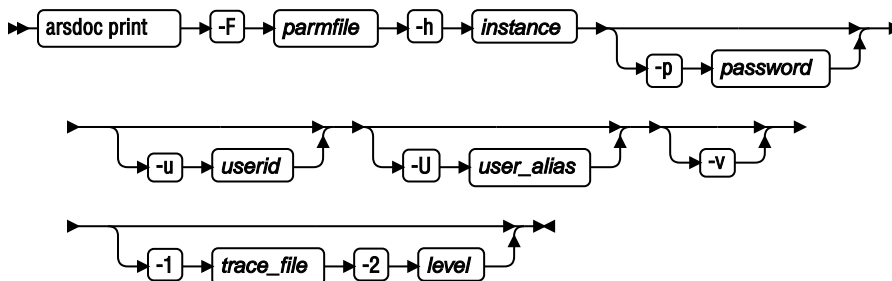
Use to send documents to an IBM Content Manager OnDemand server printer. You must name the Content Manager OnDemand library server. You specify the application group or folder that you want to search. The items that match the query are sent to the server printer named with the -P parameter. The server printer must be defined to Content Manager OnDemand by using either the OnDemand Administrator client or ARSXML. You can use the following parameters to query the database:

- SQL query (-i)
- hold name (-l)
- text search string (-t or -T)
- load ID (-X)

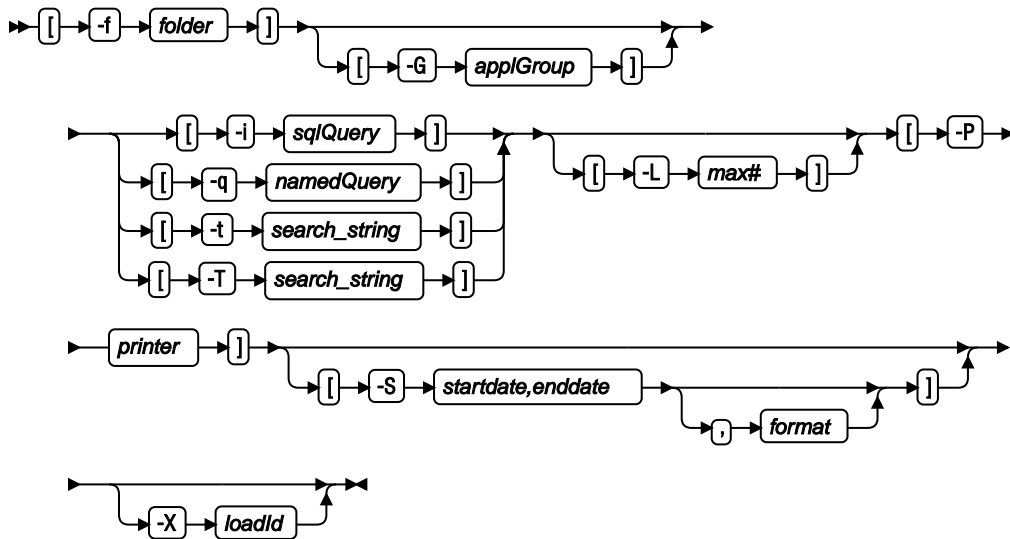
If you use an SQL query, it must contain clauses, database field names, index values, and operators. You can limit the number of documents sent to the printer by using the -L parameter and specifying the maximum number of documents that should be retrieved, regardless of the number of documents that match the query. The PRINT function does not support server fax.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the PRINT function when you use the parameter file option:



The syntax of the parameter file for the PRINT function is:



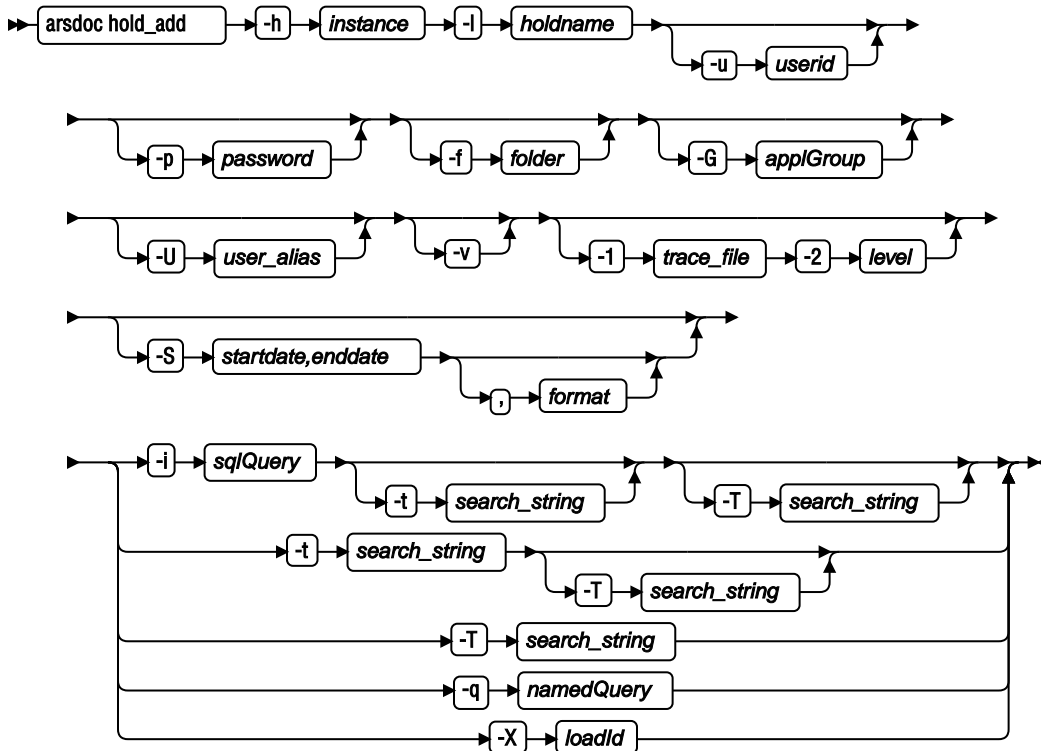
HOLD_ADD function

Use to add documents to a hold. To query the database to determine the documents to add to the hold, you can use the following parameters:

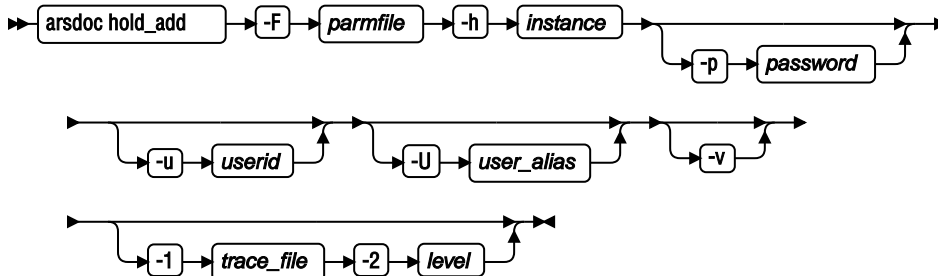
- SQL query (-i)
- text search string (-t or -T)
- load ID (-X)

Ensure that enhanced retention management is turned on for the application group and you have permission to add documents to a hold before you attempt this function. Otherwise, a permission error will occur.

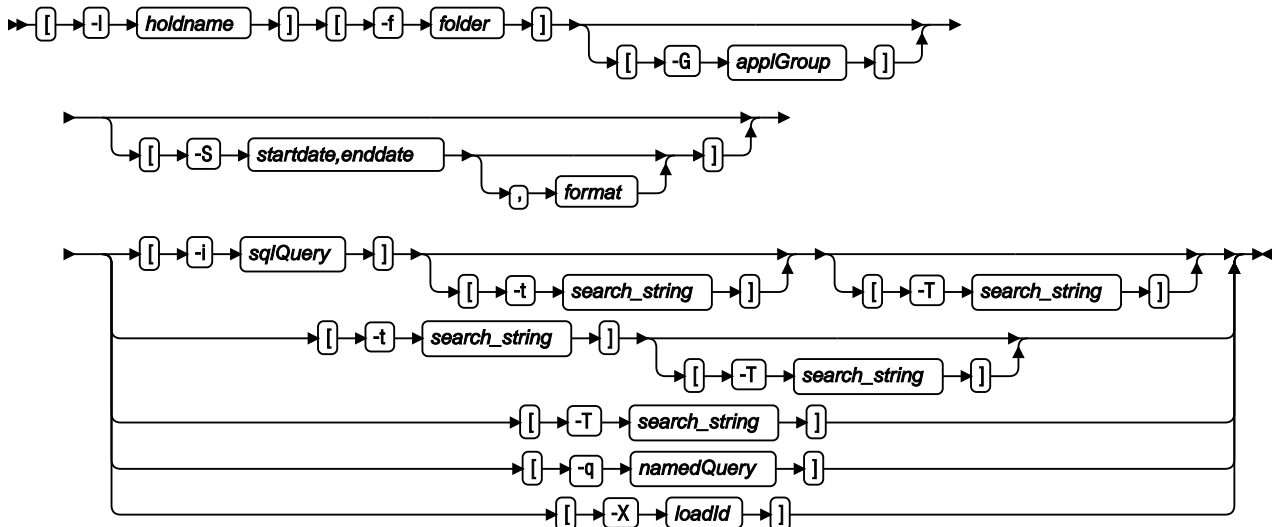
If a duplicate attempt is made to add the same documents to a hold, no error occurs, and ARSDOC displays a successful message. However, the system log might contain messages that indicate that the documents were already added.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the hold_add function when you use the parameter file option:



The syntax of the parameter file for the hold_add function is:



HOLD_RELEASE function

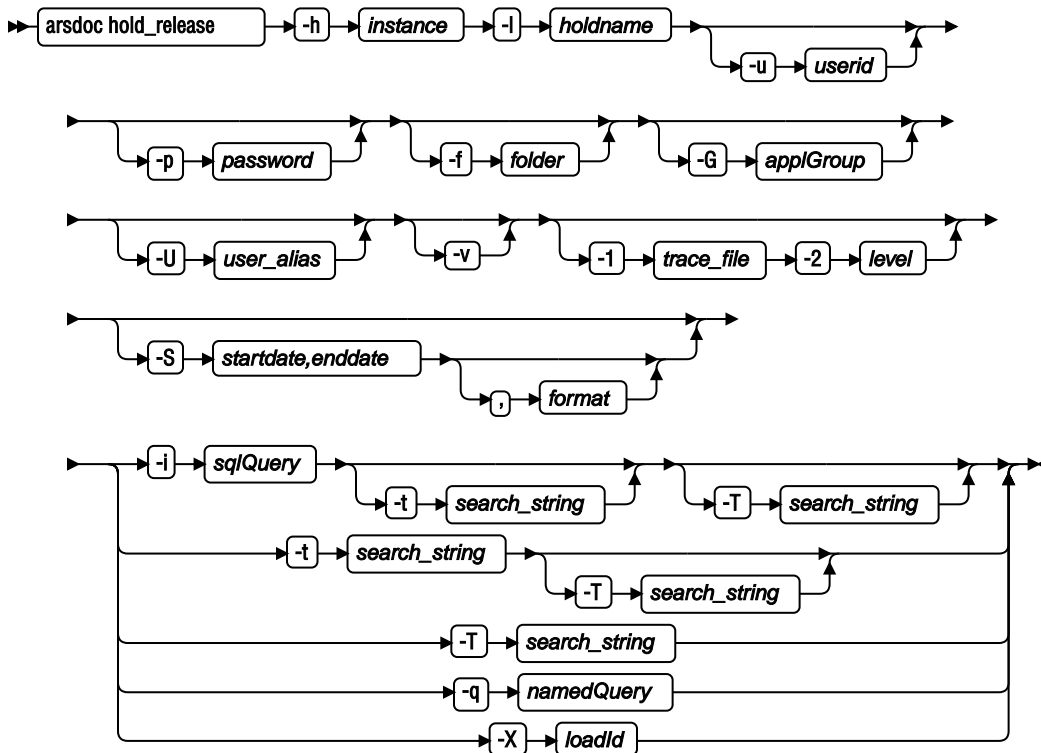
Use to remove documents from a hold. To query the database to determine the documents that should be removed from the hold, you can use the following parameters:

- SQL query (-i)
- text search string (-t or -T)
- load ID (-X)

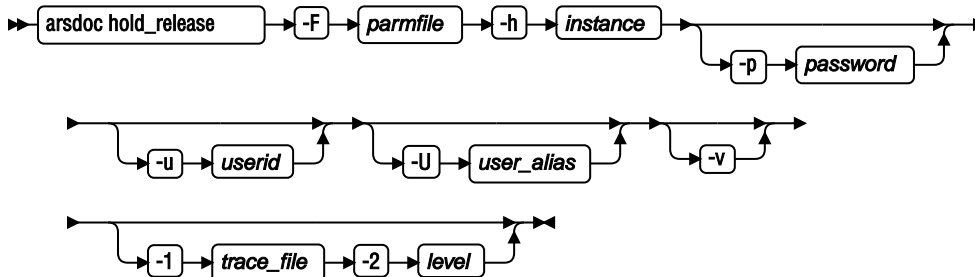
If you selected the option in the application group to use implied hold, documents are placed on hold as soon as they are loaded, and there is not a hold name that is associated with the documents that are placed on hold. To remove documents from an implied hold, specify IMPLIED_HOLD as the hold name.

Ensure that enhanced retention management is turned on for the application group and you have permission to release documents from a hold before you attempt this function. Otherwise, a permission error will occur.

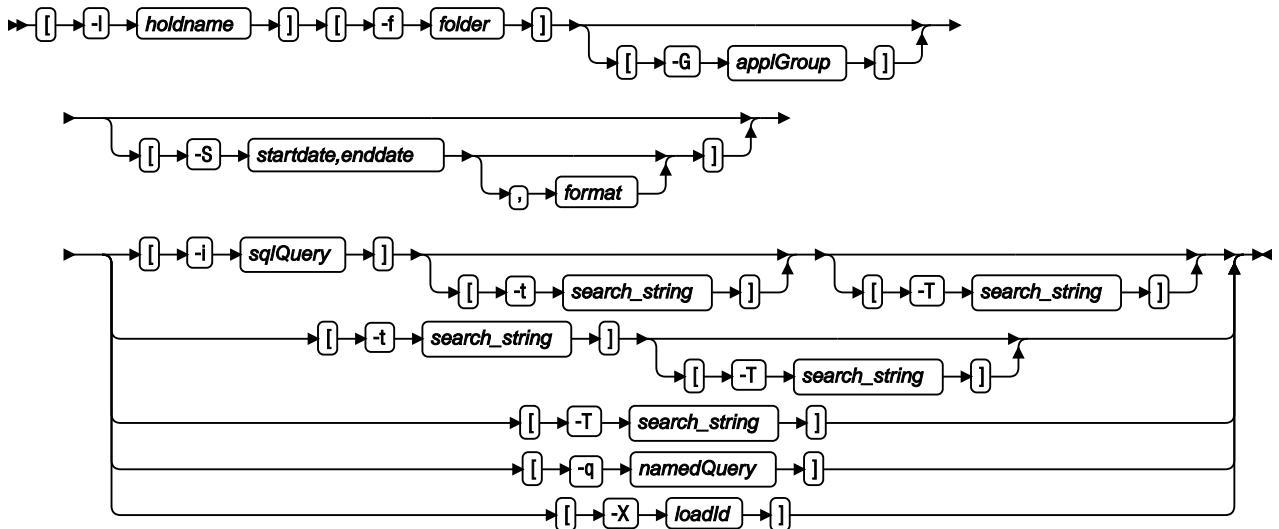
If a duplicate attempt is made to remove the same documents from a hold, no error occurs, and ARSDOC displays a successful message. However, the system log might contain messages that indicate that the documents were already released.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the hold_release function when you use the parameter file option:



The syntax of the parameter file for the hold_release function is:



CFSOD_FED function

Sends documents to CFS-CMOD and makes them available to IBM FileNet P8 features. To query the database to determine the documents to be sent to CFS-CMOD, you can use the following parameters:

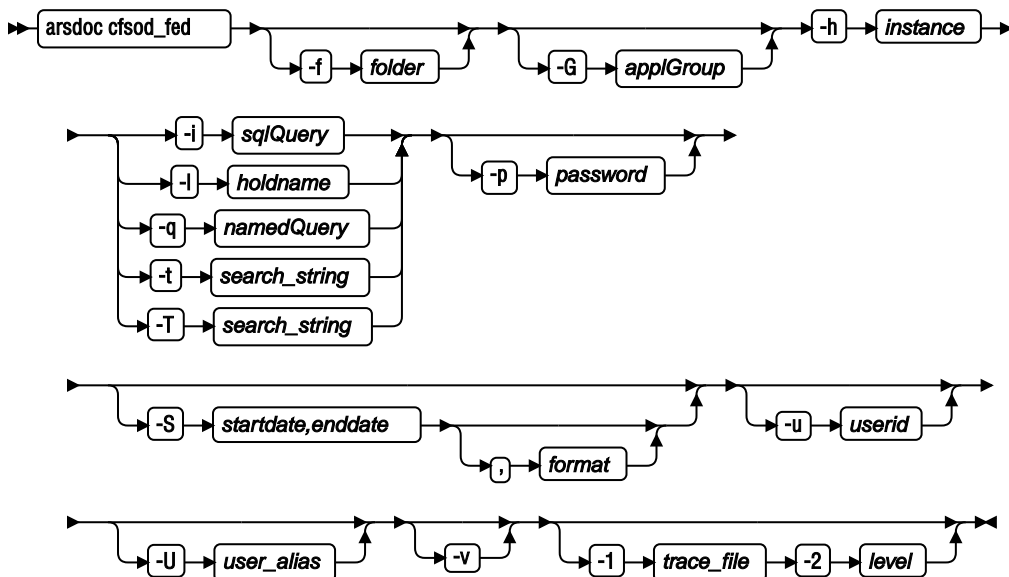
- SQL query (-i)
- hold name (-l)
- text search string (-t or -T)
- load ID (-X)

To enable the CFS-CMOD feature, add the following line to the ARS .CFG file:

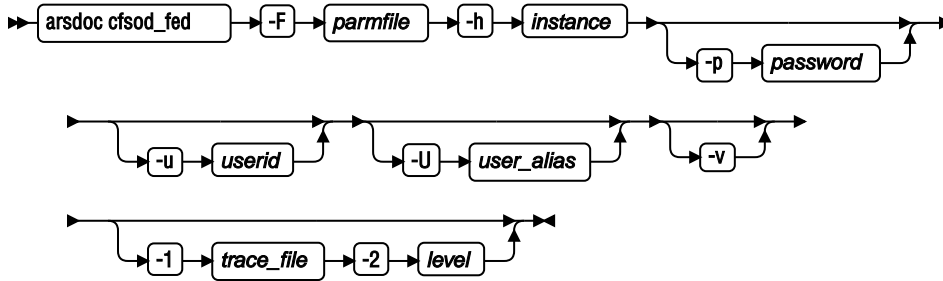
```
ARS_SUPPORT_CFSOD=1
```

Ensure that the CFS-CMOD function is turned on for the application group and you have permission to use CFS-CMOD before you attempt this function. Otherwise, a permission error will occur.

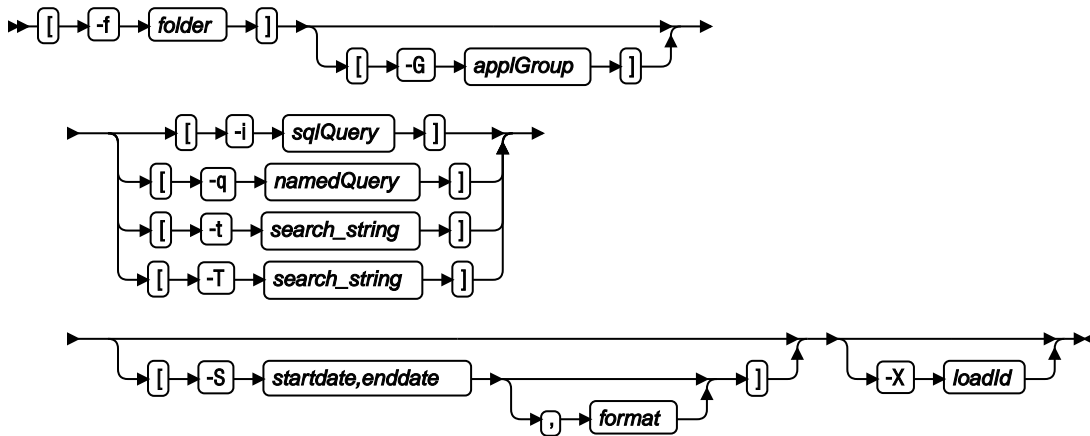
If a duplicate attempt is made to send the same documents to CFS-CMOD, no error occurs, and ARSDOC displays a successful message. However, the system log might contain messages that indicate that the documents were already sent to CFS-CMOD.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the cfsod_fed function when you use the parameter file option:



The syntax of the parameter file for the cfsod_fed function is:

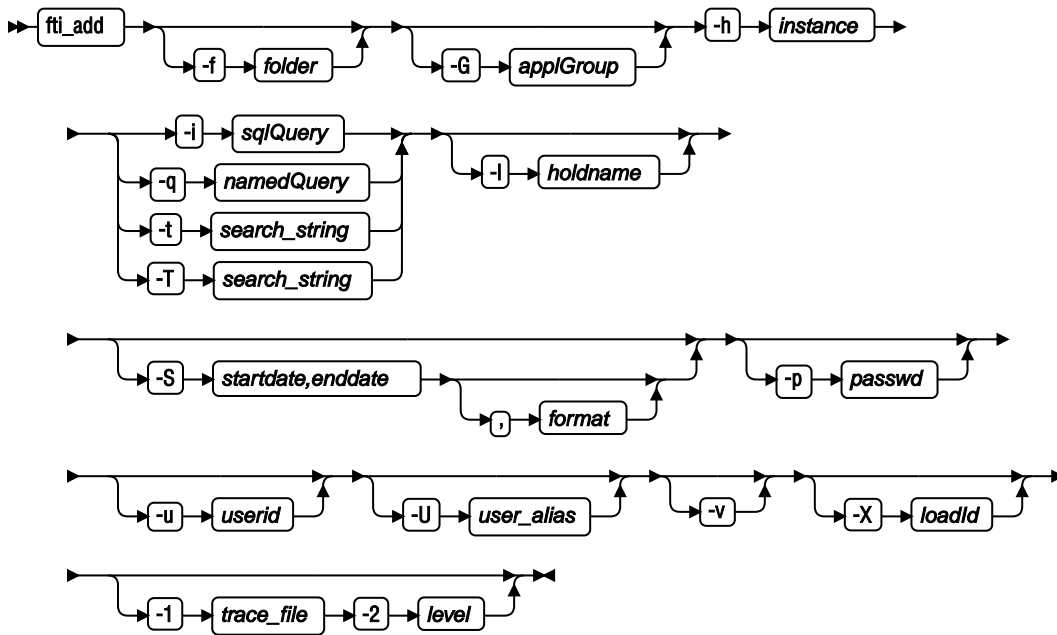


FTI_ADD function

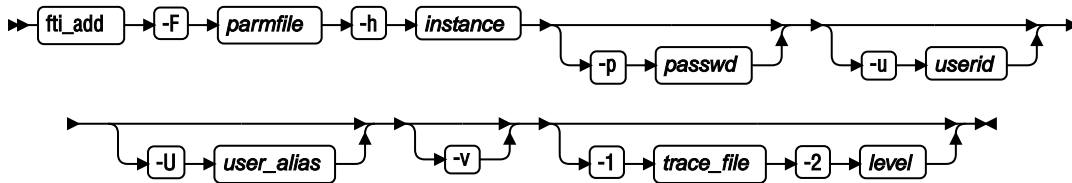
Use this function to add documents to the full text index. To query the database to determine the documents to be indexed, you can use the following parameters:

- SQL query (-i)
- hold name (-l)
- text search string (-t)
- load ID (-X)

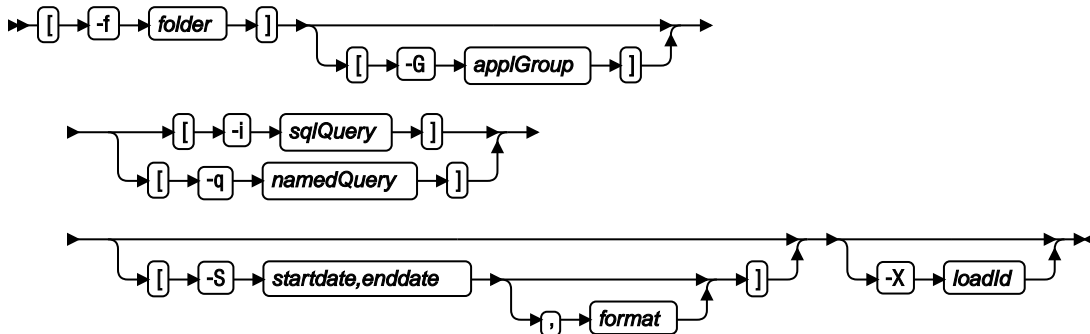
If a duplicate attempt is made to add the same documents to be indexed, no error occurs, and ARSDOC displays a successful message. However, the system log might contain messages that indicate that the documents were already indexed.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the fti_add function when you use the parameter file option:



The syntax of the parameter file for the fti_add function is:



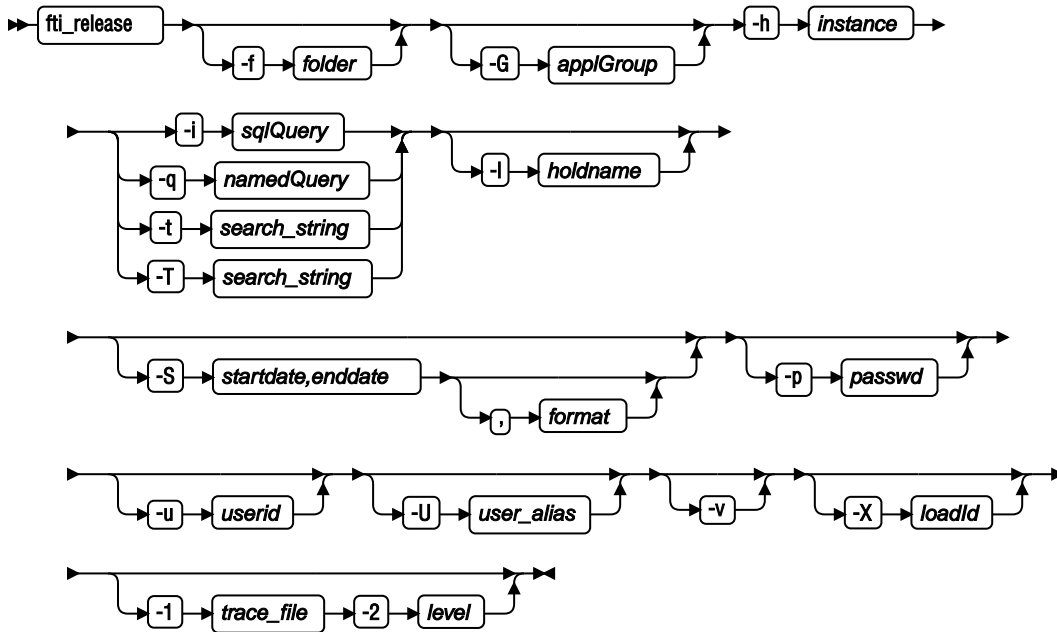
FTI_RELEASE function

Use this function to remove documents from the full text index. To query the database to determine the documents to be removed from the full text index, you can use the following parameters:

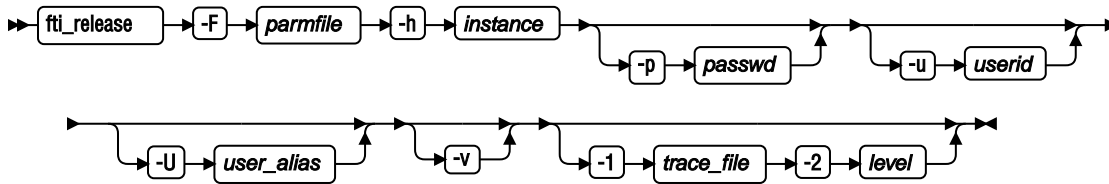
- SQL query (-i)
- hold name (-l)
- text search string (-t or -T)
- load ID (-X)

Ensure that Full Text Index is turned on for the application group and that you have permission to remove index information from the FTS server for the documents before you attempt this function. Otherwise, a permission error occurs.

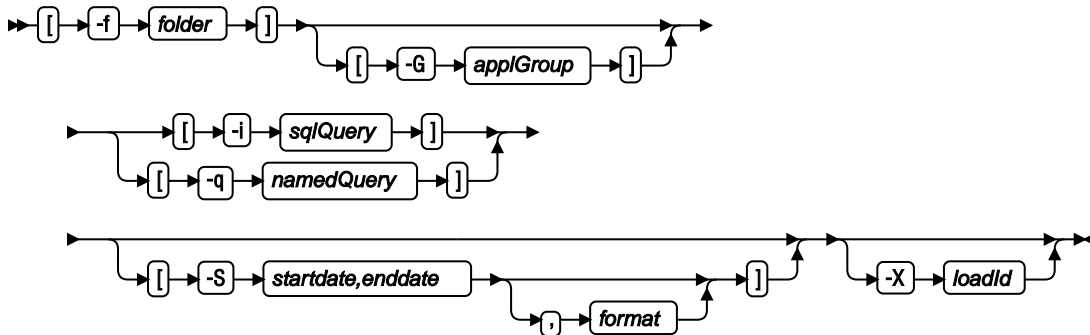
If a duplicate attempt is made to remove the index information from the FTS server, no error occurs, and ARSDOC displays a successful message. However, the system log might contain messages that indicate that the documents were already released.



When you use the parameter file option, you must specify the -F, -h, -p, -u, -U, -v, -1, and -2 parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the ft_release function when you use the parameter file option:



The syntax of the parameter file for the ft_release function is:



Parameters

-a

For the GET function, when retrieving AFP or XML documents, specify this parameter to include resources with the documents that are retrieved. If documents from the same application have different resource groups, then the ARSDOC program creates separate output files for each resource group.

For the ADD function, you must specify the name of the IBM Content Manager OnDemand application. The application must belong to the application group named with the -g (or -G) parameter.

Note that resource groups for XML documents are always written to a separate file, and are never added to the same file that contains document data even when the -c option is specified.

-A value

Use this parameter to retrieve annotations. The retrieved annotation files can be used by the ARSLOAD program if you need to reload the data and annotations in a different instance or on a different system. The following are the basic values of this parameter:

- 0** Include public text annotations
- 1** Include private annotations
- 2** Include annotations that cannot be copied to another server
- 4** Include graphic annotations

You can also add up two or more of the basic parameter values to create new values. For example:

- 3** Retrieve all public and private text annotations
- 5** Retrieve public and private annotations (text and graphic) that can be copied to another server
- 6** Retrieve all public annotations (text and graphic)
- 7** Retrieve all text and graphic annotations

Table 8: Possible values for the ARSDOC GET function -A parameter

Flag value	Public	Text	Can be copied	Private	Cannot be copied	Graphic
-A 0	X	X	X			
-A 1	X	X	X	X		
-A 2	X	X	X		X	
-A 3	X	X	X	X	X	
-A 4	X	X	X			X
-A 5	X	X	X	X		X
-A 6	X	X	X		X	X
-A 7	X	X	X	X	X	X

When the -A parameter is used and annotations exist, both .ann and .ang files are created. Both files will always be created whenever annotations are requested and at least one text or graphic annotation exists.

See “Examples” on page 170 for examples of using the -A parameter with the ARSDOC GET function.

-B orderbystring

For the QUERY and PRINT functions, used to specify which database field is used to sort the document list. The -B option cannot be specified if a load ID is specified.

-c

For the GET function, use to concatenate all of the documents that match the query into one output file. Name the output file with the -o parameter. However, even if you do specify the -c parameter, the ARSDOC program creates separate output files when any of the following conditions occur:

- If more than one application group is referenced by the folder. The ARSDOC program 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 program 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 program creates separate output files for each resource group.

For example, if a folder references two application groups, then the following specification:

```
-o student -c
```

Can result in file names such as: student.516 or 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.

If you have difficulty viewing documents that are retrieved in the same file, retrieve the documents as individual documents.

-d directory

The name of the directory where the ARSDOC program writes the output files. The directory must exist before the ARSDOC program attempts to save the output files.

-D

For the QUERY function, appends the document handle information to the end of each line. The document handle information consists of the following ten values, in the order listed:

1. Document name
2. Offset
3. Length
4. Compressed object offset
5. Compressed object length
6. Annotation type
7. Compression type
8. Resource ID
9. Primary node ID
10. 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.

-e delimiter

For the QUERY function, specifies a one character delimiter to use as a separator between values. By default, Content Manager OnDemand separates values in the output with a comma.

-f folder

The name of the Content Manager OnDemand folder. The folder name must be specified exactly as it appears in Content Manager OnDemand. The case of the folder name is significant. For example, to query the System Log folder, you must enter:

```
-f "System Log"
```

If you are using a parameter file, then you must specify the -f parameter in the parameter file. If you are not using a parameter file and you do not specify the -f parameter, then the ARSDOC program prompts you for the folder name when you run the program.

For the GET and QUERY functions, you can omit the -f parameter and specify the -G parameter to search a specific application group.

For the UPDATE function, if the folder that is specified with the -f parameter contains only one application group, then you can omit the -g or -G parameter (you do not have to specify the name of the application group).

When you specify the -X parameter, you cannot specify the -f parameter.

Note: The following information applies only when an application group name is not provided.

A folder can be used to search one or more application groups. Because the ARSDOC program generates a single SQL query to search all of the application groups, the properties of the database fields must be the same for each application group. The properties include the field name, type, and length. For example, suppose that 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 program 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

Then you could query the application groups using the ARSDOC program because the names of the database fields are the same for each application group.

-F parmFile

Specifies the name of the file that contains the actions to run and other parameters, values, and options. You typically specify this option when you want to perform more than one action.

When you specify a parameter file, delimit the options and values with left [and right] brackets. The left and right brackets identify each parameter in the file, are required in the parameter file, and the parameter values cannot contain left or right brackets. In the following example, the parameter file for the ARSDOC query function is parmfile.txt:

```
arsdoc query -u oduser -p odpasswd -h odserver -v -F parmfile.txt
```

The following lines are examples of parameter values in the parmfile.txt parameter file:

```
[-f "Credit Card Statements"] [-i "where account = '000-000-000'"]  
[-f "Credit Card Statements"] [-i "where account = '000-000-001'"]
```

If you want to use the left or right bracket as part of a parameter value, add a line to the beginning of the file that redefines the left and right delimiters with the keyword DELIMS. The keyword DELIMS is case-sensitive, must be on the first line of the file and it must start at the beginning of the line. For example, if you want to indicate that the left and right curly braces are the delimiters, add the line DELIMS={}. You can specify any two characters that are different, are not a space, and are not in any of the parameter values. The following example shows left and right curly braces defined as delimiters:

```
DELIMS={}  
{-f "Credit Card Statements"} {-i "where account = '000-000-000' and name = 'Republic Bank  
[North]'" }  
{-f "Credit Card Statements"} {-i "where account = '000-000-000' and name = 'Republic Bank  
[South]'" }
```

An action (one or more input lines) can contain a maximum of 32767 characters (bytes).

You can use the \ (backslash) character to continue the parameters of an action to two or more lines.

A parameter file can contain blank lines and comment lines. A comment line contains the # character in the first column.

-g

For the GET function, use to generate Generic indexer data for the items that match the query.

See the *IBM Content Manager OnDemand Indexing Reference* for details about the Generic indexer.

When you specify the -g parameter, you must specify the -c, -N, and -o parameters. However, you cannot specify database field names with the -o parameter.

The ARSDOC program uses the following convention to name the output files that are generated with the -g parameter:

```
-o.res_id.appl_group.appl.type
```

Where:

- -o is the value specified with the -o parameter
- res_id is the resource group identifier.
- 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 functions, specifies the name of the Content Manager OnDemand application group. The application group that you specify will be searched from the folder that is named with the -f parameter. For the UPDATE function, if the folder that is specified with the -f parameter contains only one application group, then you can omit the -g parameter (you do not have to specify the name of the application group).

-G applGroup

Use to specify the name of the application group.

For UPDATE: If the folder that is specified with the -f parameter contains only one application group, then you can omit the -G parameter (you do not have to specify the name of the application group).

For ADD: When the database query is run to retrieve the document that contains the data that is to be used in the add function, the search is limited to the specified application group, even if the folder named with the -f parameter can be used to search more than one application group. This ensures that only documents in the specified application group can be used for the add function. You can specify the name of the application group with the -g parameter or the -G parameter.

For DELETE: The -G parameter is an optional parameter. If specified, then the database query that is run to determine the document(s) to delete is limited to the specified application group. The addition of the -G parameter allows you to delete documents from a specific application group in folders that can search more than one application group. If you do not specify the -G parameter, then the query runs against all of the application groups that can be searched from the folder.

For GET: Specifies the application group to query and retrieve documents from. The -G parameter lets you retrieve documents from a specific application group from a folder that can search more than one application group. If you do not specify the -G parameter, then the query runs against all of the application groups that can be searched from the folder. You can omit the -f parameter and specify the -G parameter to search a specific application group. The -G parameter is required if you specify the -X parameter.

For PRINT: The -G parameter is an optional parameter. If specified, then the database query that is run to determine the document(s) to print is limited to the specified application group. The addition of the -G parameter allows you to print documents from a specific application group in folders that can search more than one application group. If you do not specify the -G parameter, then the query runs against all of the application groups that can be searched from the folder.

For QUERY: Specifies the application group to search. The -G parameter lets you search a specific application group from folders that can search more than one application group. If you do not specify the -G parameter, then the query runs against all of the application groups that can be searched from the folder. You can omit the -f parameter and specify the -G parameter to search a specific application group. The -G parameter is required if you specify the -X parameter.

For UPDATE: When the database query is run to determine the document(s) to update, the search is limited to the specified application group, even if the folder named with the -f parameter can search more than one application group. This guarantees that only documents in the specified application group can be updated. You can specify the name of the application group with the -g parameter or the -G parameter.

You can use the -G parameter with the -i parameter to query folders that can search more than one application group. For example, a folder contains three application groups; you want to query only one of the application groups. Use the -G parameter to specify the name of the application group that you want to query. Use the -i parameter to specify the application group's database field names. You can also use the -G and -i parameters when the application groups have different database field names. The following example shows how to search a folder and three application groups that have different database field names:

```
arsdoc get -f "Student Information" -G loans
-i "WHERE number LIKE '123456' AND loanDate = 10593"
arsdoc get -f "Student Information" -G grades
-i "WHERE number LIKE '123456' AND gradeDate = 10593"
```



```
arsdoc get -f "Student Information" -G transcripts
-i "WHERE number LIKE '123456' AND transDate = 10593"
```

You can use the -G parameter with the -q parameter to query folders that can search more than one application group. When you specify the -G parameter and you specify a public named query with the -q parameter, the ARSDOC program queries the application group named with the -G parameter instead of the application group specified in the named query. (If you do not specify the -G parameter, then the query runs against the application group specified in the named query. If the named query does not identify an application group, then the query runs against all of the application groups that can be searched from the folder.)

-h instance

The name of the Content Manager OnDemand instance to process.

This is a required parameter.

Important: If you are running multiple instances of Content Manager OnDemand on the same workstation, always specify the -h parameter to identify the name of the instance that you want to process. Verify that the system is configured with the correct information for all instances of Content Manager OnDemand.

See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for information about configuring instances.

-H

For the QUERY function, specify this parameter to generate a header record in the output. The default header record contains the application group field names. This parameter also generates a line that contains the names of the database fields. By default, the field names are delimited with the comma character. You can specify a delimiter of your choice with the -e parameter.

You can use the -H parameter to generate output that contains only the application group database field names. To do so, specify the -H parameter without specifying the -i or -q parameters. (You also must not specify the -L, -n or -N parameters.)

-i sqlQuery

A valid SQL query, that includes the names of one or more application group database fields, index values, and operators. Content Manager OnDemand does not validate the string that you specify. See the SQL reference for your database manager product for an overview of SQL concepts and details about how to construct a query.

To query the database to identify documents to be processed, the -i parameter can be used alone, or with one or more of the following parameters: -l, -t, -T, or -x.

For the DELETE or UPDATE functions, if more than one document meets the search criteria, then 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 (old style), Date/Time (old style), Date/Time (TZ) (old style), Date, Date/Time, or Date/Time (TZ), use the following SQL syntax rules:

- ODDAT_ must precede the name of the date database field name.
- There must be a space between the field name, the operator, and the value.
- Date/Time (TZ) or Date/Time (TZ) (old style) fields must be specified in Coordinated Universal Time (UTC), also known as GMT.
- For Date/Time (TZ) or Date/Time (TZ) (old style) fields, the value must be in the following format: 'YYYY-MM-DD HH:MM:SS.FFFFFFFF' where FFFFFFFF consists of six fractional digits.

-I

Important: When you use this parameter, you must specify the *f* or *p* variable, where *f* causes a search of the system log for the LoadId and *p* causes the LoadId to be created from document information. For example:

```
arsdoc query .. -I f
```

For the QUERY function, this parameter appends the Load ID to each line. 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.

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, then the string Load ID could not be found is appended to the end of the output record.

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 holdname

This parameter specifies the hold name.

To query the database to identify documents to be processed, the *-l* parameter can be used alone, or with one or more of the following parameters: *-i*, *-t*, *-T*, or *-x*. The *-l* parameter can also be used with the *-q* parameter if they are the only two search parameters specified.

If the hold name is used with the HOLD_ADD or HOLD_RELEASE function, the hold name identifies the name of the hold that will be added to or removed from the documents.

For all other functions, specifying the hold name limits the list of returned hits to only those hits that have been added to the specified hold. For example, an SQL query or a Named Query produces ten hits. If two of the ten hits have been added to a hold and the hold name is provided, the result contains only the two hits that are included in the hold.

-L max#

For the GET and PRINT functions, determines the maximum number of items retrieved from Content Manager OnDemand, regardless of the number of items that match the query.

For the QUERY function, determines the number of items included in the hit list, regardless of the number of items that match the query.

-n

For the GET function, use to retrieve items one at a time from the server. By default, the ARSDOC GET function uses a bulk retrieval method for high-speed retrieval of items from the server.

Important: Specify the *-n* and *-c* parameters if a sort order has been defined in the folder and it is a requirement that the documents be retrieved in the order specified by the sort order. If the *-c* parameter is not specified, the sort order will not be used when the documents are written to individual files. For more information about the sort order, see the online help on the Field Information page for the folder.

For the QUERY function, use to number the items in the output file. If you specify this option, the ARSDOC program sequentially numbers each line in the output file, beginning with 1 (one).

For the ADD and UPDATE functions, use to specify the application group database field names and their values using the form *-n dbfield=value*.

- Specify a null (blank) field value by using single quotes within double quotes. For example: *-n middle=" "*

- Specify a string field value that contains a null (blank) or other special character by enclosing the field value in single quotes within double quotes. For example: `-n name=" 'Sally Smith' "`

You can specify one or more field names and their values (by specifying the `-n` parameter one time for each database field name and its value). When adding a document, you must specify all of the application group fields unless you specify the `-O` parameter. When updating a document, you can specify one or more fields and their values. For a **date** field, you must specify the value using the **Display Format from the Field Information** page under folders.

-N "(dbfield1)(dbfield2)(dbfieldn)"

For the QUERY function, specify the order and names of the database fields to include in the output. For the GET function, when querying a folder that searches more than one application group or a folder that searches 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` parameter, you must specify the `-c` parameter. If you specify the `-g` parameter to generate generic index data, you must specify the `-N` parameter.

If the folder searches more than one application group or an application group contains more than one application and you do not specify the `-N` parameter, then the ARSDOC program 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` or `student.517`

Where 516 and 517 are application group identifiers. However, when you specify the `-N` parameter, the ARSDOC program 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` or `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 function, determines the application group fields that the ARSDOC program writes to the output file and the field names that appear in the header record. By default, the ARSDOC program 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"(dbfield)...(dbfield)"`.

-o name

For the GET function, use to write documents to one or more files and identify a 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` parameter 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:

- Content Manager OnDemand verifies that the field names that you specify are valid for the application groups that can be searched by the folder specified with the `-f` parameter.
- If the field name that you specify is a date field, the output format of the date is determined by the **Format on the Load Information** page under applications.
- 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 the ARSDOC program generates.
- You cannot use a field name to represent a directory name. For example, the following is not valid:

```
-o "(field_1)/(field_2)"
```

- You cannot specify the `-c` parameter 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` parameter, or specify the `-g` parameter, then the ARSDOC program 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 program creates the following files: `statements.1` and `statements.2`

You must specify the `-o` parameter when you specify the `-c` parameter.

For the QUERY function, determines the file name of the output file in which the ARSDOC program writes the list of items that match the query.

For the ADD 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 or forward slash characters that are part of a directory path. When adding a document, you can provide the input data by specifying the name of the input file that contains the data with the `-o` parameter, an SQL query with the `-i` parameter, or a public named query with the `-q` parameter. Only one document may be added at a time.

-O

For the ADD function, you must specify this parameter if you intend to omit one or more database fields. However, you can never omit the following fields:

- Date
- Date (old style)
- Date/Time
- Date/Time (old style)
- Date/Time (TZ)
- Date/Time (TZ) (old style)

When you specify the -O parameter, the ARSDOC program stores a default value in any other database field that you omit. The default value for string fields is an empty (null) string. The default value for numeric fields is 0 (zero). Numeric fields include integer and decimal fields.

The -O option is not required in the case where an index row is being added for an existing document. For any index values that are not provided, the values from the existing document are used.

-p password

Specify one of the following options for *password*:

- The name of the stash (encrypted password) file that contains the password for the user ID specified with the -u parameter.
- The password of the Content Manager OnDemand user that you named with the -u parameter. If there is no password assigned to the user that you specify, use quotation marks to show a null password. That is, specify -p "". If you do not specify the -p parameter while you specify the -u parameter, then the ARSDOC program retrieves the password for the user ID from the stash file specified in ARS.INI file for that instance. If there is no password assigned to the user that you specify, press the Enter key when prompted.

-P printer

For the PRINT function, identifies the Content Manager OnDemand server printer to which you want to send the documents that match the query.

-q namedQuery

The name of a public named query for the folder named with the -f parameter. 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.

To query the database to identify documents to be processed, the -q parameter can be used alone, or with one or more of the following parameters: -l or -x.

-Q SQLqueryfile

Use this parameter to specify a file name that contains one or more query strings. Specify only one of the parameters, -i, -q, or -Q.

The name of the file can include a directory path. The directory path that is specified by the -d flag does not apply to the -Q flag.

The SQL query string can span multiple lines in the file as long as the continuation character '\' is included as the last character on the line that is to be continued. Blank lines and comment lines (that is, lines starting with '#') are also supported.

The '#' and '\', as well the actual query string, must be in the codepage associated with the locale under which the ARSDOC get command is running.

If multiple SQL query strings are provided, one hit list is created. Duplicate hits are not removed from the hit list.

If multiple SQL query strings are used, at least one of the database queries must produce documents in the hit list before you can perform a document retrieval. If one of the database queries fails (different from success with no documents found), none of the documents is retrieved.

The maximum length of a specific query string is 32,767. However, the server must provide an SQL SELECT clause to the query string. In other words, the actual maximum length of the query string is less than 32,767.

-s seconds

For the GET function, determines the number of seconds that the ARSDOC program waits between query requests when you specify more than one query with the -F parameter. If you do not specify this option, then the ARSDOC program does not wait between query requests. That is, the default is 0 (zero) seconds.

-S startdate,enddate,format

Provides a date range that the ARSDOC program uses to limit a search to specific tables. When you specify this parameter, the ARSDOC program searches only tables that contain a segment within the specified date range.

You can optionally specify a date format. See [Chapter 19, “ARSDATE,” on page 131](#) for a list of the standard date formats. An example of a date range with a date format is:

```
-S "01011990,12311990,%m%d%Y"
```

If you do not specify a date format, then the date values must be specified by using the Display Format that is set on the Field Information page in the folder. An example of a date range without a date format is:

```
-S "01011990,12311990"
```

Important:

- For most queries, you should always specify the -S parameter and specify a date range. Doing so limits the range of a query and can significantly improve the performance of a query.
- For the ADD function, if you specify the -o parameter, you cannot specify the -S parameter.
- For the GET and QUERY functions, if you specify the -X parameter, you cannot specify the -S parameter.
- For all functions, if you specify the -q parameter, you cannot specify the -S parameter.

-t search_str

Search for the string *search_str* using the Content Manager OnDemand search.

To query the database to identify documents to be processed, the -t parameter can be used alone, or with one or more of the following parameters: -i, -l, -T, or -x.

-T search_str

Search for the string *search_str* using the full text index search.

To query the database to identify documents to be processed, the -T parameter can be used alone, or with one or more of the following parameters: -i, -l, -t, or -x.

-u userid

The Content Manager OnDemand user that is permitted to perform the specified function. The ARSDOC program verifies the following permissions:

- the user ID that you specify is a valid Content Manager OnDemand user for the library server that you name with the -h parameter
- the user ID is permitted to open the folder that you name with the -f parameter
- the user ID has permission in application groups to perform the specified function

If you omit the -u parameter, then the ARSDOC program prompts you to enter the user ID when you run the program.

If you do not specify the -p parameter while you specify the -u parameter, the ARSDOC program retrieves the password for the user ID from the stash file specified in ARS.INI file for that instance.

-U user_alias

Identifies the users when multiple users share a common user ID. The maximum length for `user_alias` is 128.

-v

Enables verbose mode, which displays all messages (informational and error). By default, the ARSDOC program displays error messages.

-x loadId

For the GET, QUERY, and UPDATE functions, use to limit the query to the set of documents that were loaded into the system under the specified `loadId`.

When you specify the `-x` parameter, use the `-f` and `-G` parameters as follows:

- Specify the `-f` parameter to search all application groups.
- Specify the `-G` parameter to search a specific application group.

To query the database to identify documents to be processed, the `-x` parameter must be used with at least one or more of the following parameters: `-i`, `-l`, `-t`, or `-T`.

Alternatively, the `-x` parameter can be used with the `-q` parameter as long as the `-q` parameter is the only additional search parameter provided.

-X loadId

For the GET function, retrieve documents by using the index file that was generated for the specified `loadId`.

For the QUERY function, build a hit list from the index file that was generated for the specified `loadId`.

For the UPDATE function, update documents by using the index file that was generated for the specified `loadId`. Because a folder is not used when the `-X` parameter is specified, default date formats are used to validate database fields when updating a document. The default formats are:

Date Type	Default Format
Date	%Y-%m-%d
Time	%X
Date/Time	%Y-%m-%d %H:%M:%S
Date/Time (TZ)	%Y-%m-%d %H:%M:%S

For the PRINT function, print documents by using the index file that was generated for the specified `loadId`.

For the CFSOD_FED function, send documents to CFS-CMOD and make them available to IBM FileNet P8 features by using the index file that was generated for the specified `loadId`.

For the FTI_ADD function, add documents to the full text index by using the index file that was generated for the specified `loadId`.

For the FTI_RELEASE function, remove documents from the full text index by using the index file that was generated for the specified `loadId`.

When you specify the `-X` parameter, you must specify the `-G` parameter and name the application group.

When you specify the `-X` parameter, you cannot specify the `-x` parameter, or the `-f`, `-i`, `-l`, `-q`, `-t`, `-S`, and `-T` parameters.

Important: The load ID that is provided with the `-X` parameter is used to identify a set of documents. When the documents are loaded, an index file is created that contains a list of all of the documents in the load. The index file is then used by ARSDOC to build a list in the

order the documents appear in the index file which is the same order the documents appear in the original input file.

Important: Any documents with index values that have been updated will not be included in the document list that is generated from the index file.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

The following ARSDOC examples include guidance on how to add a user ID to the stash file. For example, to add the user ID "sysadmin" to the stash file, enter the following command:

```
arsstash -a 1 -s stash_file -u sysadmin
```

where:

-a 1

Adds the user ID to the stash file.

-s stash_file

Name of the stash file.

-u sysadmin

User ID that is added to the stash file (if -u sysadmin is used on command line, then -p is not required since the password is obtained from the stash file).

1. The following shows how to use the GET function to retrieve documents and save them in a file in the current directory.

```
arsdoc get -h ARCHIVE -u sysadmin -f "Student Information" -o student  
-c -S 1/1/97,12/31/97 -i "WHERE student='001200340056'" -v
```

The ARSDOC program saves all of the documents that match the query in the following output file:
student

2. The following shows how to use the GET function and a parameter file to run more than one query at a time. The parameter file is in the current directory. The output files are saved in the current directory.

```
arsdoc get -h ARCHIVE -u sysadmin -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 that documents exist for all of the specified types, the ARSDOC program creates the following output files:

```
123420010056.Bills  
123420010056.Grades  
123420010056.Transcripts  
123450011917.Bills
```



```
123450011917.Grades
123450011917.Transcripts
```

- The following shows how to use the GET function to retrieve documents, write the documents to a file, and generate and save generic index data for the documents that match the query. The example shows how to specify the name of a public named query that is valid for the specified folder.

```
arsdoc get -h ARCHIVE -u sysadmin -f "Student Information"
-a -c -g -o student -q "3rd yr students GPA>3.5" -N -v
```

The number of output files that the ARSDOC program generates 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 program generates three output files. For example:

```
student.1.Student Information.TRANSSCRIPTS.ind
student.1.Student Information.TRANSSCRIPTS.out
student.1.Student Information.TRANSSCRIPTS.res
```

For AFP data, if there is more than one version of the resource group per application, then the ARSDOC program can generate additional output files. For example:

```
student.1.Student Information.TRANSSCRIPTS.ind
student.1.Student Information.TRANSSCRIPTS.out
student.1.Student Information.TRANSSCRIPTS.res
student.2.Student Information.TRANSSCRIPTS.out
student.2.Student Information.TRANSSCRIPTS.ind
student.2.Student Information.TRANSSCRIPTS.res
```

If the application group contains more than one application, then the ARSDOC program 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.2.Student Information.Grades.out
student.2.Student Information.Grades.ind
student.2.Student Information.Grades.res
```

- The following shows how to use the QUERY function to generate a list of items and save the list in a file in the current directory.

```
arsdoc query -h ARCHIVE -u sysadmin -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
```

- The following shows how to use the QUERY function and a parameter file to run more than one query at a time. The parameter file is in the current directory. The output files are saved in the current directory.

```
arsdoc query -h ARCHIVE -u sysadmin -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. The following example shows how to use the QUERY function to search a specific application group:

```
arsdoc query -h ARCHIVE -u sysadmin -i "where Date_Taken BETWEEN 9863 AND  
11531"  
-G load-scanned-images-jpeg -o jpeg -v
```

7. The following example shows how to use the ADD function to add a document to the Credit Card Statements folder:

```
arsdoc add -h ARCHIVE -u sysadmin -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
```

8. The following example shows how to use the UPDATE function to update a document in the Credit Card Statements folder, changing the balance from 123.45 to 0.00:

```
arsdoc update -h ARCHIVE -u sysadmin -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
```

9. The following example shows how to use the DELETE function to delete a document from the Credit Card Statements folder:

```
arsdoc delete -h ARCHIVE -u sysadmin -i "where account='000-000-000'  
and  
name='John Watpole'" -f "Credit Card Statements" -S 1/1/97,12/31/97  
-v
```

10. The following example shows how to use the PRINT function to send the documents that match a query to a server printer:

```
arsdoc print -h ARCHIVE -u sysadmin -P  
svrprt1 -i "where account='000-000-000' and name='John Watpole'"  
-f "Credit Card Statements" -S 1/1/97,12/31/97 -v
```

11. You can use the ARSDOC program to use the ADD function to add an index that points to an existing document. For example, assume that you loaded a bank statement for account number 000-000-000, date 5/23/97, and account name Joe Smith. You 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 ARCHIVE -u sysadmin -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
```

12. The following example shows how to use the QUERY function to limit a search to the documents that were loaded into the system under a specific load ID. For example, assume that the specified folder could be used to search several application groups; each application group contains more than one application; there are 500,000 documents in the application groups. By using the -x parameter, the

query will be limited to the set of documents that was loaded into the system under the specified load ID. Without the -x parameter, the query is run against all 500,000 documents.

```
arsdoc query -h ARCHIVE -u sysadmin -x 19867-025-0-3FAA-10136-10136
-f load-scanned-images -q query -o out -v
```

13. The following example shows how to use the QUERY function to limit a search to the documents that were loaded into the system under a specific load ID. In the example, the load ID, application group name, and query string are provided. The search is limited to the application group and only those documents that were loaded into the system under the specified load ID.

```
arsdoc query -h ARCHIVE -u sysadmin -x 19867-025-0-3FAA-10136-10136
-i "where Date_Taken BETWEEN 9863 AND 11531" -G load-scanned-images-
jpeg
-o jpeg -v
```

14. The following example shows how to use the QUERY function to limit a search to the documents that were loaded into the system under a specific load ID. In the example, the load ID, folder name, application group name, and named query are provided. The search is limited to the application group and only those documents that were loaded into the system under the specified load ID. Because a folder was specified, a named query can be used. (If an application group name is specified and a folder name is not specified, a named query cannot be used, because a named query is associated with a folder.)

```
arsdoc query -h ARCHIVE -u sysadmin -x 19867-025-0-3FAA-10136-10136
-f load_scanned-images -q query -G load-scanned-images-jpeg -o jpeg -v
```

15. The following example shows how to use the GET function and an index file to retrieve documents from the system. The index file was generated for the set of documents that was loaded into the system under the specified load ID. When using the -X parameter, documents are retrieved based on the information in the index file.

```
arsdoc get -h ARCHIVE -u sysadmin -X 19867-025-0-3FAA-10136-10136
-G load-scanned-images-jpeg -o jpeg -v
```

16. The following is an example of the GET function to retrieve public text annotations:

```
arsdoc get -u user -p passwd -h akira -f "CRD"
-q named_query -o loaddata -a -g -c -N -A 0 -v
```

17. The following example is for the -B option. You want the command line to print all of the documents in the Credit Card Statements folder that have an account number of 000-000-000. The documents are to be printed in sorted ascending order based on the balance:

```
arsdoc print -h ARCHIVE -u sysadmin -i "where account = '000-000-000'"
-f "Credit Card Statements" -P odprinter -B balance -v
```

The following is an example of the ARSDOC program command line:

```
arsdoc get -h ARCHIVE -u sysadmin -f CRD -Q SQL_filename -o loaddata -a -
g
-c -N -v
```

The following is an example file:

```
# 1 document for account number 000-000-000
where account = '000-000-000'

#1 document for account number 000-000-001 continued on multiple lines
where\
```

```
account\  
= '000-000-001'
```

```
# 1 document for account number 000-000-002  
where account = '000-000-002'
```

18. The following is an example of the GET function specifying that the SQL query strings come from a file `SQL_filename` in the current working directory.

```
arsdoc get -u user -p passwd -h akira -f "CRD"  
-Q SQL_filename -o loaddata -a -g -c -N -v
```

The `SQL_filename` file contains the following:

```
# search for account number 000-000-000  
where account = '000-000-000'  
# search for account number 000-000-001  
using multiple lines where\ account\ = '000-000-001'  
# search for account number 000-000-002  
where account = '000-000-002'
```

19. You can run ARSDOC GET with the `-a` and `-g` parameters to create generic indexer files. This also creates the `.res` resource file.

```
arsdoc get -c -h <host> -u <usr> -p <pwd> -G <ApplGrp> -F <Folder> -i  
<SQL Query> -a -g -n -o <filename> -S <data range>
```

If the `-S <data range>` is not specified, all tables are searched, and that results in poor performance.

20. HOLD_ADD function:

```
arsdoc hold_add -h ARCHIVE -u sysadmin -l hold_audit -f  
"Monthly Status Reports" -i "where code='TX' and  
ODDAT_sdate='2016-08-10'" -v
```

21. HOLD_RELEASE function:

```
arsdoc hold_release -h ARCHIVE -u sysadmin -l hold_audit -f  
"Monthly Status Reports" -i "where code='TX' and  
ODDAT_sdate='2016-08-10'" -v
```

22. CFSOD_FED function:

```
arsdoc cfsod_fed -h ARCHIVE -u sysadmin -f  
"Monthly Status Reports" -i "where code='TX' and  
ODDAT_sdate='2016-08-10'" -v
```

23. FTI_ADD function:

```
arsdoc fti_add -h ARCHIVE -u sysadmin -f  
"Monthly Status Reports" -i "where code='TX' and  
ODDAT_sdate='2016-08-10'" -v
```

24. The following example shows how to search the Content Manager OnDemand System Log:

```
arsdoc query -u oduser -h odserver -f "System Log"  
-i "where ODDAT_time_stamp > '2015-01-01 12:50:00.000000' and userid  
like 'ODUSER%'"
```

25. The following example shows a second method to search the Content Manager OnDemand System Log:

```
arsdoc query -u oduser -h odserver -f "System Log"
-i "WHERE ODDAT_time_stamp BETWEEN '2015-01-01 00:00:00.000000' AND '2015-01-06
23:59:59.999999' AND
userid LIKE 'ODUSER%'"
```

26. The following example shows a third method to search the Content Manager OnDemand System Log:

```
arsdoc query -u oduser -h odserver -f "System Log"
-i "WHERE ODDAT_time_stamp BETWEEN '2015-01-01 00:00:00.000000' AND
'2015-01-06 23:59:59.999999' AND userid LIKE 'ODUSER%'"
-S "2014-01-01 00:00:00.000000,2015-01-06 00:00:00.000000,
%Y-%m-%d %H:%M:%S.%F"
```

27. The following example shows how to use a tokenized query string:

```
arsdoc query -u oduser -h odserver -f "Credit Card Statements"
-i "2;ODDAT_crd_date,account;'2018-03-03','000-000-000';where
ODDAT_crd_date = ? and account = ?"
-S "2017-01-01,2018-04-30,%Y-%m-%d"
```

Exit codes used by ARSDOC

The ARSDOC command uses the following exit codes:

- 0** Success
- 1** No hits
- 2** Syntax error. For example, the parameter or value is invalid.
- 3** An error occurred with the specified request. For example: the user ID has an incorrect permission level, the specified folder does not exist, or the specified database field does not exist.
- 4** Unrecoverable error. For example, the operation failed, or the user cannot connect to the server.

Notes

The ADD function will fail unless the Database Organization for the application group named with the -g parameter is Multiple Loads per Database Table and the Expiration Type is Segment or Document.

The ADD function can be run without providing an input document from a file or by retrieving an existing document from the system. This means that you can add database field values without adding a document. To add database field values without adding a document, do not specify the -o, -i, or -q parameters; specify the database field names and their values using one or more -n parameters.

The ARSDOC program can issue an informational message that provides the server version number. When you run the ARSDOC GET function, messages can also be included that provide the number of documents that were queried and retrieved, and a status message for each document that is retrieved. To enable these messages, specify the -v parameter for verbose mode which displays all messages (information and error). If you do not specify the -v parameter, the ARSDOC program displays only error messages.

Files

/usr/lpp/ars/V10R1M0/bin/arsdoc

The z/OS executable program.

Chapter 22. ARSEXOAM

Purpose

The ARSEXOAM program is used to process the rows in the ARSOAM_DELETE table that indicate that Content Manager OnDemand OAM objects have expired, and have the associated table entries for those objects removed.

Note:

1. If one object for a given load ID is deleted, all of the index entries for that load ID are deleted.
2. Index entries of all OAM objects that are recorded as being deleted by rows in the ARSOAM_DELETE table are deleted regardless of the settings in the Life of Data and Indexes section on the Storage Management page of the application group. If you plan to use Storage Management expiration, ensure that you set the expiration type of all application groups to Storage Manager.

Input and output

The ARSEXOAM program uses the following input:

- The ARSOAM_DELETE table.
- The `ars.ini` file.
- ODBC initialization file (DSNAOINI).
- A control data set, which contains parameters that control processing. For example, the Content Manager OnDemand instance name.

The ARSEXOAM program uses the following output:

- A message data set that contains any informational or error messages.
- A data set used by the run-time library for error messages.
- A data set used by the run-time library for additional diagnostic information.

Control

You control the ARSEXOAM program with job and utility control statements. The job control statements run the ARSEXOAM program and define the data sets that are used and produced by the program. The utility control statements control the functions of the ARSEXOAM program.

Job control statements

The table shows the job control statements for the ARSEXOAM program.

<i>Table 9: Job control statements for the ARSEXOAM program</i>	
Statement	Use
JOB	Starts the job.
EXEC	Specifies the program name.
STEPLIB	Specifies the location of the SARSLOAD, SDSNEXIT, and SDSNLOAD data sets.

Statement	Use
DSNAOINI	Specifies the location of the ODBC Initialization file. If the DB2 subsystem which contains the ARSOAM_DELETE table is not the same DB2 subsystem as the Content Manager OnDemand system tables, DSNAOINI DD must specify a different file than the one ARSSOCKD uses.
SYSPRINT	Defines a sequential data set for messages.
SYSOUT	Defines a sequential data set for messages from the Language Environment runtime.
CEEDUMP	Defines a sequential data set for diagnostic dumps from the Language Environment run-time.
ARSPARM	Defines the control data set. The control data set normally resides in the input stream, however, it can be defined as a member of a partitioned data set or PDSE.

Utility control statements

ARSEXOAM is controlled by utility control statements. The statements must appear one per line, with no line numbers. Control statements cannot be continued across lines. The statements consist of a single keyword-value pair. The table lists the keywords.

Keyword	Use
ODINSTANCE	Specifies the Content Manager OnDemand instance name for which objects are to be expired. This parameter is required and must specify an instance name in the <code>ars.ini</code> file and not a host name.
ODUSER	Specifies the Content Manager OnDemand user ID to use for processing. <ul style="list-style-type: none"> • If unified login is enabled (that is, the ARS.PTGN dynamic exit has been enabled), the user ID returned by USS will be used; therefore the ODUSER can be omitted. • The user ID must have permission to delete objects from the application group.
ODUSERPW	Specifies the password associated with ODUSER. Note: If unified login is enabled (that is, the ARS.PTGN dynamic exit has been enabled), a Passticket will be used; therefore, the ODUSERPW may be omitted.
COMMITCNT	Specifies the number of fetches from the ARSOAM_DELETE, ARSOD, and ARSODIND tables that are to be done between COMMITS. If not specified, 1000 is used. If zero (0) is specified, no commits are done while fetching. The ARSOD and ARSODIND table are only processed if Content Manager OnDemand for OS/390 Version 2 migrated index rows are being deleted.
UNLOADMAX	Specifies how many objects to hold in memory at any one time. The default is 100,000.
REQLIMIT	Specifies the maximum number of objects to send to the server in each request. This number defaults to the ARS_EXPIRE_REQLIMIT parameter in the <code>ars.cfg</code> , or 100 if ARS_EXPIRE_REQLIMIT is not specified.

Example

The following example shows a sample job ARSEXOAM.

```
//ARSEXOAM JOB (account),userid,MSGCLASS=A
//
//STEP1 EXEC PGM=ARSEXOAM,REGION=0M
//STEPLIB DD DISP=SHR,DSN=ARS.V10R1M0.SARSLoad
// DD DISP=SHR,DSN=DB2.V10R1M0.SDSNEXIT
// DD DISP=SHR,DSN=DB2.V10R1M0.SDSNLOAD
//DSNAOINI DD PATH='/opt/IBM/ondemand/V10R1M0/config/cli.ini'
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//CEEDUMP DD SYSOUT=*
//ARSPARM DD *
odinstance=ARCHIVE
oduser=arsuser
oduserpw="password"
```

Figure 13: A sample job ARSEXOAM

Chapter 23. ARSEXP

Purpose

The ARSEXP program can be used to process SMF records that indicate Content Manager OnDemand objects have expired, and have the associated index entries for those objects removed.

The program uses the SMF type 65, type 85, and the user type records written by ARSSMFWR to determine which objects have been deleted. The ARSEXP program then has the server remove the index entries.

Note:

1. If one object for a given load ID is deleted, all of the index entries for that load ID are deleted.
2. Index entries of all objects that are recorded as being deleted by the SMF records are deleted regardless of the settings in the Life of Data and Indexes section on the Storage Management page of the application group. If you want to use Storage Management expiration, then ensure that you set the expiration types of all application groups to Storage Manager.

Input and output

The ARSEXP program uses the following input:

- An input data set that contains the SMF data including SMF types 65, 85, and the user-defined type if using ARSSMFWR. The input may contain other SMF record types, however any other types will be ignored.
- A control data set, which contains parameters that control processing. For example, the Content Manager OnDemand instance name.

The ARSEXP program uses the following output:

- A message data set that contains any informational or error messages produced by the program.
- A data set used by the run-time library for error messages.
- A data set used by the run-time library for additional diagnostic information.

Control

You control the ARSEXP program with job and utility control statements. The job control statements run the ARSEXP program and define the data sets that are used and produced by the program. The utility control statements control the functions of the ARSEXP program.

Job control statements

The table shows the job control statements for the ARSEXP program.

Statement	Use
JOB	Starts the job.
EXEC	Specifies the program name (PGM=ARSEXP).
STEPLIB	Specifies the location of the SARSLOAD data set.

Statement	Use
SYSPRINT	Defines a sequential data set for messages.
SYSOUT	Defines a sequential data set for messages from the Language Environment run-time.
CEEDUMP	Defines a sequential data set for diagnostic dumps from the Language Environment run-time.
ARSPARM	Defines the control data set. The control data set normally resides in the input stream; however, it can be defined as a member in a partitioned data set or PDSE.
SMFIN	Defines the data set containing the SMF data. This data set is created by the IFASMFDP program using data dumped from the SYS1 .MANx data sets. See <i>z/OS MVS System Management Facilities (SMF)</i> , SA22-7630 for additional information.

Utility control statements

ARSEXPDIR is controlled by utility control statements. The statements must appear one per line, with no line numbers permitted. Control statement cannot be continued across lines. The statements consist of a single keyword=value pair. The table lists the keywords.

Keyword	Use
ODINSTANCE	Specifies the Content Manager OnDemand instance name for which objects are to be expired. This parameter is required.
ODUSER	Specifies the Content Manager OnDemand user ID to use for processing. <ul style="list-style-type: none"> • If unified log in is enabled (that is, the ARS.PTGN dynamic exit has been enabled), the user ID that is returned by USS will be used; therefore, the ODUSER may be omitted. • The userid must have permission to delete objects from the application group.
ODUSERPW	Specifies the password associated with ODUSER. <p>Tip: If unified login is enabled (that is, the ARS.PTGN dynamic exit has been enabled), the password that is returned by USS will be used; therefore, the ODUSERPW may be omitted.</p>
USERSMF	Specifies the SMF record type that is written by the ARSSMFWR exit (if used). This parameter may be omitted if ARSSMFWR is omitted. See <i>IBM Content Manager OnDemand for z/OS: Configuration Guide</i> for information about the ARSSMFWR exit.
UNLOADMAX	Specifies how many objects to hold in memory at any one time. The default is 100,000.
COMMITCNT	Specifies the number of fetches from the ARSOD and ARSODIND table that are to be done between COMMITS. If this number is not specified, 1000 is used. If this number is zero (0), no commits are done while fetching. This parameter is used only if Content Manager OnDemand for OS/390 Version 2 migrated index rows are being deleted.
REQLIMIT	Specifies the maximum number of objects to send to the server in each request. This defaults to the ARS_EXPIRE_REQLIMIT parameter in the ars.cfg, or 100 if ARS_EXPIRE_REQLIMIT is not specified.

Example

The following example shows example JCL to run the ARSEXPPIR program.

```
//ARSEXPPIR JOB (TSS40000,B123,99999), 'User',MSGCLASS=1,CLASS=C,
//          MSGLEVEL=(1,1),NOTIFY=&SYSUID,TIME=1440
//PROCESS  OUTPUT  DEFAULT=YES,CLASS=*,JESDS=ALL,OUTDISP=HOLD
//*****
//STEP1    EXEC PGM=ARSEXPPIR,REGION=0M
//STEPLIB DD DISP=SHR,DSN=ARSV710.ODMP710.SARSLoad
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//CEEDUMP  DD SYSOUT=*
//SMFIN    DD DISP=SHR,DSN=SYSQ.SMF.MAN1
//ARSPARM  DD *
           odinstance=ARCHIVE
           usersmf=193
/*
```

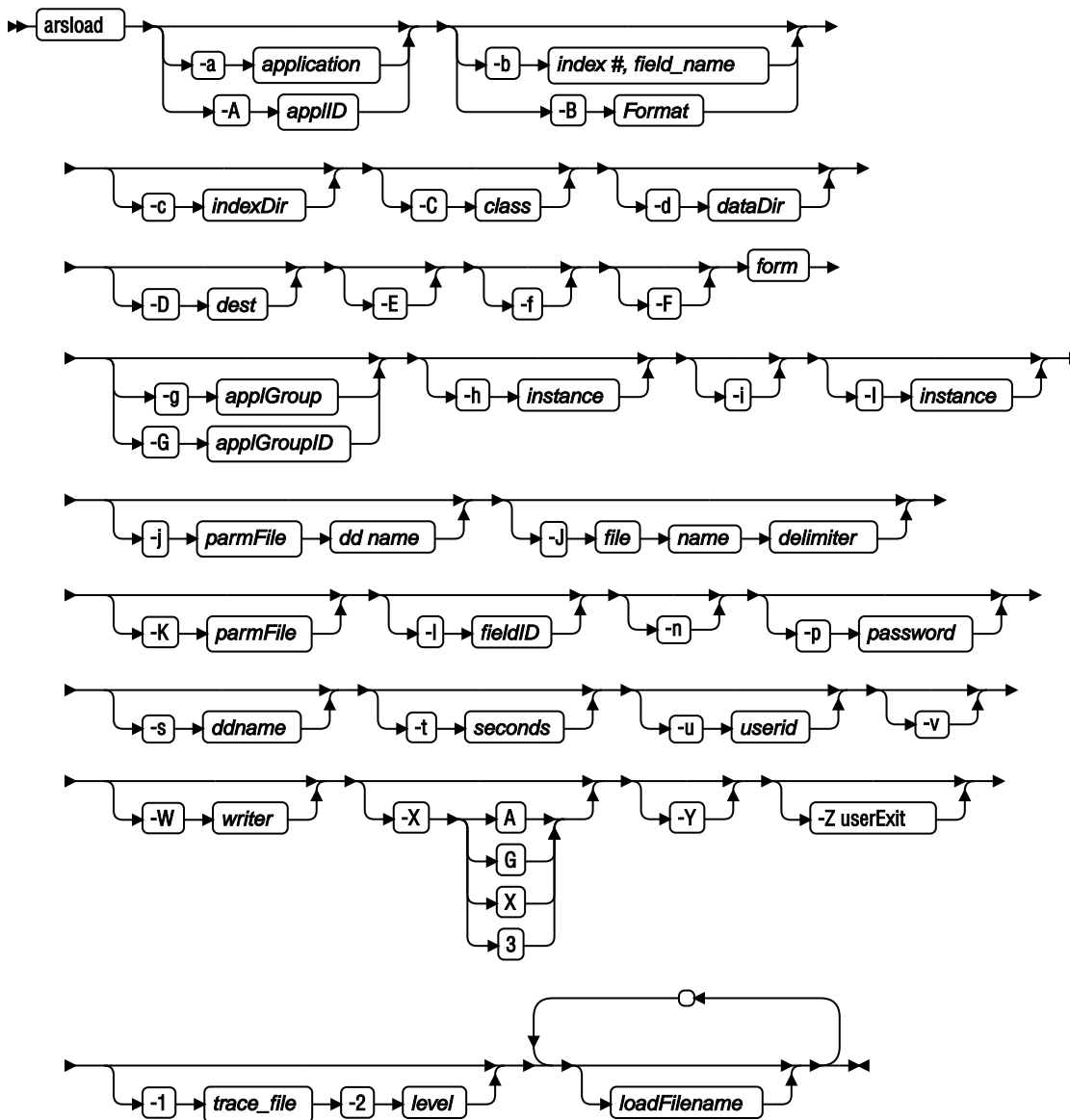
Figure 14: Example JCL to run the ARSEXPPIR program

Chapter 24. ARSLOAD

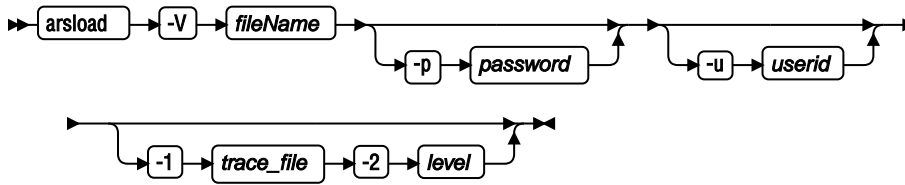
Purpose

The ARSLOAD program can be used to process the input files that you want to load into the system. The ARSLOAD program determines if the input data needs to be indexed, and if so, calls the appropriate indexing program. The ARSLOAD program calls the storage manager programs to load report data on storage volumes and the database manager to update the IBM Content Manager OnDemand database with the index information that was extracted from or specified for the input file.

Syntax



File input. Reads the parameters from a specified file. If the optional -u and -p are specified on both the command line and in the file, the values in the parameter file will be used. Make sure that you specify the -V parameter as the first keyword.



Description

The ARSLOAD program is the main Content Manager OnDemand data indexing and loading program. The ARSLOAD program calls the indexing program if the input data needs to be indexed, creates input files for the storage manager, and updates the Content Manager OnDemand database. The ARSLOAD program saves processing messages in the system log. You can open the System Log folder and list the messages that were generated when an input file was processed.

You typically configure the ARSLOAD program to run as a started task, to periodically check the spool for SYSOUT data sets to process.

Note: The ARSLOAD program can use the following sources for input files to process:

- one or more classes specified with the -C parameter,
- one or more file systems specified with one or more -d parameters,
- a destination that is specified with the -D parameter,
- a FORM name that is specified with the -F parameter,
- a WRITER name that is specified with the -W parameter, or
- one or more load file names.

If you omit the load file name, the ARSLOAD program will run in daemon mode and attempt to load input data from the JES spool (if any of the JES spool options -C, -D, -F, or -W were specified) or poll the directories that are specified by the -d parameter for input data to process. If you omit the load file name and you do not specify at least one of the daemon mode parameters (-C, -D, -F, -W, or -d), the ARSLOAD program will issue a usage note and exit.

Normally, ARSLOAD uses TCP/IP to communicate with the server. This provides a lot of flexibility in where ARSLOAD is run and under which RACF user and group IDs it runs. However, if ARSLOAD is running on the same system as the server, it can manipulate the Content Manager OnDemand database directly, and store objects into OAM or VSAM from the ARSLOAD program if the following conditions are met:

- ARSLOAD must be running with the same RACF user and group ID as the server.
- ARSLOAD must be running on the same system as the server.
- The -h parameter must specify the instance name of the server in the `ars.ini` file and not a hostname. The instance name must also be 16 characters or less.
- A DSNAOINI DD must be present.

You can force ARSLOAD to use TCP/IP even if all the above are true by specifying the `ARSMVS_ARSADMIN_USETCPIP=1` environment variable. If any of the conditions above is not met, ARSLOAD uses TCP/IP regardless of the setting of `ARSMVS_ARSADMIN_USETCPIP`.

When you run the ARSLOAD program, you must provide the user ID and password of a Content Manager OnDemand user with administrator authority for the application group into which the input data will be loaded. There are several ways that you can provide the userid and password:

- Specify the -u and -p parameters each time that you run the ARSLOAD program.
- Assign a userid to the ARSLOAD started task via the RACF STARTED class and use the ARS.PTGN exit to provide a PassTicket. See *z/OS SecureWay Security Server RACF System Programmer's Guide*, SA22-7683 for information about the RACF STARTED class. See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for information about the ARS.PTGN exit and creating PassTickets.

You can run the ARSLOAD program from a z/OS server. The data is automatically indexed using the OS/390 indexer, and then loaded to a Multiplatforms server.

Important: After you upgrade to Content Manager OnDemand Version 10.1, keep in mind that ARSLOAD now automatically excludes duplicate rows during load (thereby lessening the count for the number of rows loaded by ARSLOAD). Meaning, if duplicate rows are present, then only one is loaded.

Parameters

-a application

The name of the application to load. If the application group contains more than one application, then you must identify the application to load. Otherwise, the load will fail. If you plan to automate the loading of data into different application groups and applications, then use the -A parameter to specify the component of the input file name that identifies the name of the application.

-A applID

Determines the component of the input file name that identifies the application to load. If the application group contains more than one application, then you must identify the application to load. Otherwise, the load will fail. You typically use this parameter when you run the ARSLOAD program as a started task to automate the loading of data into different application groups and applications.

- By default, the ARSLOAD program uses the FORM component of the input file name to identify the application group to load. You can use the -G parameter to specify a different component of the input file name that identifies the application group. For example, `arsload -G JOBNAME`.
- If the application group to load contains more than one application (source of data), then you must identify the application to load. Otherwise, the load will fail. For example, to use the DATASET component of the input file name to identify the application, run the ARSLOAD program with the -A DATASET parameter. For a started task or job, DATASET is the DD name that defines the input file.

Choose one of the WRITER, MVS, JOBNAME, DATASET, or FORM components of the input file name to identify the application.

-b <index #,field_name>

Specify the name of an index field and a number that identifies the part of the input file name that the ARSLOAD programs assigns to that index field. For *index_number*, you can specify 1 - 8 and the number must match a part of the input file name identified by the -B parameter.

In the following example, assume that the input file name is `App1Group1.App1.ARD`:

```
arsload -b "1,field1" -B "AG.1APP.ARD" -g App1Group1 ...
```

During the load, the ARSLOAD program assigns the value `App1` from the input file name to the index field "field1" of application group "App1Group1".

-B "format"

Use this parameter to do the following tasks:

- Define the file name formats for MVS download files and files processed by the ARSLOAD daemon.
- Identify the parts of the file name that the ARSLOAD program assigns to an index field specified by the -b parameter.

Use the following identifiers to define the file name format:

1 - 8

A number prefixed to a part of the file name that identifies that part as a value to assign to an index field.

AG

Application group name

APP

Application name

DAT

Abbreviation for DATASET

EXT

Extension

FLD

Field place holder for indexing

FOR

Abbreviation for FORMS

HH

Abbreviation for HHMMM

IGN

Ignore

JOB

Abbreviation for JOBNAME

MVS

MVS

WRI

Abbreviation for WRITER

YY

Abbreviation for YYDDD

The following example uses the full word of the JOB, DAT, FOR, YY, and HH identifiers to define the file name format:

```
MVS . JOBNAME . DATASET . FORMS . YYDDD . HHMMM . ARD
```

The following example shows the same file name format with the abbreviated words:

```
MVS . JOB . DAT . FOR . YY . HH . EXT
```

The following example shows the same file name format and indicates that the ARSLOAD program assigns the JOB part of file name to index field 1:

```
MVS . 1JOB . DAT . FOR . YY . HH . EXT
```

The following examples show different ways that you can define the file name formats:

```
ARSLOAD -G FORMS -D "-" -B "WRITER-MVS-IGN-IGN-FORMs"
ARSLOAD -G JOBNAME -B "WRITER.IGN.JOB.IGN.DAT.IGN.EXT"
ARSLOAD -D "-" -B "IGN-IGN-AG-APP-IGN.EXT"
ARSLOAD -B "APP.IGN.AG.IGN.IGN"
```

The following examples show how to identify the parts of the file name to assigns to an index field:

- Single index format, where the ARSLOAD daemon assigns the DAT part of the file name to index field `account_num`:

```
arsload -b "1,account_num" -B "AG.APP.1DAT"
```

- Multiple index format, where the ARSLOAD daemon assigns the different parts of the file name to different index fields:

```
arsload -b "1,report_name" -b "2,run_date" -b "3,app_name" -B "AG.3APP.1FLD.2YY"
```

The following table maps the index field name to the part of the file name

Index field name	Part of the file name
report_name	FLD
run_date	YY
app_name	APP

You can use the `-B` parameter or combine the `-g` and `-a` parameters, but you cannot use all three of them together. If you specify the `-B` parameter, you identify the application group name and the application name as parts of the file name.

-c indexDir

The file system in which Content Manager OnDemand temporarily stores data created by the indexing program. The default location is the directory from which the ARSLOAD program was invoked.

-C class

Specifies the JES SYSOUT class or classes that the ARSLOAD program monitors for input data to process when running as a started task. You can specify up to 36 classes. The classes should be specified as a string of characters. For example: `-C A` or `-C ACDFLPQ`.

-d dataDir

Specifies an HFS file system to monitor for input data to process.

Restriction: If you specify the `-C`, `-F`, or `-W` parameters, the `-d` parameter is ignored.

The name of an input file residing in this file system must conform to the following format: `WRITER.MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSS.ARD`

Important: The `.ARD` file name extension is required to initiate a load process, and all associated files must have `.ARD` as part of the filename.

By default, the ARSLOAD program uses the `FORMS` component of the file name to identify the application group to load. However, you can use the `-G` and `-A` parameters to specify a different component of the file name (`WRITER`, `MVS`, `JOBNAME`, `DATASET`, or `FORMS`) to identify the application group and application to load.

Restriction: The ARSLOAD program cannot invoke an indexing program when the `-d` parameter is specified. Therefore, the input data to be loaded must have been indexed. The `WRITER.MVS.JOBNAME.DATASET.FORMS.ARD.ind`, `WRITER.MVS.JOBNAME.DATASET.FORMS.ARD.out`, and if applicable, `WRITER.MVS.JOBNAME.DATASET.FORMS.ARD.res` files produced by the indexer must exist in the directory specified by the `-d` parameter before the ARSLOAD program is started.

Not all indexers can accept an input file from the HFS. Files should not be placed in a `-d` directory that require one of those indexers, such as `ACIF`.

You can specify the `-d` parameter more than once, to specify multiple file systems for the ARSLOAD program to process.

You cannot specify the `-n` or `-v` parameters when you specify the `-d` parameter.

-D dest

Allows the specification of a destination to use for SYSOUT selection. If specified, the ARSLOAD program will select and process only the SYSOUT for the specified destination.

-E

When specified, exits that are associated with the ARS.RSADUPDT logical exit point are to be invoked. If the -E parameter is not specified, then the exit routines are not invoked. The exit routines may be used to modify some of the parameters that are used by Content Manager OnDemand when document data is being captured (loaded) by the ARSLOAD program, such as the name of the application group, application, or storage node into which the data will be loaded. For more information, see *Report Specifications Archive Definition Exit* in the *IBM Content Manager OnDemand for z/OS: Configuration Guide*.

-f

Use to unload the data if the load process fails. If the database manager step fails, then Content Manager OnDemand should remove any index data that was added to the database. If the storage manager step fails, then Content Manager OnDemand should remove any storage objects that were copied to storage volumes.

Tip: If an input file fails to load, you should review the message log that was created during the load process. You can retrieve a message log from the system log. If the message log contains a Load ID, then it means that for some reason, Content Manager OnDemand stored at least some of the input data in the application group. Before you attempt to reload the input data, you must remove the data that was created during the failed load process by using the UNLOAD function of the ARSADMIN program. See Chapter 17, “ARSADMIN,” on page 119 for help with removing the data that is created when a file is loaded into the system.

-F form

Specifies the JES FORM name that the ARSLOAD program monitors for input data to process when running as a started task.

-g applGroup

The name of the application group to load. This parameter is required if you specify a load file name to process. This parameter is optional if you specify the -d parameter. If you specify the -d parameter, then by default, the ARSLOAD program uses the FORM component of the input file name to determine the name of the application group to load. If you plan to automate the loading of input files into different application groups and applications, then you should use the -G parameter to specify the component of the input file name that identifies the application group to load.

-G applGroupID

Determines the component of the input file name used to identify the name of the application group to load. You typically use this parameter when you run the ARSLOAD program as a started task to automate the loading of input files into different application groups and applications.

- By default, the ARSLOAD program uses the FORM part of the file name to identify the application group to load. You can use the -G parameter to specify a different part of the file name that identifies the application group. For example, `arsload -G JOBNAME`.
- If the application group to load contains more than one application (source of data), then you must identify the application to load. Otherwise, the load will fail. For example, to use the DATASET part of the file name to identify the application, run the ARSLOAD program with the -A DATASET parameter. For a started task or job, DATASET is the DD name that defines the input file.

Choose one of the WRITER, MVS, JOBNAME, DATASET, or FORM parts of the file name to identify the application group.

-h instance

The name of the Content Manager OnDemand instance to process. The ARSLOAD program will attempt to locate the specified instance name in the ARS.INI file, to obtain the TCP/IP address, host name or host name alias of the system on which the instance is running. If the ARSLOAD program cannot locate the instance name in the ARS.INI file, then the specified value is treated as a host name. **Note:** The ARSLOAD program also supports the -I parameter. You can use either parameter to specify the name of the instance to process. If you specify both, the value of the last parameter specified is used.

You must specify this parameter and name the instance if:

- The name of the default instance is not ARCHIVE.
- You are running more than one instance on the same system and you want to process an instance other than the default instance.
- You are running the ARSLOAD program on a system other than the system on which the instance that you want to process is running.

Note: If you are running multiple instances of Content Manager OnDemand on the same system, then you should always specify the -h parameter to identify the name of the instance that you want to process. You should also verify that the ARS.INI file is configured with the correct information for all instances of Content Manager OnDemand. See the *IBM Content Manager OnDemand for z/OS: Configuration Guide* for more information about working with multiple instances of Content Manager OnDemand.

-i

Use to run the data indexing program only; do not copy report data to storage volumes or add the index data to the database.

-I instance

The name of the Content Manager OnDemand instance to process. The ARSLOAD program will attempt to locate the specified instance name in the ARS.INI file, to obtain the TCP/IP address, host name or host name alias of the system on which the instance is running. If the ARSLOAD program cannot locate the instance name in the ARS.INI file, then the specified value is treated as a host name. **Note:** The ARSLOAD program also supports the -h parameter. You can use either parameter to specify the name of the instance to process. If you specify both, the value of the last parameter specified is used.

You must specify this parameter and name the instance if:

- The name of the default instance is not ARCHIVE.
- You are running more than one instance on the same system and you want to process an instance other than the default instance.
- You are running the ARSLOAD program on a system other than the system on which the instance that you want to process is running.

If you are running multiple instances of Content Manager OnDemand on the same system, then you should always specify the -I parameter to identify the name of the instance that you want to process. You should also verify that the ARS.INI file is configured with the correct information for all instances of Content Manager OnDemand. See the *IBM Content Manager OnDemand for z/OS: Configuration Guide* for more information about working with multiple instances of Content Manager OnDemand.

-j parmFile

Use to specify the name of a file that contains additional indexing parameters. Each line (or record) in the file represents one complete indexing parameter, including keywords and data values.

When you specify the -j parameter and name a file, the ARSLOAD program adds the indexing parameters from the specified file to the indexing parameters that it extracts from the application. (Indexing parameters are typically specified on the Indexing Information page in the application.) If an indexing parameter appears in both the application and the file that you specify, then the ARSLOAD program uses the value from the file.

For IBM Content Manager OnDemand for z/OS, if the parmFile starts with //DD:name, "name" specifies the name of a DD statement in the ARSLOAD JCL. Otherwise, the parmFile specifies a file in the Unix System Services file system.

The following shows an example of a file that contains additional indexing parameters to be processed by ACIF:

```
FORMDEF=F1ABBB0
PAGEDEF=P1ABBB0
USERLIB=HLQ.SAPKULI1
```

The following shows an example of the -j "//DD:name" parameter, and some instream indexing instructions:

```
-j "//DD:PD"
    |
    //PD DD *
    FIELD5=X'F1F061F3F161F0F5' /* 10/31/05 */
    INDEX6=X'D7D6E2E3C9D5C76DC4C1E3C5',FIELD5,
    (TYPE=GROUP,BREAK=NO)
```

-J file name delimiter

Use this parameter to define file name formats for MVS download files and files that are processed by the ARSLOAD daemon. By default, this parameter is "."

-K

Specifies that ARSLOAD should not detach ARSADMIN. Normally, ARSLOAD attaches and detaches ARSADMIN several times for each file loaded. If you specify -K, ARSLOAD does not detach ARSADMIN. This may improve performance by not requiring the LE enclave associated with ARSADMIN to be constantly created or destroyed. In addition, if ARSLOAD is running without TCP/IP, the DB2 connection can persist.

Restriction: Do not specify the -K parameter if you are using any 390 Indexer exits until you verify that the exits function correctly in this environment. In particular, the LE enclave is no longer terminated between reports. This impacts exits that rely on enclave termination to perform cleanup, for example, closing files.

-l fieldID

Indicates that the ARSLOAD program will capture and place the entire name of the input file into the field identifier specified in the *fieldID* variable.

-n

Determines whether Content Manager OnDemand deletes the input files when the ARSLOAD program ends.

- If you specify the -n parameter, then Content Manager OnDemand does not delete the input files when the ARSLOAD program ends.
- If you do not specify the -n parameter, then Content Manager OnDemand deletes the input files when the ARSLOAD program ends.

In either case, if the ARSLOAD program fails in the load step because of a device or system problem, then you can restart the load step after you correct the problem by using the intermediate files that were created by the ARSLOAD program. The ARSLOAD program stores the intermediate files in the directory named with the -c parameter (or the directory from which you started the ARSLOAD program, if you did not specify the -c parameter). The intermediate files have the same name 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, specify -p "").

-s ddname

Specifies a DD name in the JCL that identifies the input data set for ACIF, the OS/390, or the IBM Content Manager OnDemand PDF Indexer for Multiplatforms. Not supported by the Generic indexer. This parameter should be specified when the ARSLOAD program is invoked with a load file name.

The -s parameter causes the ARSLOAD program to pass the associated DD name to the indexing program as the value of the INPUTDD parameter. Because none of the indexers can accept input from the HFS (except for the Generic indexer, which really does not perform indexing), you must specify the -s parameter. For the Generic indexer, input files must be in the HFS; they cannot be specified by using the -s parameter.

-t seconds

Determines the polling time in seconds. This is the interval of time in which the ARSLOAD program checks the input directories for input files to process. The default value is 600 seconds, which means that the ARSLOAD program checks the input directories every ten minutes.

-u userid

The userid of a Content Manager 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, and the ARS.PTGN dynamic exit has been enabled, your user ID as returned by UNIX System Services will be used, provided that you also omit the -p parameter.

The *IBM Content Manager OnDemand for z/OS: Configuration Guide* describes how to enable the ARS.PTGN exit.

-V fileName

Specifies the name of the file that contains the keywords and values that are to be used by ARSLOAD. Each line is used for a single keyword or value. Trailing blanks are stripped, and lines should not have sequence numbers. Single and double quotation marks in lines that start and end with single or double quotation marks (" or ') are removed. The first line becomes the first keyword or value, and the second line becomes the second keyword or value, continuing until the input file is exhausted. The parmfile is in the format acceptable to the fopen() C-runtime function call. For example, to specify a DD statement called PARM, you can use //DD:PARM

-v

Enables verbose mode, which displays all messages (informational and error). By default, the ARSLOAD program displays error messages.

-W writer

Specifies the JES WRITER that the ARSLOAD program monitors for input files to process when running as a started task.

-X indexer

Allows you to override the indexing program that was specified on the Indexer Information page in the application.

The possible values are A, G, X, and 3:

- Specify -X A to use ACIF.
- Specify -X G to use the Content Manager OnDemand Generic indexer.
- Specify -X X to use the Content Manager OnDemand XML indexer.
- Specify -X 3 to use the Content Manager OnDemand OS/390 indexer.

You must specify an indexing program if the input contains index data in a format other than the one supported by the indexing program that was specified for the application. For example, assume that you defined an application and specified ACIF as the indexing

program. Later, you used the ARSDOC program to extract documents from an application group. The ARSDOC command generates index data that is in the Generic indexer format. To load the index data into the application, you must specify:

```
arsload -X G . . .
```

-Y

Specifies that ARSLOAD is performing a migration load. You must specify this parameter for migration loads to ensure that processing that is specific to a migration load is performed.

-Z userExit

Specifies that ARSLOAD is performing a migration load. You must specify this parameter for migration loads to ensure that processing that is specific to a migration load is performed.

loadFilename

Specifies an input file to process.

You may specify the names of one or more input files to process. If you specify more than one input file, separate the file names with a blank character.

The ARSLOAD program appends the following file type extensions to the file name that you specify: .ind, .out, .res, .ann, and .ang

- The .ind file contains the index data
- The .out file contains the report data
- The .res file contains the resource data
- The .ann file contains text annotations
- The .ang file contains graphic annotations

For example, if you specify `arsload -g BILLS po3510`, the ARSLOAD program looks for and processes the following files:

- po3510.ind
- po3510.out
- po3510.res
- po3510.ann
- po3510.ang

The file extensions on all .ind, .out, .res, .ann, and .ang files must be lowercase. File names with extensions .IND, .OUT, .RES, .ANN, and .ANG will not be processed.

If the ARSLOAD program does not find a file with the .ind file type extension, it automatically calls the indexing program to process the input file.

If you specify the -s parameter, then specify /dev/null for the loadFilename parameter.

Tip: If you do not specify a load file name, the ARSLOAD program can use the following sources for input files to process:

- one or more classes specified with the -C parameter,
- one or more file systems specified with one or more -d parameters,
- a destination that is specified with the -D parameter,
- a FORM name that is specified with the -F parameter, or
- a WRITER name that is specified with the -W parameter.

If you omit the load file name, the ARSLOAD program will run in daemon mode and attempt to load input data from the JES spool (if any of the JES spool options -C, -D, -F, or -W were specified) or poll the directories that are specified by the -d parameter for input data to process. If you omit the load file name and you do not specify at least one of the daemon

mode parameters (-C, -D, -F, -W, or -d), the ARSLOAD program will issue a usage note and exit.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

1. The following shows how to run the ARSLOAD program to check class Q for input files to process. The ARSLOAD program stores temporary work files in the location specified with the -c parameter. By default, the ARSLOAD program uses the FORMS part of the file name to determine the application group to load, and the application group contains only one application.

```
arsload -c /arsacif/acif1 -u admin -C Q
```

2. The following shows how to run the ARSLOAD program to check class Q for input files to process. The ARSLOAD program stores temporary work files in the directory specified with the -c parameter. The ARSLOAD program will use 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.

```
arsload -c /arsacif/acif1 -u admin -C Q -A DATASET -G JOBNAME
```

Sample JCL

The following figure shows an example of JCL that you can submit to start the ARSLOAD program in a UNIX System Services environment. The parameters for the ARSLOAD program are provided by using the PARM keyword on the EXEC statement.

```
//USERLOAD JOB (TSS40000), 'user', MSGCLASS=1, CLASS=A,
//          MSGLEVEL=(1,1), NOTIFY=&SYSUID, TIME=1440
//PROCESS  OUTPUT  DEFAULT=YES, CLASS=*, JESDS=ALL, OUTDISP=HOLD
//*****
//STEP1    EXEC PGM=ARSLOAD, REGION=0M,
// PARM=('/-h ARCHIVE -u odadmin -p passwd -n -g appGrp -s INPUT
//          afp ')
//STEPLIB DD DSN=ARS.V10R1M0.SARSLOAD, DISP=SHR
//* DD DSN=DB2.V10R1M0.SDSNEXIT, DISP=SHR
//* DD DSN=DB2.V10R1M0.SDSNLOAD, DISP=SHR
//          DD DSN=ACIF.V2R3M1.SAPKMOD1, DISP=SHR
//*DSNA0INI DD PATH='/opt/IBM/ondemand/V10R1M0/config/cli.ini'
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//INPUT    DD DSN=PSF.SAPKSAM3(APKBANK), DISP=SHR
//CEEDUMP  DD SYSOUT=*
```

Figure 15: Sample JCL for ARSLOAD

The following figure shows the same example but specifies the parameters from a file instead of the PARM keyword on the EXEC statement:

```
//USERLOAD JOB (TSS40000), 'user', MSGCLASS=1, CLASS=A,
//          MSGLEVEL=(1,1), NOTIFY=&SYSUID, TIME=1440
//PROCESS  OUTPUT  DEFAULT=YES, CLASS=*, JESDS=ALL, OUTDISP=HOLD
//*****
//STEP1    EXEC PGM=ARSLOAD, REGION=0M,
// PARM=('/-V //DD:PARAM -u odadmin -p passwd')
```

```

//STEPLIB DD DSN=ARS.V10R1M0.SARSLOAD,DISP=SHR
//* DD DSN=DB2.V10R1M0.SDSNEXIT,DISP=SHR
//* DD DSN=DB2.V10R1M0.SDSNLOAD,DISP=SHR
//
// DD DSN=ACIF.V2R3M1.SAPKMOD1,DISP=SHR
//*DSNAOINI DD PATH='/opt/IBM/ondemand/V10R1M0/config/cli.ini'
//SYSPRINT DD SYSOUT=* //SYSOUT DD SYSOUT=*
//INPUT DD DSN=PSF.SAPKSAM3(APKBANK),DISP=SHR
//CEEDUMP DD SYSOUT=*.
//PARM DD *
-h
ARCHIVE
-n
-g
appGrp
-s
INPUT
afp

```

About the JCL statements

The JCL statements are explained as follows.

//STEP1 EXEC PGM=ARSLOAD ...

The EXEC statement specifies the name of the ARSLOAD program.

// PARM=('/-h ARCHIVE -u odadmin -p passwd -n -g appGrp -s INPUT afp')

The PARM keyword on the EXEC statement specifies the parameters for the ARSLOAD program. See [“Parameters” on page 187](#) for information about the parameters that you can specify. The example specifies the following parameters:

Important: The leading / in the PARM= keyword is important, as this ensures that the Language Environment will not attempt to interpret the parameters as Language Environment runtime options.

-h ARCHIVE specifies the name of the Content Manager OnDemand instance to process. The system locates the specified instance name in the ARS.INI file to determine the TCP/IP address, host name alias, or fully-qualified host name of the library server and other configuration information for the instance. The example -h parameter specifies an instance name of ARCHIVE.

-u odadmin specifies the Content Manager OnDemand userid that has administrator authority to load documents into the application group.

-p passwd specifies the password for the Content Manager OnDemand userid that was specified with the -u parameter.

-n specifies that the ARSLOAD program should keep the input files after successfully completing processing.

-g appGrp specifies the name of the Content Manager OnDemand application group to load. The name of the application group must be specified exactly as it exists in the Content Manager OnDemand database.

-s INPUT specifies the DD name that identifies the input data set that ACIF will index (INPUT in this example).

afp causes the ARSLOAD program to append the processid to the specified file name. This is to allow for running multiple instances of the ARSLOAD program. If you do not specify the -s parameter, the processid will not be appended.

//STEPLIB DD DSN=ARS.V10R1M0.SARSLOAD,DISP=SHR

Specifies the load library for the Content Manager OnDemand programs, which must be customized for the local naming conventions.

//* DD DSN=DB2.V10R1M0.SDSNEXIT

If ODF is being used, uncomment and customize for the local naming conventions.

//* DD DSN=DB2.V10R1M0.SDSNLOAD

If ODF is being used, uncomment and customize for the local naming conventions.

// DD DSN=ACIF.V2R3M1.SAPKMOD1

Specifies the ACIF library, which must be customized for the local naming conventions.

//*DSNAOINI DD PATH='/opt/IBM/ondemand/V10R1M0/config/cli.ini'

If ODF is being used, or ARSLOAD should attempt to not use TCP/IP, uncomment and customize for the local naming conventions.

//INPUT DD ...

Specifies the input data set that ACIF will index. This DD statement is named by the -s parameter.

//CEEDUMP DD ...

Specifies the data set to which Language Environment writes any dumps.

Notes

The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD) must be running, otherwise the ARSLOAD program will fail.

If you need to specify a DDNAME as input to ARSLOAD, use the new -s ddname option to explicitly specify a DD statement (for example, -s INPUT), and change the /dev/null to a non-existent file. For example:

```
// PARM=(' /-u xxxxx -p yyyyyyyy -s INPUT -n -g MYGROUP afpdata')
```

In the above example, afpdata is used to indicate to ARSLOAD that an explicit file is to be loaded. It can be replaced by any non-existent file name. Without this parameter, ARSLOAD attempts to load files from the JES spool and fails. Other parameters must be specified for spool processing to work, and you must have the correct RACF permissions to the SPOOL. In addition, the non-existent file name is used to construct the temporary file names that are created.

Files

/usr/lpp/ars/V10R1M0/bin/arsload

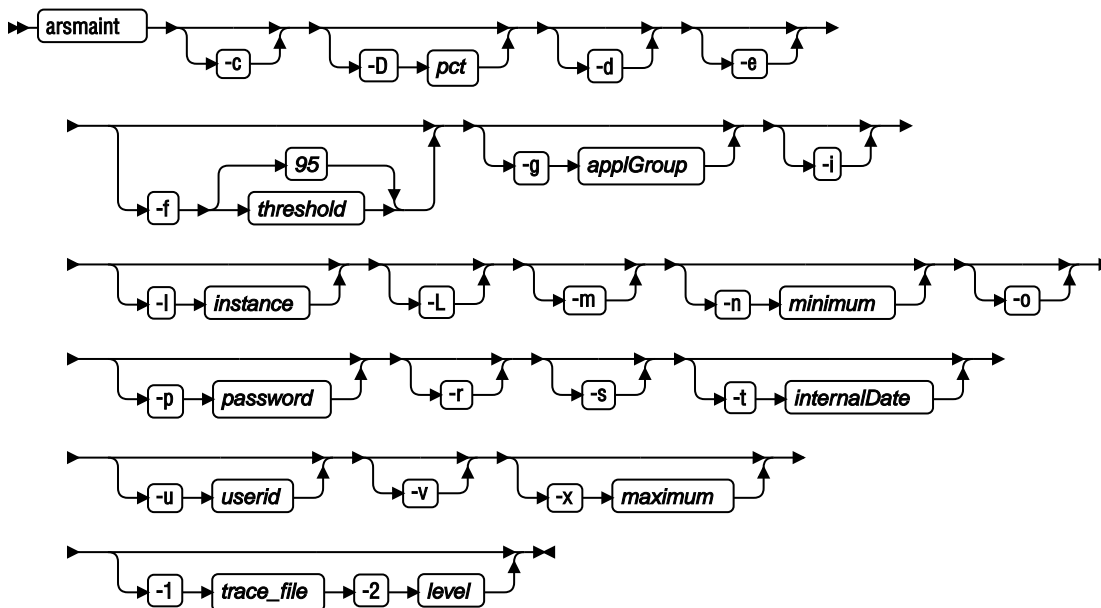
The z/OS executable program.

Chapter 25. ARSMaint

Purpose

Maintain application group data in the Content Manager OnDemand database and in cache storage. You typically run the ARSMaint program on a regular schedule to migrate files from cache storage to archive storage, delete files from cache storage, optionally migrate index data from the database to archive storage, and delete index data from the database.

Syntax



Description

The ARSMaint program maintains application group data that is stored in the Content Manager OnDemand database and in cache storage. The ARSMaint program maintains the system by using the storage management values from the application groups that are defined to the system. The ARSMaint program uses the following storage management values:

- 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
- Length of Time to Maintain Imported Migrated Indexes
- Expiration Type. The ARSMaint program uses the Expiration Type to determine how to delete index data from an application group. The ARSMaint program can expire a table of application group data at a time, an input file at a time, or individual documents.

Use the -d, -e, and -i parameters to migrate and expire index data from the database.

Use the -c and -m parameters to migrate and expire files from cache storage.

Note: Most maintenance processes should run when no other applications are updating the database or need exclusive access to the database and when you are sure that no one will be retrieving documents from the system. You should not maintain the database at the same time that you load data into the system. There may be other processes that conflict; for example, you cannot 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 that you specify will vary, based on your requirements. For example, if you do not load data into the system every day or you add few rows to the database when you do load data, you should be able to maintain the database less often, perhaps once a month. However, if you load a high volume of data every day, you may need to maintain the database after the load processing completes. If your system has sufficient space available to store index and report data, you may be able to maintain the database less often.

The examples in this section illustrate one way to maintain the database and cache storage. You should use the examples as a guideline. The options that you use on your system and the times that you can run the ARSMaint program on your system will be different.

ARSMaint also queries the database for any application groups that are flagged for migration to an external storage manager, and creates migration links in the cache against which ARSMaint is run. The migration date is the current date so when ARSMaint normally processes files for migration, it picks up these newly created links.

Parameters

-c

Expires files from cache storage. Most customers should plan to schedule the ARSMaint program to expire files every day.

-d

Expires indexes from the Content Manager OnDemand database. Most customers should plan to schedule the ARSMaint program to expire indexes every day.

Important: When the application group has an Expiration Type of Load, the -d parameter expires the associated objects at the same time.

-D percentage

Specifies a threshold that, if the actual value is lower than this threshold, you want OnDemand to reload the load. The threshold measures the percentage of documents in a load that have a hold. When OnDemand processes a load, it calculates the percentage of the load that has a hold. If the actual value is lower than this threshold, OnDemand reloads this load. Specify a number between 0 and 100. The default is 0, which means that OnDemand does not reload the load.

For example, assume you have a load with 100 document and you specify -D 50. When OnDemand processes a load, if the percentage of documents with a hold is lower than 50%, OnDemand reloads the load. The following list describes OnDemand behavior as the number of documents with holds increases for this example:

- No documents have a hold. Therefore, the actual percentage is 0%. OnDemand does not reload the load.
- One document has a hold. Therefore, the actual percentage is 1%. OnDemand reloads the load.
- Forty nine document have a hold. Therefore, the actual percentage is 49%. OnDemand reloads the load.
- Fifty documents have a hold. Therefore, the actual percentage is 50%. OnDemand does not reload the load.
- Fifty one documents have a hold. Therefore, the actual percentage is 51%. OnDemand does not reload the load.

-e

Migrates index data from the database to archive storage. Customers that plan to use migration processing should schedule the ARSMaint program to migrate data every day.

Migration is the process of moving tables of index data from the database to archive storage. The migration process allows you to maintain index data for a very long time, on indirect access media. You typically migrate index data after users no longer need to access the documents to which the indexes point, but for legal or other requirements, you still need to maintain the index data for some number of months or years.

Important: Before you can migrate index data, the index tables must be closed. To close a table to loading before the Maximum Rows value is reached, use the ARSTBLSP program with the `-a1` parameter. For more information, see the chapter about ARSTBLSP program.

You must configure application groups for migration. Content Manager OnDemand uses the Length of Time Before Migrating Indexes to determine when to migrate index data. If you need to import index data from archive storage back into the database, then OnDemand will maintain the index data in the database for the number of days specified in the Length of Time to Keep Imported Migrated Indexes. (Content Manager OnDemand maintains migrated index data in archive storage until it reaches its Life of Data and Indexes.)

Before you can migrate index data to archive storage, you must configure the System Migration application group. See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for help with configuring the System Migration application group.

-f threshold

Sends an alert message when a cache storage file system is filled to the specified percentage. The default value is 95 (percent). The range is 0 (zero) to 100.

When a cache storage file system exceeds the threshold, it usually means that the ARSMaint program cannot delete enough documents from the file system to fall below the threshold. You may need to run migration and expiration processing more often, increase the size of your cache storage file systems, or add new cache storage file systems.

The ARSMaint program sends the alert message to the console, the system log, and the system log user exit program.

-g applGroup

The name of the application group to process. Unless you specify this parameter, the ARSMaint program maintains the database and cache storage for all of the application groups defined on the library server. You can specify this parameter and name an application group when you need to maintain the database and cache storage for a specific application group. You can specify multiple application groups by specifying the `-g` parameter multiple times. The `-g` parameter is valid only with the `-c`, `-d`, `-e`, `-i`, `-m`, and `-r` parameters.

-i

Expires imported index data from the database. Customers that need to use migration processing should schedule the ARSMaint program to expire imported indexes every day.

An administrator must import index data that was previously migrated to archive storage back into the database to satisfy a query. After maintaining the imported index data for the number of days specified in the Keep Imported Migrated Indexes field (see the Advanced button on the Storage Management page in the application group), the data is eligible to be removed from the database (and will be removed the next time that the ARSMaint program runs with the `-i` parameter).

-I instance

The name of the Content Manager OnDemand instance to process. The ARSMaint program will attempt to locate the specified instance name in the ARS.INI file, to obtain the

TCP/IP address, host name or host name alias of the system on which the instance is running. If the ARSMaint program cannot locate the instance name in the ARS.INI file, then the specified value is treated as a host name.

You must specify this parameter and name the instance if:

- The name of the default instance is not ARCHIVE.
- You are running more than one instance on the same system and you want to process an instance other than the default instance.

If you are running multiple instances of Content Manager OnDemand on the same workstation, always specify the `-I` parameter to identify the name of the instance that you want to process.

Verify that the ARS.INI file is configured with the correct information for all instances of OnDemand. See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for more information about working with multiple instances of OnDemand.

-L

Queries the ARSLOAD table and updates the `aid` value for all the load IDs that have not been previously found. For multiple applications in an application group, queries the data tables to find a corresponding hit for the load in order to determine the `aid` value.

-m

Migrates files from cache storage to archive storage. IBM recommends that you schedule the ARSMaint program to migrate files every day.

-n minimum

Specifies the low expiration threshold percentage for cache storage file systems. This value determines when the ARSMaint program stops expiring files from cache storage file systems (see the `-c` and `-x` parameters). The range is 0 (zero) to 100.

The default value is 80 (percent), which should keep cache storage file systems fairly full, to provide good retrieval hit rates while leaving adequate free space to load new input files. IBM recommends that you periodically check the amount of space available in your cache storage file systems. Depending on the amount of data that you load into the system and the length of time that you maintain documents in cache storage, you may need to increase the size of the cache storage file systems or add new cache storage file systems.

-o

Expires Content Manager OnDemand Distribution Facility (ODF) table entries. Expired entries in the ARSDFPPT, ARSDFDRT, and ARSDFDST tables will be deleted. By default, entries in these tables older than 30 days are considered expired and are automatically deleted by the ARSODF program. The number of days can be changed by using the `ARSODF_EXPIRE_DAYS` parameter in the `ARS.CFG` configuration file. This means that running ARSMaint `-o` is optional, because the ARSODF program will automatically delete expired entries.

-p password

Specify one of the following options for *password*:

- The name of the stash (encrypted password) file that contains the password for the user ID specified with the `-u` parameter.
- The password for the Content Manager OnDemand user ID specified with the `-u` parameter. If the user is not assigned a password, enter a null password (that is, specify `-p ""`). If you do not specify the `-p` parameter while you specify the `-u` parameter, then the program retrieves the password for the user ID from the stash file specified in ARS.INI file for that instance. If the user is not assigned a password, then press the Enter key when prompted.

-r

Runs database statistics, which causes the database manager to optimize application group index data and make access to information as efficient as possible. To optimize the database, schedule the ARSMaint program after the following activities:

- After a table is loaded.
- After an index is physically created.
- After a table space is reorganized if inline statistics were not collected.
- After you run extensive updates, deletions, or insertions in a tablespace.
- After you run any of the following utilities without collecting inline statistics:
 - RECOVER TABLESPACE
 - REBUILD INDEX
 - REORG INDEX

Before you run REORG with the OFFPOSLIMIT, INDREFLIMIT, or LEAFDISTLIMIT options.

-s

Produces a report that contains status information about the cache storage file systems which is written to the Content Manager OnDemand system log. The report includes information such as the name of the file systems and the amount of free space for each file system. IBM recommends that you schedule the ARSMaint program every day to report the status.

-t date

Specifies that you want the ARSMaint program to process the database and cache storage by using a date other than the current system date (the default value). For example, you can specify a date in the future to cause the ARSMaint program to process data and files before their actual expiration and migration period (for example, if you wanted to clear cache storage). The default format for *date* is YYYY-MM-DD. When you specify this option, you must specify the -u parameter. The ARSMaint program processes all data and files belonging to the application groups for which the user (specified with the -u parameter) has been given permission to delete documents.

Note: Use this option with caution. Contact the IBM support center before you run the ARSMaint program with this option.

-u userid

Specifies a Content Manager OnDemand user that has administrator permission for the application groups to be processed. If you specify the -g parameter, the user must have permission to delete documents from the application groups. If you do not specify this parameter, the ARSMaint program prompts you to enter the userid when you run the program.

If you do not specify the -p parameter with the -u parameter, the ARSADMIN program retrieves the password for the user ID from the ARS.INI file for that instance.

-x maximum

Specifies the high expiration threshold percentage for cache storage file systems. This value determines when the ARSMaint program begins expiring files from cache storage file systems (see the -c parameter). When a cache storage file system reaches this threshold, the ARSMaint program begins deleting data from the file system by using the data migration and caching values from the application groups. The ARSMaint program stops deleting files from a file system when the amount of used space in the file system drops below the low expiration threshold (see the -n parameter). The range is 0 (zero) to 100.

The default value is 80 (percent), which should keep cache storage file systems fairly full, to provide good retrieval hit rates while leaving adequate free space to load new input files. IBM recommends that you periodically check the amount of space available in your cache storage file systems. Depending on the amount of data that you load into the system and

the length of time that you maintain data in cache storage, you may need to increase the size of the cache storage file systems or add new cache storage file systems.

-v

Validates cache storage. When you specify this option, the ARSMaint program inspects all cache storage file systems to make sure that they are correctly linked with the proper file permissions.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

The following ARSMaint examples include guidance on how to add a user ID to the stash file. For example, to add the user ID "sysadmin" to the stash file, enter the following command:

```
arsstash -a 1 -s stash_file -u sysadmin
```

where:

-a 1

Adds the user ID to the stash file.

-s stash_file

Name of the stash file.

-u sysadmin

User ID that is added to the stash file (if -u sysadmin is used on command line, then -p is not required since the password is obtained from the stash file).

1. The following shows how to run the ARSMaint program 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 standard library/object server system (with archive storage on the library server).

```
arsmaint -h ARCHIVE -u sysadmin -cdeimrsv
```

2. The following shows how to run the ARSMaint program 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 cache storage and archive storage on an object server that resides on a different workstation or node than the library server.

```
arsmaint -h ARCHIVE -u sysadmin -cmsv
```

Notes

See the online help that is provided with the administrative client for assistance with configuring the storage management information in your application groups.

The Content Manager OnDemand server programs can remain active while you run the ARSMaint program. However, IBM recommends that you schedule the ARSMaint program to run at a time when little or no other Content Manager OnDemand activity takes place on the system.

It is recommended that you create a backup image of the Content Manager OnDemand database before you run the ARSMaint program.

The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD) must be running, otherwise the ARSMaint program will fail.

Before you can migrate index data, the index tables must be closed. To close a table to loading before the Maximum Rows value is reached, use the ARSTBLSP program with the -a1 parameter. For more information, see [Chapter 30, “ARSTBLSP,” on page 219](#).

If the Content Manager OnDemand cache file system becomes unavailable (as during a system failure), automatic retrieval from the archive storage might be possible, depending on the outage cause.

For example, assume that the HBA Adapter fails and causes the file systems to be unmounted. In this instance, the system looks in the cache first. If the file system is unmounted, the data will not be there, OnDemand then looks in the archive storage (if the data was stored there). There are several other factors that might affect this function:

- The file system is damaged, and commands such as `as1s` will not work.
- OnDemand does not look in the archive storage if there is an error reading the cache. The permissions of the unmounted file system are incorrect. AFP data has resources that are put back into the cache if they are ever removed.
- The unmounted file system might not have enough storage space, so retrievals fail.

For the situations listed above, update the application group and turn off the search cache option. The search cache is located on the Storage Management tab of the application group properties. This causes OnDemand to retrieve directly from the archive storage.

Files

`/usr/lpp/ars/V10R1M0/bin/arsmaint`

The z/OS executable program.

Chapter 26. ARSRPT

Purpose

The reporting command produces a report for a selectable 24-hour period. The reporting command can produce the following optional files:

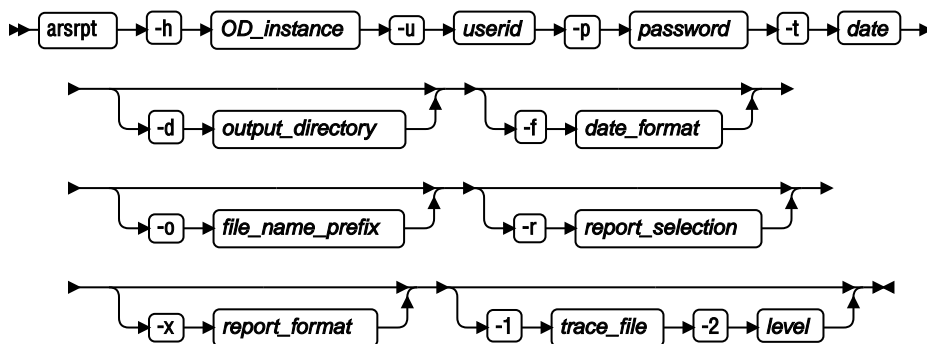
- An XML file with raw data extracted from the Content Manager OnDemand System Log.
- A file summarizing the raw data.

The report provides information about the following sets of operations: log ins, loads, and queries with any resulting retrieves. The report shows the following information about each set:

- user or host
- operational counts
- hourly time periods of when the operations occurred

You can use the summary data to help plan system activities and monitor your system for peak usage and irregularities.

Syntax



Description

The command requires a Content Manager OnDemand instance name, user ID, password, and the date to extract data. The command generates two files: `arsrpt.date.out` and `arsrpt.date.xml` where `date` is derived from the input date given with the `-t` parameter. The remaining parameters are optional. You can use the `-t` and `-x` parameters to create reports focused on a particular activity or output format.

Parameters

-d output_directory

Specifies the directory where you want the report utility to save the output files.

-f date_format

Specifies the format of the date string that you provide with the `-t` parameter. The ARSRPT command assumes that you specify the date in `yyyy-mm-dd` format. You can specify a different format by doing one of the following options:

- Select a format from the following table and specify the corresponding date format specifier.
- Specify a format that is not listed in the table by using the values (for example, %m) and separators (such as the slash (/), dash (-), colon (:), or dot (.) characters). A specification for a date format that uses any of these three chars (slash, dash, or dot) can be used interchangeably; the program allows any of the three, regardless of which one you specified in the date format. A specification for a time format that uses either of these two chars (colon or dot) can be used interchangeably; the program allows either of the two, regardless of which one you specified in the time format.

Table 13: Standard date formats supported by Content Manager OnDemand.

Date format specifier	Date format	Example
%m/%d/%y	mm/dd/yy	01/31/17
%d/%m/%y	dd/mm/yy	31/01/17
%f/%e/%y	m/d/yy	1/31/17
%e/%f/%y	d/m/yy	31/1/17
%y.%m.%d	yy.mm.dd	17.01.31
%m.%d.%Y	mm.dd.yyyy	01.31.2017
%m-%d-%y	mm-dd-yy	01-31-17
%d-%m-%y	dd-mm-yy	31-01-17
%m%d%y	mmddy	013117
%m%d%Y	mmddyyyy	01312017
%y%m%d	yymmdd	170131
%Y%m%d	yyyymmdd	20170131
%b %e, %Y	Mth d, yyyy	Jan 31, 2017
%B %e, %Y	Month d, yyyy	January 31, 2017

-h *OD_instance*

Specifies the name of the Content Manager OnDemand instance.

-I

Specifies the name of the Content Manager OnDemand instance.

-o *file_name_prefix*

Specifies the prefix to add to the output file names.

-p *password*

The password that corresponds to the user ID specified with the -u parameter.

-r *report_selection*

Specifies which activities to save in the output files. If you do not specify this parameter, the reporting utility collects information about all activities. The following list describes the values you can specify for *report_selection*:

1

Specifies that the report include all login activity.

2

Specifies that the report include all load activity.

3

Specifies that the report include all query activity and any corresponding retrieve activity.

-t date

Specifies the date for which the reporting utility collects information. The default format is yyyy-m-dd. If you want to specify the date in a different format, specify the -f parameter with the date format specifier that describes the format.

-u userid

Specifies the Content Manager OnDemand user ID.

-x report_format

Specifies which output files to create. If you do not specify this parameter, the reporting utility creates both output files. The following list describes the values you can specify for this parameter:

1

Create the summary.

2

Create the XML file.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

The following ARSRPT examples include guidance on how to add a user ID to the stash file. For example, to add the user ID "sysadmin" to the stash file, enter the following command:

```
arsstash -a 1 -s stash_file -u sysadmin
```

where:

-a 1

Adds the user ID to the stash file.

-s stash_file

Name of the stash file.

-u sysadmin

User ID that is added to the stash file (if -u sysadmin is used on command line, then -p is not required since the password is obtained from the stash file).

1. If you run the following command:

```
arsrpt -u sysadmin -h ARCHIVE -o outfile -t 2017-10-31
```

You will produce these files:

```
outfile.arsrpt.20171031.out  
outfile.arsrpt.20171031.xml
```

2. If you run the following command:

```
arsrpt -u sysadmin -h ARCHIVE -o outfile -t 10.31.2017 -f %m.%d.%Y
```

You will produce these files:

```
outfile.arsrpt.20171031.out  
outfile.arsrpt.20171031.xml
```

3. If you run the following command:

```
arsrpt -u sysadmin -h ARCHIVE -o outfile -t 2017-10-31 -x 1
```

You will produce this file:

```
outfile.arsrpt.20171031.out
```

4. If you run the following command:

```
arsrpt -u sysadmin -h ARCHIVE -o outfile -t 2017-10-31 -x 2
```

You will produce this file:

```
outfile.arsrpt.20171031.xml
```

5. For a detailed example of how to generate a monthly report showing loads by data type, see <http://www.ibm.com/support/docview.wss?rs=129&uid=swg27027027>.

Notes

For more information, see <http://www.ibm.com/support/docview.wss?rs=129&uid=swg27027027>.

Files

/usr/lpp/ars/V10R1M0/bin/arsrpt

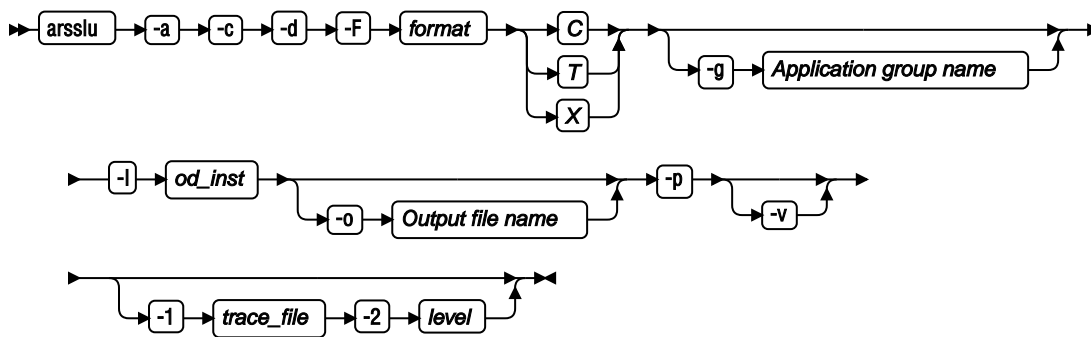
The z/OS executable program.

Chapter 27. ARSSLU

Purpose

The ARSSLU command helps you audit your Content Manager OnDemand database for inconsistencies. Run this command only on the library server.

Syntax



Parameters

-a Audit the Content Manager OnDemand database

This option checks the Content Manager OnDemand database for inconsistencies or potential problems.

-c Create message table and index

Creates a table and index for load data information which is extracted from the Content Manager OnDemand system log. This data is used for auditing purposes.

-d Drop message table and index

Deletes the table and index that are created with the `-c` parameter and any data contained. Do not combine this parameter with any other parameter except for `-v` and `-I`.

-F <format> Trace output information

Optional parameter when `-T` is specified. The default output is text format. The possible values are C, T, and X:

C or CSV

The CSV format is a comma separated file that is useful in spreadsheets.

T or TEXT

Default output format.

X or XML

The XML format outputs XML structured data. Using the XML output requires a header and trailer to be added to the trace file to complete the XML syntax and create valid XML. After the XML structured data is combined with .XSL and .DTD files, it can be viewed in a browser or XML editor. You can obtain these needed files from: <ftp://service.software.ibm.com/software/ondemand/utils/trace>

-g <Application group name>

Specifies an application group name. If the parameter is not specified, the default is all application groups. This parameter is used with the `-a` parameter.

-I <od_inst>

Specifies the database instance name.

-o <Output file name>

This parameter is optional. Any output is sent to the console if it is not specified. This option provides a file name for any log output that is produced by the utility when you use the `-a` parameter. Output to a file is in XML format. Output to the console is in a human readable plain text format.

-p Populate message table

Extracts load information from the System Log and populates the table created with the `-c` parameter. This parameter is used for auditing purposes.

-v Verbose

Prints additional status messages to the console.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

Here are several examples of using the ARSSLU command:

```
arsslu -I ARCHIVE -c -v
Creating table odadmin.arsslu
Creating index odadmin.arsslu.arsapp_idx
Updating runstat statistics for table odadmin.arsslu
```

```
arsslu -I ARCHIVE -p -v
System log data tables: 1
Table name: SL2 Messages present: Yes
Processing table SL2
Total log messages: 333
Log messages processed: 330
Log messages failed: 0
Log messages skipped: 3
Reorganizing table odadmin.arsslu
Updating runstat statistics for table odadmin.arsslu
```

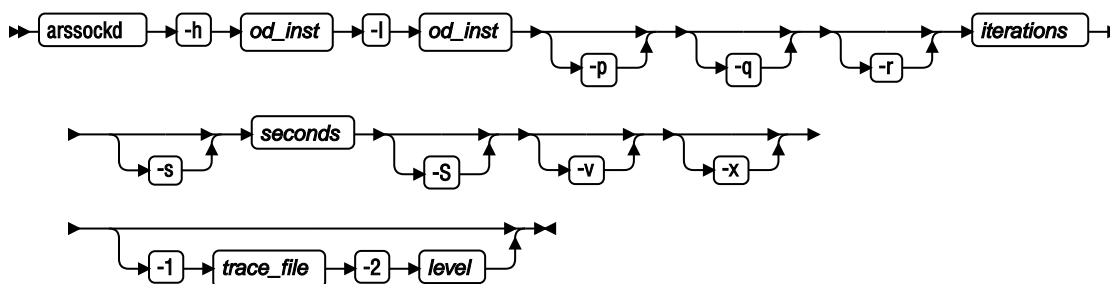
```
arsslu -I ARCHIVE -d -v
Dropping index odadmin.arsslu.arsapp_idx
Dropping table odadmin.arsslu
```

Chapter 28. ARSSOCKD

Purpose

The ARSSOCKD command starts, stops, and displays Content Manager OnDemand process and configuration server information.

Syntax



Description

IBM provides commands to stop the server process on servers and display server usage and configuration information. For example, the syntax to start a server is:

```
arssockd -I instance -S
```

Where *instance* is the name of the Content Manager OnDemand instance to stop. Instance names are listed in the ARS.INI file. The default instance name is *archive*.

Parameters

-h *od_inst*

Content Manager OnDemand instance name or host name and is the same as the -I instance parameter.

-I *od_inst*

Content Manager OnDemand instance name or host name and is the same name as *od_inst* of the -h parameter.

-p

Displays the process usage information for the instance.

-P

Send a ping to the Content Manager OnDemand instance.

-q

Displays the configuration and version information for the instance.

-r *iterations*

Sets the number of iterations (or times) to run the ARSSOCKD command. The default is 1. If you specify a number other than 1, specify the time interval between each iteration with the -s parameter.

-s seconds

Sets the number of seconds between iterations. The default is 1.

-S

Starts the Content Manager OnDemand server for the instance.

-v

Displays the verbose output information.

-x

Displays more information when it is used with the -pparameter.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

Start the server

To start the server, the syntax is: `arsockd -I instance -S`

Process usage information

To display process usage information, the syntax is: `arsockd -I instance -p`

For example:

```
arsockd -I instance -p
```

PID	TID	START TIME	CPU	MEM	STYPE	USERID	INFO
1138878	-	07/04/11 04:09:12	1:25.049844	36724	Program	-	ARCHIVE
1138878	1	07/04/11 04:09:13	0:1.790156	-	Main	-	Accepting
1138878	258	07/04/11 04:09:14	0:0.356935	-	Activity	-	0
1138878	515	07/04/11 04:09:15	0:10.630209	-	DB	-	Idle
1138878	772	07/04/11 04:09:15	0:9.086973	-	DB	-	Idle
1138878	1029	07/04/11 04:09:15	0:15.093788	-	DB	-	Idle
1138878	1286	07/04/11 04:09:15	0:14.202843	-	DB	-	Idle

where

PID

The process ID.

TID

The thread ID.

START TIME

When the process or thread was created.

CPU

The amount of CPU in minutes, seconds, and microseconds (MINS:SECS.MICROSECS) used for the process or thread.

Important: A CPU memory value might not display for every platform. You might also see a 0 value in the command line results.

MEM

The amount of memory used by the process or thread.

USERID

The name of the user currently active on the thread.

INFO

Additional information used to display current activity of the process or thread. For the process, this is the name of the Content Manager OnDemand instance.

Configuration information of the instance

To display configuration information of the instance, the syntax is: `arssockd -I instance -q`

For example:

```
arssockd -I instance -q
```

```
Command: ARSSOCKD
```

```
Version:
```

```
Operating System:
```

```
Operating System Version:
```

```
Database:
```

```
Database Version:
```

```
Storage Manager:
```

```
Storage Manager Version:
```

```
Instance: ARCHIVE
```

```
[@SRV@_ARCHIVE]
```

```
PORT=1445
```

```
SRVR_INSTANCE=ARCHIVE
```

```
SRVR_INSTANCE_OWNER=root
```

```
SRVR_OD_CFG=/opt/IBM/ondemand/V10.1/config/ars.cfg
```

```
SRVR_DB_CFG=/opt/IBM/ondemand/V10.1/config/ars.dbfs
```

```
SRVR_SM_CFG=/opt/IBM/ondemand/V10.1/config/ars.cache
```

```
SSL_PORT=14450
```

```
SSL_KEYRING_FILE=/opt/IBM/ondemand/V10.1/config/ondemand.kdb
```

```
SSL_KEYRING_STASH=/opt/IBM/ondemand/V10.1/config/ondemand.sth
```

```
SSL_KEYRING_LABEL=IBM Content Manager OnDemand
```

```
SSL_CLNT_USE_SSL=0
```

```
PROTOCOL=2
```

```
[CFG]
```

```
ARS_LANGUAGE=ENU
```

```
ARS_ORIGINAL_CODEPAGE=923
```

```
ARS_NUM_DBSRVR=4
```

```
ARS_TMP=/ondemand/arstmp/logs/db2
```

```
ARS_PRINT_PATH=/ondemand/arstmp/logs/db2
```

```
ARS_DB_ENGINE=DB2
```

```
ARS_DB_IMPORT=0
```

```
DB2INSTANCE=archive
```

```
ARS_DB2_DATABASE_PATH=/ondemand/arsdb
```

```
ARS_DB2_PRIMARY_LOGPATH=/ondemand/arsdb_primarylog
```

```
ARS_DB2_LOGFILE_SIZE=1000
```

```
ARS_DB2_LOG_NUMBER=40
```

```
ARS_DB2_TSM_CONFIG=/usr/tivoli/tsm/client/api/bin64/dsm.opt.db2
```

```
ARS_STORAGE_MANAGER=TSM
```

```
ARS_TRACE_SETTINGS=/opt/IBM/ondemand/V10.1/config/archive.trace.settings
```

```
DSMI_DIR=/usr/tivoli/tsm/client/api/bin64
```

```
DSMI_CONFIG=/usr/tivoli/tsm/client/api/bin64/dsm.opt
```

```
DSMI_LOG=/tmp
```

```
[DBFS]
```

```
DBFS1=/ondemand/arsdb/SMS SMS
```

```
DBFS_NUM=1
```

```
[CACHE]
SMFS1=/ondemand/arscache/db2
SMFS_NUM=1
```

Notes

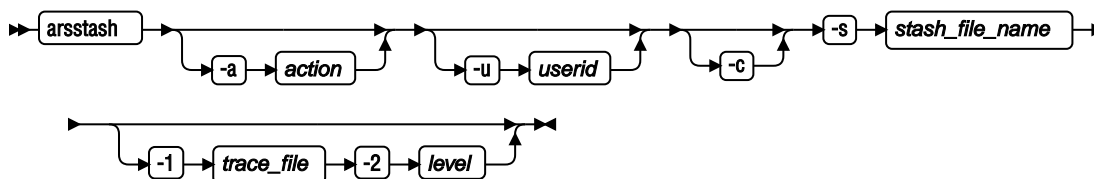
1. When you stop the server process, all users who are connected over the network to the Content Manager OnDemand system are disconnected. Therefore, it is a good idea to warn connected users before stopping the server process.

Chapter 29. ARSSTASH

Purpose

The ARSSTASH command creates and modifies stash files that can be used to store user IDs and passwords.

Syntax



Description

The ARSSTASH command is used to create and modify the stash file. Storing passwords in a stash file can help you improve security because you do not need to specify the password on the command line, where the password might be visible to others. You can store the user ID and password for the following situations in one stash file:

- Each Content Manager OnDemand instance
- Each Content Manager OnDemand program that runs as a daemon or service (for example, ARSLOAD)

You store the stash file in a directory and specify that directory in the SRVR_OD_STASH parameter of the ARS.INI file. Content Manager OnDemand and the Content Manager OnDemand programs locate the stash file in that directory. If you need to override the user ID and password stored in the stash file, create a stash file and store it in a directory where you run a Content Manager OnDemand program. For security reasons, limit access to the file through file permissions or delete it when you no longer need it.

To store the user IDs and passwords into a stash file, complete the following steps:

1. Create a stash file by running the ARSSTASH command. The command prompts you for the password.
2. Save the stash file in a directory and limit access to that file through file permissions.

Parameters

-a *action*

Specifies whether to store or delete a user ID and password from the stash file. If you do not specify the -a parameter, the command runs action 1. If you do not specify the -u parameter with actions 3 - 10, then the ARSSTASH command displays the user ID currently assigned to that instance, daemon, or service. You can specify one of the following values for the *action* parameter:

1

Store the Content Manager OnDemand user ID and password. If you do not specify an action, this action is the default action.

2

Delete the Content Manager OnDemand user ID and password.

3

Store the Content Manager OnDemand user ID and password for the ARSLOAD program.

4

Store the Content Manager OnDemand user ID and password for Content Manager OnDemand Distribution Facility (ODF).

6

Store the Content Manager OnDemand user ID and password for Production Data Distribution.

7

Store the LDAP user ID and password.

8

Store the DB2 user ID and password.

10

Store the ZooKeeper user ID and password.

-c

Create a stash file.

-s *stash_file_name*

Name of the stash file.

-u *userid*

User ID for the Content Manager OnDemand instance, daemon, or service specified by the *action* variable.

-1 *trace_file*

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 *level*

Specify a numeric value as directed by IBM Software Support.

Examples

1. To store the user ID DEMOUSER and password for use by any Content Manager OnDemand instance, daemon, or service, in an existing stash file called `ars.stash`, enter the following command:
`arsstash -a 1 -s ars.stash -u DEMOUSER`
2. To store the Oracle user ID USER1 and password in a stash file called `ars.stash` and create the stash file, enter the following command:
`arsstash -a 9 -c -s ars.stash -u USER1`
3. To store the user ID USER2 and password for the LDAP service in an existing stash file called `ars.stash`, enter the following command:
`arsstash -a 7 -s ars.stash -u USER2`
4. To store the user ID USER3 and password for the ARSLOAD daemon in an existing stash file called `ars.stash`, enter the following command:
`arsstash -a 3 -s ars.stash -u USER3`
5. To view the user ID currently assigned to ARSLOAD, enter the following command:
`arsstash -a 3 -s ars.stash`

Files

`/usr/lpp/ars/V10R1M0/bin/arsstash`

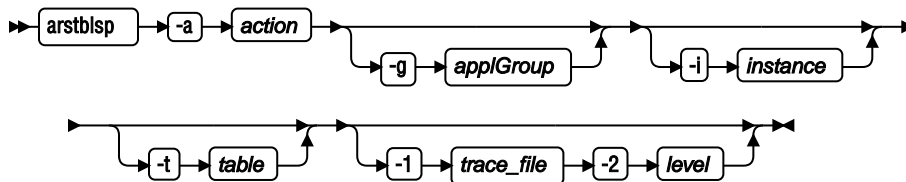
The z/OS executable program.

Chapter 30. ARSTBLSP

Purpose

The ARSTBLSP program can be run to change the table that Content Manager OnDemand loads data into.

Syntax



Description

During normal operation, Content Manager OnDemand loads index rows into a table until the Maximum Rows value for the application group has been reached. Such a table is said to be open for loading. When the Maximum Rows value is reached, the table is closed and a new table and table space are created.

Under certain circumstances, an installation may want to close a table to loading before the Maximum Rows value is reached. For example, migration processing (by using `arsmaint -e`) will not process a table that is open for loading, and the installation may want to migrate the table earlier than initially anticipated.

Parameters

-a action

The action to perform. The action can be one of the following values:

1

Close a table that is still open for loading. This action causes Content Manager OnDemand to close the table that is currently open for loading in the specified application group. The next time that data is loaded into the application group, the data is loaded into a table space. Optionally specify a table to close with the `-t` parameter.

2

List the tables of application group data that have not been migrated to separate table spaces.

3

List the tables of application group data that are open for loading. An open table must be closed before you can migrate it to a table space.

-g applGroup

The name of the application group to process.

-i instance

The name of the Content Manager OnDemand instance to process. The ARSTBLSP program will attempt to locate the specified instance name in the `ARS.INI` file, to obtain the TCP/IP address, host name or host name alias of the system on which the instance is running. If

the ARSTBLSP program cannot locate the instance name in the ARS . INI file, then the specified value is treated as a host name.

You must specify this parameter and name the instance if:

- The name of the default instance is not ARCHIVE.
- You are running more than one instance on the same system and you want to process an instance other than the default instance.
- You are running the ARSTBLSP program on a system other than the system on which the instance that you want to process is running.

Note: If you are running multiple instances of Content Manager OnDemand on the same system, then you should always specify the `-i` parameter to identify the name of the instance that you want to process. You should also verify that the ARS . INI file is configured with the correct information for all instances of Content Manager OnDemand. See the *IBM Content Manager OnDemand for z/OS: Configuration Guide*, SC19 –1212 for more information about working with multiple instances of OnDemand.

-t table

When used with the `-a 1` parameter, identifies the table to close.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

1. The following shows how to list the tables that have not been migrated for a specific application group:

```
arstblsp -a 2 -g credit
```

2. The following shows how to list the tables that are open for loading for a specific application group:

```
arstblsp -a 3 -g credit
```

3. The following shows how to close any open tables for a specific application group:

```
arstblsp -a 1 -g credit
```

4. The following shows how to close a specific table for a specific application group:

```
arstblsp -a 1 -g credit -t AAA26
```

5. The following shows how to close the System Log table and forces the system log to a new tablespace:

```
arstblsp -I instance_name -a 1 -g "System Log"
```

Notes

The ARSTBLSP program can run without the server active. It directly manipulates the Content Manager OnDemand system tables through ODBC. You might need to set the `DSNAOINI` environment variable in the ODBC initialization file. See *DB2 Universal Database™ for OS/390 and z/OS ODBC Guide and Reference*, SC26-9941 for more information.

Important: The ARSTBLSP program can run without an active server. The server must be down to update the System Log table.

Files

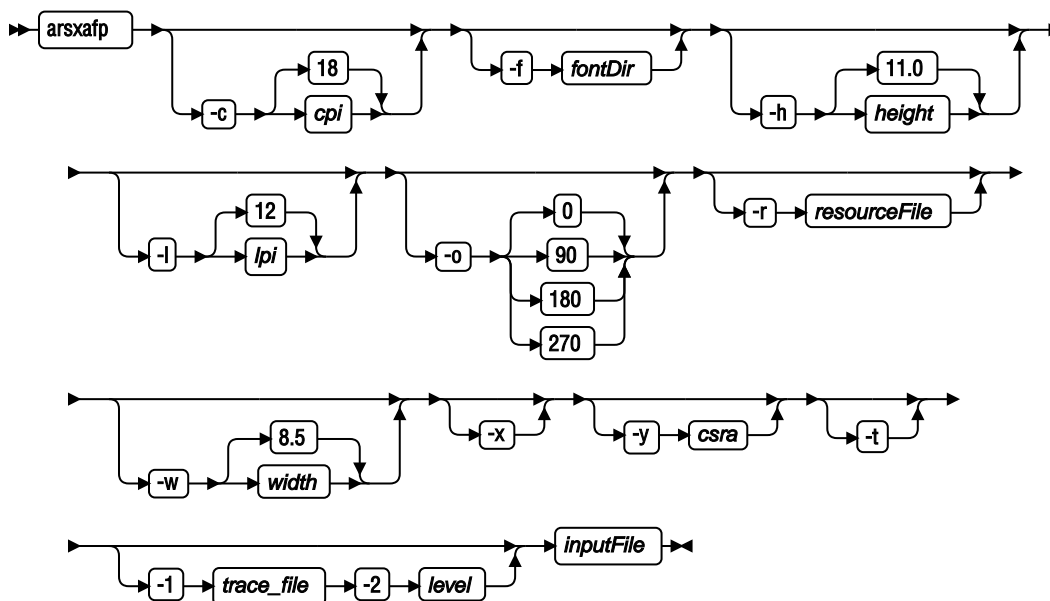
/usr/lpp/ars/V10R1M0/bin/arstblsp
The z/OS executable program.

Chapter 31. ARSXAFP

Purpose

The ARSXAFP program processes an AFP document and creates a line data file suitable for viewing on a 3270 display or other line-mode terminal.

Syntax



Description

You can use the Content Manager OnDemand CICS client program to view documents retrieved from the system on 3270 displays (line-mode terminals). If the document contains AFP data, Content Manager OnDemand calls the ARSXAFP program to process the AFP data and create a line data file. Content Manager OnDemand then sends the line data file to the client program. The ARSXAFP program can also be used as a general purpose AFP to line data batch conversion utility.

You may run ARSXAFP as a shell script, you can invoke it in TSO using the OMVS command; you can also invoke the ARSXAFP program in USS from a user-defined program or from the command line.

You can optionally specify a directory that contains user-defined character sets and code page files that the ARSXAFP program can use to process the input data. See *IBM Content Manager OnDemand: Windows Client Customization Guide* for information about defining character set and code page files for use with Content Manager OnDemand client programs.

The ARSXAFP program ignores any graphics, images, and typographic fonts that may be present in the input data. All fonts in the input data 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 ARSXAFP program 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, then you would specify `-c 12`. The default value is 18 characters per inch.

-f fontDir

The name of the directory that contains user-defined character set and code page files required to process the input file. See *IBM Content Manager OnDemand: Windows Client Customization Guide* for information about defining character set and code page files for use with Content Manager OnDemand client programs.

-h height

The height of a page in the input file. The `-w` and `-h` parameters define the paper size used to print a page of the input file. The default value is 11.0 inches.

-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.

-o orientation

The orientation of the data on the page in the input file. You can specify that the data appears rotated 0, 90, 180, or 270 degrees. The default value is 0 degrees.

-r resourceFile

The name of the resource group file that contains the form definition and overlay associated with the input file. The ARSXAFP program can process information contained in the form definition, such as the offset, and include text from an overlay in the output file.

-t file

AFP viewer trace file.

-w width

The width of a page in the input file. The `-w` and `-h` parameters define the paper size used to print a page of the input file. The default value is 8.5 inches.

-x

Use to specify that you do not want the output to contain any horizontal or vertical lines that were present in the AFP document.

-y

Use to specify the value that represents the character space reset algorithm with which to place text in the output.

inputFile

The name of the AFP input file to process

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Examples

1. The following shows how to process an AFP input file located in the current directory and redirect the line data output to a file. By default, the input file contains 8.5 x 11 inch pages:

```
arsxafp -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:

```
arsxafp 96089900.AFP
```

Files

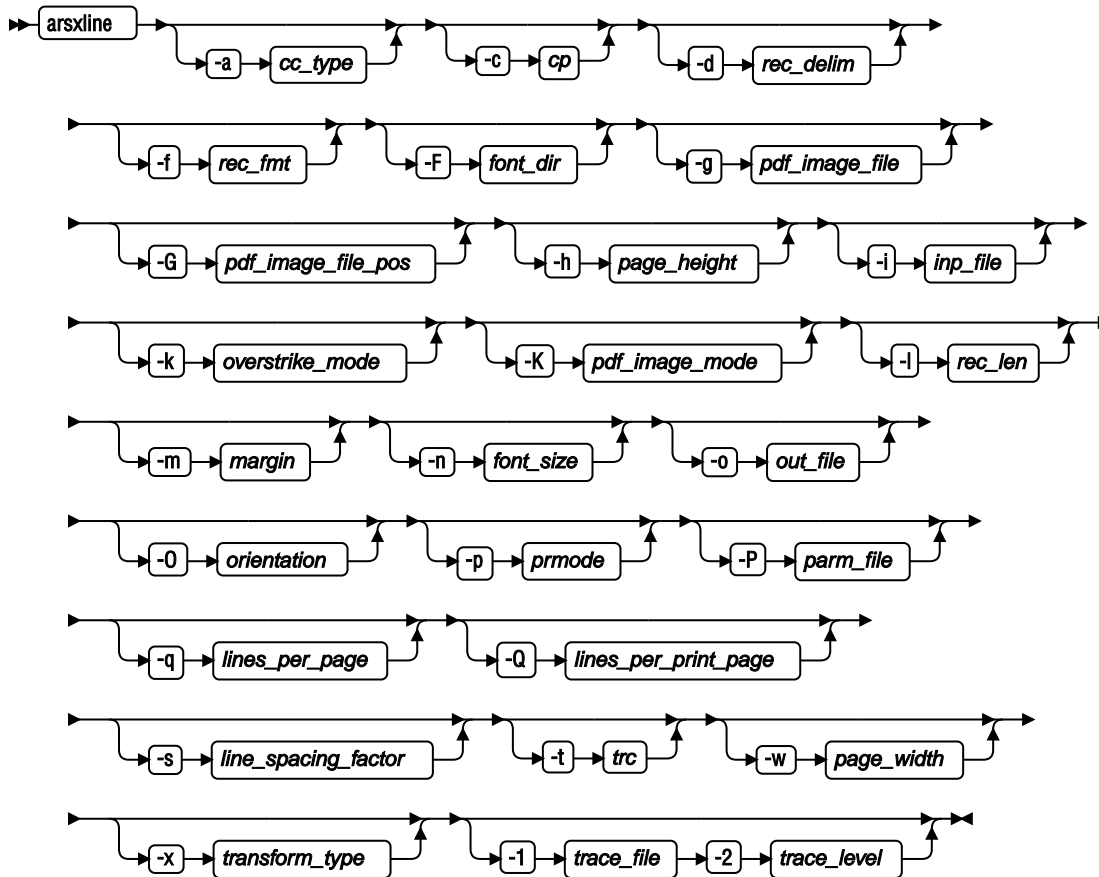
/usr/lpp/ars/V10R1M0/bin/arsxafp
The z/OS executable program.

Chapter 32. ARSXLIN

Purpose

The ARSXLIN program converts line data to UTF-8 or PDF.

Syntax



Description

Converts line data to UTF-8 or PDF.

You specify the format of the input file with the parameters.

Parameters

-a cc_type

Specify one of the following values to indicate the type of carriage control used in the input file:

A

ANSI. This is the default value.

M
Machine

N
None

-c cp

Specify the code page used by the input file. The default value is 500.

-d rec_delim

Specify the value, in hexadecimal, used in the input file to indicate the end of record. The default value is 0A.

-f rec_fmt

Specify one of the following values to indicate the record format used in the input file:

F
Fixed. This is the default value.

V
Variable.

S
Stream. If Stream is specified, the default carriage control is None.

-F font_dir

Specify the path or paths of the font files required by the input file.

-g pdf_image_file

Specify the path and name of the file containing a PDF image. The file name must end with .png or .jpg extension.

-G pdf_image_file_pos

Specify the location on the page to place the top left corner of the image file. The upper left corner of the page is 0,0 which is the default.

-h page_height

Specify the height of the page in the output file.

-i inp_file

Specify the name of the input file.

-k overstrike_mode

Specify one of the following values to indicate how the overstrike carriage control is processed:

M
Merge. This is the default value.

S
Separate.

F
Full.

-K pdf_image_mode

Specify one of the following values to indicate the PDF image mode:

A
Place the image on all of the pages. This is the default value.

F
Place the image on the first page only.

L
Place the image on the last page only.

-l rec_len

If the record format is fixed, specify the length of the record. The default value is 133.

-m margin

Specify the size of the margin in the following format:

size_of_top_margin, size_of_bottom_margin, size_of_left_margin, size_of_right_margin
top, bottom, left, right

-n font_size

Specify the size of the font used in the output file. The default value is 12.

-o out_file

Specify the name of the output file.

-O orientation

Specify one of the following values to indicate the page orientation:

P

Portrait.

L

Landscape.

-p prmode

Specify one of the following values to indicate the presentation mode used by the input file:

N

None. This is the default value.

1

SOSI1

2

SOSI2

3

SOSI3

-P parm_file

Specify the name of the parameter file.

-q lines_per_page

Specify the number of lines per page. The default value is 0.

-Q lines_per_print_page

Specify the number of lines per page when the document is printed. The default value is 0.

-s line_spacing_factor

Specify the size of the space between lines. The default value is 1.16.

-t trc

Specify one of the following values to indicate if the file contains TRC:

0

No. This is the default.

1

Yes.

-w page_width

Specify the width of the page in the output file.

-x transform_type

Specify one of the following values to indicate the target format:

U

UTF-8. This is the default.

P
PDF.

-1 *trace_file*

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 *trace_level*

Specify a numeric value as directed by IBM Software Support.

Note: The -g, -G, -h, -K, -m, -n, -O, -s, and -w parameters are only used when the document is converted to PDF.

Examples

To convert an EBCDIC fixed length (with record length 200) file with ANSI carriage controls to UTF-8, enter the following command: `arsxline -l 200 -i input_file -o output_file`

Chapter 33. ARSXML

Purpose

The ARSXML program imports objects within an existing IBM Content Manager OnDemand XML file into a Content Manager OnDemand system, and exports existing Content Manager OnDemand objects into a Content Manager OnDemand XML file.

Syntax

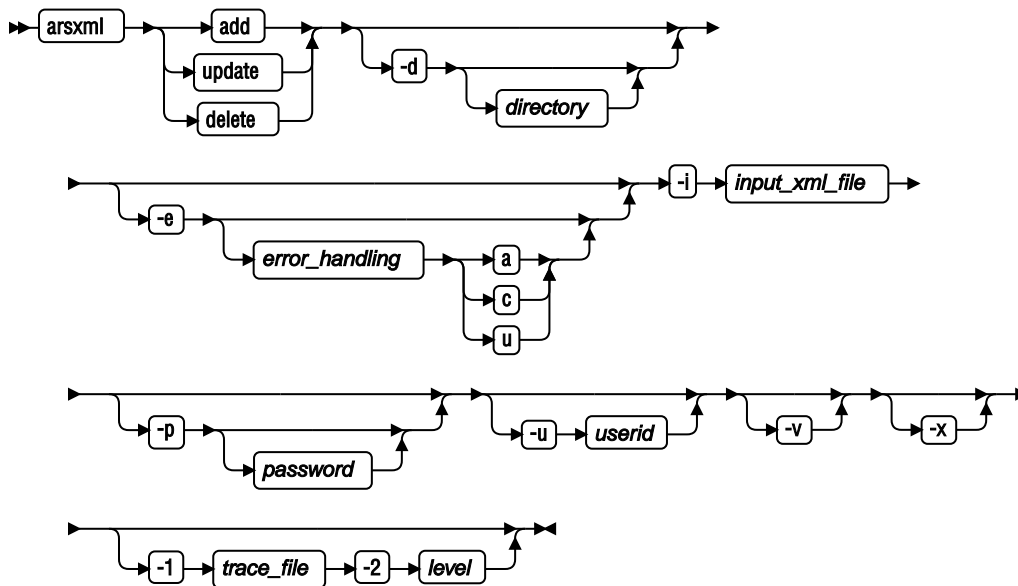


Figure 16: Syntax for adding, updating, and deleting administrative objects in a Content Manager OnDemand system

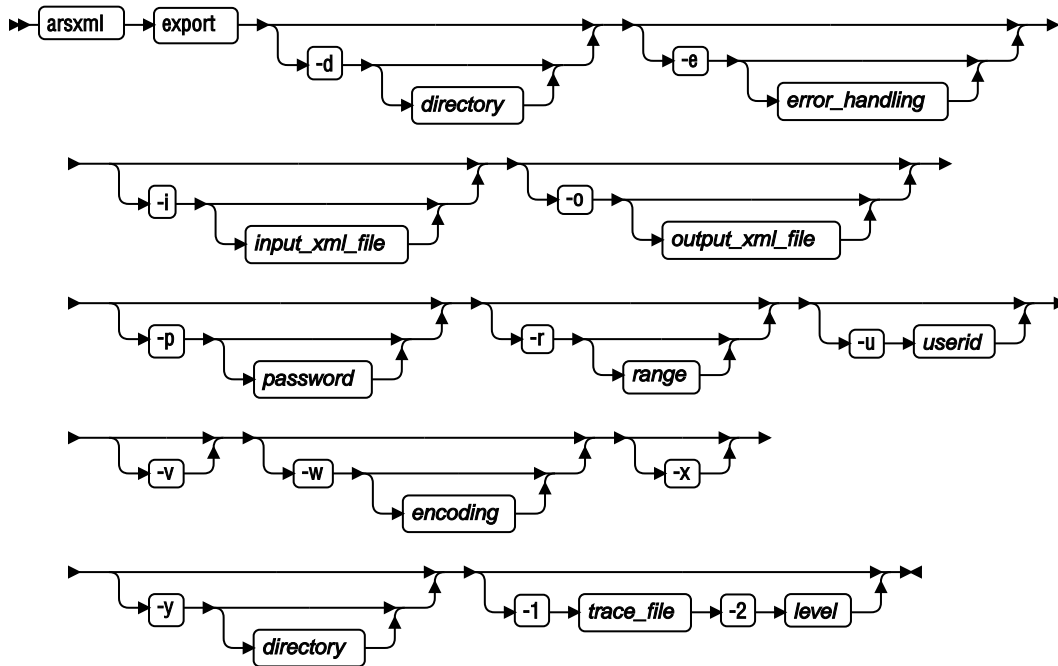


Figure 17: Syntax for exporting existing Content Manager OnDemand objects into a Content Manager OnDemand XML file

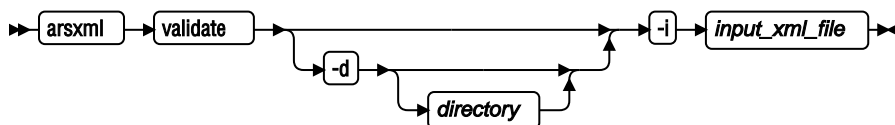


Figure 18: Syntax for validating a Content Manager OnDemand XML file

Description

IBM Content Manager OnDemand includes an XML interface that imports and exports administrative objects in Content Manager OnDemand. In this model, all administrative objects are exported into an XML file, and can be imported into the same system or another system later. You can also create an XML file through a user application or Web interface according to the defined specifications, and import a single object or multiple objects into the system by importing the XML file.

The data import feature allows you to import a single object, a set of defined objects, or even an entire OnDemand system. This features enables you to complete the following tasks:

- Update a system with newly defined objects
- Backup a complete system
- Copy a set of objects from one system to another system

The XML import file can be one of the following:

- A file which was previously created during a Content Manager OnDemand export process.
- A user created XML file which conforms to the Content Manager OnDemand XML schema.

The following table shows the ARSXML return codes and their meanings.

Return code	Meaning
0	success

Table 14: ARSXML return codes (continued)

Return code	Meaning
1	warning
2	syntax error
3	error
4	unrecoverable error

Important:

- The syntax of each Content Manager OnDemand object that is supported by the import process is specified in the Content Manager OnDemand XML schema file. Each object within the XML file must conform to the exact syntax as specified in the Content Manager OnDemand XML schema file that is shipped with the product. Objects must appear in the order described in the following document: <http://www.ibm.com/support/docview.wss?uid=swg27019614>.
- The import XML file references an `ondemand.xsd` file, which is provided with Content Manager OnDemand. The `ondemand.xsd` file must reside in the same directory where the input XML file is located, otherwise, the reference in the input XML file must contain a full path name for the `ondemand.xsd` file.
- The capitalization of the object names is important and should be used exactly as shown in the data tables in .
- The ARSXML validate function validates only the input XML file schema. It is possible that your input XML file passes the validation but still fails due to object inconsistencies. For example, if you attempt to update a user that does not exist, validation can succeed but you will receive an error message in the update process.

Parameters for ARSXML add, update, delete

add

This is the default action. It adds all of the objects in the input XML file into the specified IBM Content Manager OnDemand system. For objects that contain child objects, such as a list of users or a set of permissions, each child object that is found in the XML file is added to the current list of objects.

update

All the fields that are specified in the input XML file are updated for the object. For example, if the XML file contains a user with a name of `harry` and a description of `The New Description` specified, then only the description field of the user `harry` is updated. All other fields remain unchanged.

Restriction: Not all fields of all objects can be updated.

For objects which contain child objects, such as a list of users or a set of permissions, each child object that is found in the XML file are added to the list of objects. A special attribute on each child object can be set to `delete` for the cases where a member of a list need to be removed. For example, if you want to add the user `newUser` to a group, and delete the user `oldUser`, the following XML code can be used during an update:

```
...
<group name="MyGroup">
  <user name="newUser" />
  <user name="oldUser" task="delete" />
</group>
...
```

delete

All of the objects that are specified in the input XML file are deleted from the system. The only attribute that is examined in the XML file is the name attribute. While all other fields might be present in the XML file, they are ignored.

-d

Specifies the directory that contains the XML file. If you do not use this parameter, ARSXML looks for the input XML file specified by the `-i` flag in the current working directory where the command is run.

-e error_handling

Controls how the import process handles any Content Manager OnDemand errors.

Important: The XML file syntax errors and other XML-specific errors are detected by the parsing code, and no objects are processed.

The `error_handling` parameter can take three values:

a

Abort. This is the default value.

c

Continue. If an error occurs because of problems in the objects or the Content Manager OnDemand system (for example, you attempt to add an object but that object already exists, or you attempt to delete an object that does not exist), the object containing the error is skipped, and the process continues. However, if an XML parsing error occurs, ARSXML stops regardless of whether or not this option is specified.

u

Update. This value should only be used during an add action.

During an add action:

Abort

If an error occurs during an add (for example, an invalid parameter, or the object already exists), the error is logged, the entire import process is stopped, and no further objects are added.

Continue

If an error occurs during an add action, the object containing the error is skipped and an error message is logged. However, the import process continues.

Update

If an object already exists, perform an update action instead of an add action.

During a delete action:

Abort

If an error occurs during a delete (for example, there is an invalid object or the object does not exist), the error is logged, the entire delete process is stopped, and no further objects are deleted.

Continue

If an error occurs during a delete action, the object containing the error is skipped and an error message is logged. However, the import process continues.

During an update action:

Abort

If an error occurs during a update (for example, there is an invalid object or the object does not exist), the error is logged, the entire update process is stopped, and no further objects are updated.

Continue

If an error occurs during an update action, the object containing the error is skipped and an error message is logged. However, the import process continues.

-i

Specifies the input XML file.

-p password

The password is optional. If you omit the -p parameter but you do specify a user ID with the -u parameter, the program prompts you to enter the password. If you omit both the -u and the -p parameters, the system attempts a unified logon. If the unified logon fails, you are prompted to enter a user ID and password.

-u userid

The user ID is optional. If you omit this parameter, the ARSXML program prompts you for a password. If you omit both the -p and -u parameters, the program attempts a unified logon. If the unified logon fails, you are prompted to enter a user ID and password.

-v

Enables verbose mode, which displays all messages (informational and error). By default, the ARSXML program displays error messages.

-x

Used to prevent prompting from occurring. If you do not use this option, during a delete operation, you are prompted whether you really want to complete the operation, for example:

```
The printer object named 'LabPrinter' is about to be deleted.  
Do you want to delete this object? (Y/N)
```

You need to respond with a y or Y to confirm the delete operation. Any other response terminates the operation. If you use this option, you do not get the prompt during an operation.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Important: While importing objects, if an object references another object in its definition but the referenced object cannot be found, and the import command uses the -e c parameter, an error message is issued, and the default value is used. For example, if a user is defined with a default printer and the default printer cannot be found, the user is added with a default printer of *NONE. If the -e c parameter is not used, the above action will not occur.

Parameters for ARSXML export

-d

Use this option to specify the directory for the input XML file. If you do not use this parameter, the ARSXML command looks for the input XML file that is specified by the -i flag in the current working directory where the command is run.

-e error_handling

Specifies what to do if an error occurs during the export process. The error_handling parameter can take two values:

a

Abort. This is the default value. The export process stops when an error occurs.

c

Continue. If an error occurs due to problems in the objects or the IBM Content Manager OnDemand system (for example, you attempt to export an object but that object already exists), the object that contains the error is skipped, and the process continues.

However, if an XML parsing error occurs, ARSXML stops regardless of whether this option is specified.

-i

Use to specify the input XML file. All of the objects that are specified in the XML file are exported from the system into the output XML file. In this case, the only attribute in the XML file that is examined is the name attribute. While all other attributes are present in the XML file, they are ignored. If an object is specified with the name `_ALL`, all of the objects of that type defined on the system are exported. If the `-i` parameter is not present, all of the objects in the specified system are exported. However, you are prompted to confirm whether this is what is intended, because the export process could take a long time. You can use the `-x` option to turn off the prompting.

-o

Used to specify the name of the output XML file. If this parameter is not specified, the output is directed to `STDOUT`.

-r range

Use this option to specify how much data to export. The range parameter can take four values:

a

When an application group is exported, all of the applications that are contained within the application group are exported.

d

Export the specified objects and all dependent objects. If you use this parameter, each object is examined for any dependent objects, and those objects are exported as well. For example, if only user groups are exported and the `d` option is used for the extent, then any users that the groups refer to are exported as well.

l

Export the objects and any logical views that are associated with them.

p

Export the objects and a list of permissions for each object.

More than one range option can be specified in the command line, in other words, the options may be combined. For example,

```
-r pl
```

If you do not specify the `-r p` option, then Content Manager OnDemand exports only the specified object or objects and `*PUBLIC` permission nodes.

-v

Enables verbose mode, which displays all messages (informational and error). By default, the ARSXML program displays error messages.

-w

Specifies the encoding in which the resultant output XML file will be created. The default is UTF-8. The list of supported code pages can be found in the `arscpcs.cfg` file located under the locale directory for the server. For example: `/opt/IBM/ondemand/V10.1/locale/arscpcs.cfg`

-x

Used to prevent prompting from occurring. If you do not use this option, during an export operation, you are prompted whether you really want to complete the operation, for example:

```
No input file was specified.  
Do you want to export all of the objects on the system? (Y/N)
```

You need to respond with a y or Y to confirm the process. Any other response terminates the process. If you use this option, you do not get the prompt during an operation.

-y

Use this option to specify the directory for the output XML file. If this option is not used, the output file that is specified by the -o parameter is written to the current working directory where the command is run.

-1 trace_file

Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level

Specify a numeric value as directed by IBM Software Support.

Parameters for ARSXML validate

-i file

Use to specify the input XML file.

-d directory

Use to specify an input directory where the -i file can be found

Examples

Example 1: Adding users

A user wants to add several users to an IBM Content Manager OnDemand system. That user has created an XML file called `newusers.xml`.

To complete this task, the following command line function is called:

```
arsxml add -h neptune.ny.ibm.com -u admin -i newusers.xml
```

The file `newusers.xml` might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <user name="Bill" password="xxxxx" userType="User Admin"/>
  <user name="Erin" password="xxxxx" userType="User"/>
  <user name="Brie" password="xxxxx" userType="User"/>
</onDemand>
```

Example 2: Updating users

A user wants to update the telephone numbers of the users in the system. This user created an XML file `phone.xml`, which contains the data for the users and their phone numbers.

To complete this task, the following command line function is called:

```
arsxml update -h neptune.ny.ibm.com -u admin -i phone.xml
```

The file `phone.xml` might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <user name="Bill" phone="(212) 555-0919"/>
  <user name="Erin" phone="(212) 555-4295"/>
</onDemand>
```

```
<user name="Brie" phone="(212) 555-0301"/>
</onDemand>
```

Example 3: Deleting users

A user wants to delete another user that no longer needs access to a printer on an IBM Content Manager OnDemand system. That user has created an XML file called `userExpDel.xml`.

To complete this task, the following command line function is called:

```
arsxml delete -h neptune.ny.ibm.com -u admin -p password -v-x-e c i
userExpDel.xml
```

The file `userExpDel.xml` might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <user name="Bill" password="xxxxx"/>
</onDemand>
```

Example 4: Exporting multiple objects

A user wants to export the printer objects named `Prz1` and `Prz2`. This user also want to export the storage set named `FavoriteSS`. An XML file `exportlist.xml` has been created, and contains the information on these objects.

To complete this task, the following command line function is called:

```
>arsxml export -h jupiter.ibm.com -i exportlist.xml -o output.xml
```

The file `exportlist.xml` might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <printer name="Prz1"/>
  <printer name="Prz2"/>
  <storageSet name="FavoriteSS"/>
</onDemand>
```

After the user runs the command, an output file named `output.xml` is created, and contains the information for the two printers and the storage set.

Example 5: Adding groups

A user wants to add a group of users to an IBM Content Manager OnDemand system. That user has created an XML file called `groupAdd.xml`.

To complete this task, the following command line function is called:

```
arsxml add -h neptune.ny.ibm.com -u admin -i groupAdd.xml
```

The file `groupAdd.xml` might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <group name="BigGroup" password="xxxxx" groupType="Group Admin"/>
</onDemand>
```

Example 6: Exporting a group

A user wants to export a group named BigGroup and any users and user groups that are referenced by it. A file named exportbiggroup.xml has been created and looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <group name="BigGroup"/>
</onDemand>
```

To complete the task, the following command should be used:

```
>arsxml export -h jupiter.ibm.com -i exportbiggroup.xml -r d -o output.xml
```

This command creates an XML file that contains the BigGroup object and all the users and groups that are referenced by it. Also, any users within the referenced groups are exported.

Example 7: Updating groups

A user wants to update the name of an existing group. This user created an XML file groupUpdate.xml, which contains the data for the users and their user names.

To complete this task, the following command line function is called:

```
arsxml update -h neptune.ny.ibm.com -u admin -i groupUpdate.xml
```

The file groupUpdate.xml might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <group name="BigGroup" name="SmallGroup"/>
</onDemand>
```

Example 8: Exporting all of the users

A system administrator wants to export all of the users in an IBM Content Manager OnDemand system. A file named exportallusers.xml is created, and looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <user name="_ALL"/>
</onDemand>
```

To complete the task, the following command should be used:

```
arsxml export -h jupiter.ibm.com -i exportallusers.xml -o users.xml
```

This command creates an XML file users.xml that contains the information on all of the users that are defined on the specified system.

Example 9: Validating input XML file schema

A user created an XML file called newusers.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <user name="Bill" password="xxxxx" userType="User Admin"/>
</onDemand>
```

```
<user name="Erin" password="xxxxx" userType="User"/>
<user name="Brie" password="xxxxx" userType="User"/>
</onDemand>
```

The user then uses this command to validate the `newusers.xml`:

```
arsxml validate -i newusers.xml
```

Example 10: Adding printers

A user wants to add new printers to an IBM Content Manager OnDemand system. That user has created an XML file called `printerAdd.xml`.

To complete this task, the following command line function is called:

```
arsxml add -h neptune.ny.ibm.com -u -p password -v -i printerAdd.xml
```

The file `printerAdd.xml` might look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xsi:noNamespaceSchemaLocation="ondemand.xsd">
  <printer name="Printer1"/>
  <printer name="Printer2"/>
  <printer name="Printer3"/>
</onDemand>
```

Chapter 34. ARSYSPIN

Introduction

The JES Spool Data Capture facility provides a means to collect and consolidate JES Spool (SYSOUT) data sets into one (or more) files so they can be archived by Content Manager OnDemand.

The facility executes as a started task in its own address space. Throughout the remainder of this document, the facility is referred to by its program name, ARSYSPIN.

A control statement file is used to provide ARSYSPIN parameters. These parameters specify JES Spool file selection criteria (for example, from which SYSOUT classes output is to be selected) as well as other operational characteristics.

ARSYSPIN creates an intermediate output file which contains one or more Spool file(s) from one or more job(s). Each captured spool file is bracketed by a pair of separator records (see [“ARSYSPIN intermediate file structure”](#) on page 266 for the structure of these records). A "Begin" separator record is written before the first record of the spool file and an "End" separator record is written after the last record of the spool file data. The separator records contain information that can be used to construct index values to facilitate the retrieval and viewing of the captured data sets after they have been stored by OnDemand.

The intermediate output file is indexed and stored in Content Manager OnDemand using the ARSLOAD program. ARSYSPIN invokes ARSLOAD when sufficient data has been captured in the intermediate output file. ARSLOAD indexes the data by calling an indexer program (usually APKACIF) to extract the index values from the data and store them in an index file. ARSLOAD then adds the indexes to the database and loads the data onto archive media.

A sample APKACIF input exit is provided with ARSYSPIN which illustrates a technique for inserting a data record into the APKACIF input stream for the purpose of providing additional indexing information. This sample exit scans the JES Job Log and System Messages spool files (if they are present) to locate critical system and job processing messages (for example, messages which indicate the execution of job steps which terminated via ABEND or with non-zero completion codes), extract related values and place this information in the inserted record for subsequent indexing. See *IBM Content Manager OnDemand: Messages and Codes*, SC27-1379 for a list of ABEND codes and reason codes that are associated with ARSYSPIN.

This sample APKACIF input exit is not an element of ARSYSPIN. It is provided solely as an example of a method for inserting additional records into the data stream. ARSYSPIN has its own exit facility, which might be used to insert records into the data stream. It is recommended that you use the ARSYSPIN exits instead of indexer program exits when it is practical.

After the captured Spool data has been stored into Content Manager OnDemand, ARSYSPIN repeats the process of collecting and consolidating other spool files so they too can be archived in Content Manager OnDemand. The cycle of collecting, consolidating and loading data continues until the program is terminated via the MVS STOP command.

JCL requirements

The following figure illustrates the JCL required for the operation of ARSYSPIN.

```

1 //IEFPROC EXEC PGM=ARSYSPIN,REGION=0M
2 //STEPLIB DD DISP=SHR,DSN=ARSV841.SARSLOAD
           // DD DISP=SHR,DSN=CEE.SCEERUN
           // DD DISP=SHR,DSN=CBC.SCBCCMP
           // DD DISP=SHR,DSN=DB2.V10R1M0.SDSNEXIT
           // DD DISP=SHR,DSN=DB2.V10R1M0.SDSNLOAD
           // DD DISP=SHR,DSN=PSF.SAPKMOD1
3 //SYSIN DD DUMMY,DCB=(LRECL=80,BLKSIZE=80,RECFM=FB)
4 //SYSPRINT DD SYSOUT=*
5 //SYSOUT DD SYSOUT=*
6 //CEEDUMP DD SYSOUT=*
7 //ARSYLIST DD SYSOUT=*
8 //ARSYPARM DD DISP=SHR,DSN=control.data.set.name
  
```

Figure 19: JCL required for the operation of ARSYSPIN

The JCL statements are described in the following figure.

Statement	DDNAME	Description/Notes
1		The EXEC statement should specify a REGION size of 0M. Currently, no PARM text is expected. If PARM text is supplied, it is ignored. However, to ensure compatibility with future enhancements, the EXEC statement PARM parameter should not be specified.
2	STEPLIB	Depending upon your system configuration, the STEPLIB DD statement may or may not be required. If a STEPLIB DD statement is needed, all of the referenced libraries must be APF Authorized. The CEE.* and CBC.* libraries provide the C and other Language Environment runtime support modules. The DB2.* libraries provide DB2 runtime support modules. The *.SARSLOAD library is where ARSYSPIN and other related Content Manager OnDemand modules (such as ARSLOAD) reside. The *.SAPKMOD1 library is where APKACIF resides
3	SYSIN	This DD statement is used by the C runtime for the STDIN stream. The statement should be specified exactly as shown.
4	SYSPRINT	This DD statement is used by the C runtime for the STDOUT stream. See <i>z/OS Language Environment Programming Guide</i> , (SA22-7561) for details regarding the characteristics of the file referenced by this DD statement.
5	SYSOUT	This DD statement is used by the C runtime for the STDERR stream. See <i>z/OS Language Environment Programming Guide</i> , (SA22-7561) for details regarding the characteristics of the file referenced by this DD statement.

Table 15: JCL statements for ARSYSPIN (continued)

Statement	DDNAME	Description/Notes
6	CEEDUMP	This DD statement is used by the Language Environment runtime for the production of serviceability and traceback information in the event of a runtime detected failure. See <i>z/OS Language Environment Programming Guide</i> , (SA22-7561) for details regarding the characteristics of the file referenced by this DD statement.
7	ARSYLIST	The file referenced by this DD statement contains the messages produced by ARSYSPIN. Any file supported by QSAM may be specified. This file is always written as RECFM=FBA, LRECL=133. If a BLKSIZE value is not supplied, a system determined BLKSIZE value is used.
8	ARSYPARM	The file referenced by this DD statement contains the control statement stream. Any file supported by QSAM may be specified. LRECL: Any record length up to a maximum value of 512 is accepted. The smallest record length is that which is needed to fully contain a keyword/operand pair. From a practical point of view, the smallest record length is 80. RECFM: Any of the following are supported: Fixed (RECFM=F) Fixed Blocked (RECFM=FB) Variable (RECFM=V) Variable Blocked (RECFM=VB)

Control statements

Control statements have the following general format:

```
label      keyword=operand      comment
```

Where:

label

Is an optional string of non-blank characters. If a given statement does not use a label, the statement must begin with at least one blank. A label of an asterisk is used to designate a comments statement; the remainder of the statement is ignored.

keyword

Is one of the keywords listed in the table "Operand characteristics". The keyword string may be specified in upper, lower, or mixed case. A keyword is separated from its operand by an equal sign, which may in turn, be surrounded by blanks.

operand

Is the value associated with the keyword. See the table "Operand types" for a the description of operand types.

comment

Is any text, separated from the operand by at least one blank.

A statement can contain one, and only one, keyword=operand pair and it cannot be continued across multiple logical records. Blank lines are ignored.

The table "Operand characteristics" describes the general characteristics of the operands for each of the control statement keywords.

- The Type column indicates the type of operand that must be specified for a given keyword.
- The AS IS column indicates if the case of the operand is accepted AS IS (that is, the column contains the word Yes) or is converted to Upper Case (that is, the column is blank).
- The Wild Card column indicates if the operand may be specified using wild card characters. When wildcard characters are allowed in an operand, a question mark (?) is used to represent a single character and an asterisk (*) is used to represent zero or more characters.
- The Max Length column indicates the maximum length operand which may be specified; this length does not include operand delimiters (that is, surrounding quotes or parentheses).
- The ARSLOAD column identifies the corresponding ARSLOAD parameter.
- The Default column indicates the default value for the operand.

Table 16: Operand characteristics

Keyword	Type	AS IS	Wild Card	Max Length	ARSLOAD	Default
APPL	Qstring	Yes		60	-a	null
APPLGROUP	Qstring	Yes		60	-g	null
CAPDSKEEP	Boolean					False
DSNPFX	String			26		False
ERROPT	String			8		Accept
GTRACE	Integer			4		null
HALTLIMIT	Integer			15		180
JESCLASS	String			36		Z
JESNAME	String			4		The JES under which ARSYSPIN is executing.
JOBBREAK	Boolean					True
LOADPGM	String			8		ARSLOAD
MAXDORM	Integer			15		120
MAXJESDELAY	Integer			15		60
MAXLPP	Integer			15		0
MAXSFC	Integer			15		100
NOSPINDELAY	Boolean			5		False
ODHOST	String	Yes		128	-h	null
ODINSTANCE	String	Yes		20	-I	null
ODUSER	String	Yes		20	-u	null
ODUSERPW	String	Yes		20	-p	null
OUTCC	String			8		ASA

Table 16: Operand characteristics (continued)

Keyword	Type	AS IS	Wild Card	Max Length	ARSLOAD	Default
OUTSEP	Boolean					True
REPORTID	String	Yes		8		blanks
RSADEXIT	Boolean			5	-E	False
SELDEST	String		Yes	18		null
SELFORM	Pstring		Yes	73		null
SELHOLD	Boolean					True
SELWTR	String		Yes	8		null
STORLIMIT	Pstring			32		(NOLIMIT, ABEND)
TEMPPATH	String	Yes		255	-c	null
TEMPUNIT	String			8		SYSALLDA
USERSTRING	Qstring	Yes		128		null
UXnn	Pstring			32		No
ZSTRING	Qstring	Yes		128	-Z	null

Table 17: Operand Types

Operand	Description
Boolean	<p>Indicates the operand must be specified as a boolean.</p> <p>The following values can be used to indicate a value of True: True, Yes, T, Y, 1</p> <p>The following values can be used to indicate a value of False: False, No, F, N, 0</p> <p>Example: JobBreak=Yes</p>
Integer	<p>Indicates the operand must be specified as a series of decimal digits. Example: MaxDom=120</p>
Pstring	<p>Indicates the operand must be specified as a parenthesized list of comma separated items. Items in the list may be surrounded by blanks, which are ignored. Example: SelForm=(Form1, Form2)</p>
Qstring	<p>Indicates the operand must be specified as a quoted string. All blanks in the string (including leading, trailing and embedded) are considered to be a part of the string. Example: Appl="My Application"</p>
String	<p>Indicates the operand must be specified as a string of characters containing no embedded blanks. Example: JESname=JES2</p>

Keyword descriptions

APPL

Specifies the Content Manager OnDemand Application Name. See the description of the -a parameter in [Chapter 24, “ARSLOAD,” on page 185](#).

Operand type:

Quoted String

Maximum Length:

60 characters

Case:

Mixed

Default:

null

Corresponding ARSLOAD parameter:

-a

Example:

Specify the application name **JCL Capture**

```
APPL="JCL Capture"
```

APPLGROUP

Specifies the Content Manager OnDemand Application Group Name. See the description of the -g parameter in [Chapter 24, “ARSLOAD,” on page 185](#).

Operand type:

Quoted String

Maximum Length:

60 characters

Case:

Mixed

Default:

null

Corresponding ARSLOAD parameter:

-g

Example:

Specify the application group name Job Logs

```
APPLGROUP="Job Logs"
```

CAPDSKEEP

Specifies if the intermediate capture data sets are to be kept or deleted after being processed by ARSLOAD. The primary use of keeping an intermediate capture data set is to facilitate the construction of indexing parameters with the Content Manager OnDemand Report Wizard.

CAPDSKEEP is ignored if the DSNPFx parameter is not specified.

Operand type:

Boolean

Default:

False

Example:

Indicate the intermediate capture data sets are to be kept.

```
CAPDSKEEP=YES
```

DSNPFX

Specifies the left-most 26 characters of the data set name string used when creating intermediate data sets. A given intermediate data set contains the captured spool file data which is used as input to the load program.

The right-most two qualifiers of the data set name are constructed to be of the form: *Dyyyyddd.xhmmss.th*. The *yyyyddd* portion of the first of these qualifiers is the current system date. The second of these qualifiers is the current system time, where the *x* is one of the letters A, B or C, corresponding to the first digit of the hour being 0, 1 or 2, respectively.

When the DSNPFX parameter is specified, the effective data set disposition is a function of the CAPDSKEEP parameter. When CAPDSKEEP=YES is specified, the effective disposition is NEW,CATLG,CATLG. When CAPDSKEEP=NO is specified, the effective disposition is NEW,DELETE,CATLG.

If the DSNPFX parameter is not specified, the intermediate data sets are allocated as temporary data sets with an effective disposition of NEW,DELETE,DELETE.

Operand type:

String

Maximum Length:

26 characters

Case:

Converted to Upper Case

Default:

null

Example:

Specify the leftmost portion of the generated data set name string as OPS.TRACKING

```
DSNPFX=OPS.TRACKING
```

If the current system date is day 95 of year 2008 and the time is 11:30:15.65, the generated data set name is:

```
OPS.TRACKING.D2008095.B1301565
```

ERROPT

Specifies the action to be taken by ARSYSPIN when spool file I/O errors are encountered.

The operand must be one of the following:

ABEND

ARSYSPIN terminates processing via ABEND. All of the spool files which had been captured in the current cycle remain available for selection.

ACCEPT

The error is treated as though an End-of-File condition had been encountered. Processing continues normally.

HOLD

The error is treated as though an End-of-File condition had been encountered. The spool file that encountered the error is placed in a System Hold (JES2 HOLDRC=6) state. The remainder of processing continues normally.

Operand type:

String

Default:

ACCEPT

Example:

Indicate ARSYSPIN is to ABEND upon encountering a spool file I/O error.

```
ERROPT=ABEND
```

GTRACE

The GTRACE control statement is used to enable diagnostic tracing. Trace events are recorded using the MVS Generalized Trace Facility (GTF).

The operand specifies the GTF User Event ID to be used, and is a decimal integer within the range 0 through 1023, inclusive.

Specifying this control statement enables diagnostic tracing; omitting this control statement disables diagnostic tracing.

Refer to *z/OS MVS Diagnosis: Tools and Service Aids (GA22-7589)* for general information on GTF.

Refer to *z/OS MVS interactive Problem Control System (IPCS) Commands (SA22-7594)* for information about processing and formatting GTF trace data.

Operand type:

Integer

Maximum Length:

4 digits

Default:

GTF tracing is disabled

Example:

Indicate that GTF tracing is to be enabled, using GTF User Event ID 137 (decimal).

```
GTRACE=137
```

HALTLIMIT

Specifies the maximum amount of time (in whole seconds) to wait for subtasks to terminate when ARSYSPIN is in the process of terminating.

If this time period expires and there are still subtasks that have not terminated, those subtasks are forcibly terminated via CALLRTM, specifying ABEND Code U0039, Reason Code x'047E'.

If the specified value is less than or equal to zero, the default value is used.

If the specified value is greater than the maximum permissible value (900), the maximum value is used.

Operand Type:

Integer

Maximum Length:

15 digits

Default:

180

Maximum:

900

Example:

Indicate that ARSYSPIN is to allow subtasks up to 10 minutes to terminate during ARSYSPIN shutdown processing.

```
HaltLimit=600
```

JESCLASS

Identifies the set of SYSOUT classes to be processed by ARSYSPIN. The operand should be specified as a string that may consist of the characters A through Z and the digits 0 through 9.

Operand Type:

String

Maximum Length:

36 characters

Case:

Converted to upper case

Default:

Z

Example:

Indicate that SYSOUT classes H, K, and N are to be processed.

```
JESCLASS=HKN
```

JESNAME

Identifies the name of the Job Entry Subsystem with which ARSYSPIN is to interact.

If JES3 is specified, then ARSYSPIN must itself be executing under the control of JES3. If this condition is not satisfied, ARSYSPIN will terminate by ABEND (U0039, Reason Code X'0490').

Operand type:

String

Maximum Length:

4 characters

Case:

Converted to Upper Case

Default:

The JES under whose control ARSYSPIN is executing.

Example:

Indicate ARSYSPIN is to interact with a Job Entry Subsystem named JEST

```
JESNAME=JEST
```

JOBBREAK

Indicates if a given collection of captured spool files is to contain the data for one job (JOBBREAK=YES) or for multiple jobs (JOBBREAK=NO).

When JOBBREAK=YES is specified, the MAXDORM and MAXSFC parameters are ignored.

When JOBBREAK=NO is specified, the MAXDORM and MAXSFC parameters work in concert as follows:

- Spool files for a given job that satisfy the specified selection criteria are gathered into the current collection.
- Once JES indicates there are no additional Spool files for the Job currently being processed, the MAXSFC limit value is checked. If that limit has now been reached or

exceeded, the current collection is regarded as complete and the load program (usually ARSLOAD) is invoked to process (that is, store) the collection.

- If the MAXSFC limit has not been reached, a timer is set whose value (in seconds) was specified via the MAXDORM parameter. If additional Spool files (for any job that satisfies the selection criteria) become available before this timer expires, those data sets are added to the current collection.
- Once the timer expires, the current collection is regarded as complete.

Important: When possible (and independently of the JOBBREAK specification) the Spool files for a given job are contained in the same collection. See information about the MAXSFC keyword.

Regardless of whether JOBBREAK=YES or JOBBREAK=NO is specified, the Spool files for a given Job are always contained in the same collection.

Operand type:

Boolean

Default:

True

Example:

Indicate the Spool files for multiple jobs are to be gathered into a single collection, the approximate maximum number of Spool files is **100**, and the maximum amount of time to wait for the arrival of additional Spool files is **two minutes**.

```
JobBreak=No
MAXSFC=100
MaxDorm=120
```

LOADPGM

Provides the name of the program to be used to process captured Spool file collections. The principal use of this parameter (combined with specifying CAPDSKEEP=YES) is to facilitate the construction of indexing parameters with the Content Manager OnDemand Report Wizard.

The specified program is invoked as a subtask (via ATTACH) within the ARSYSPIN address space and must reside within an APF Authorized library. The processing environment at entry is:

PSW:

Problem State,Primary ASC Mode,Same PSW Key as ARSYSPIN

Authorization:

APF Authorized

DU-AL:

Not shared with ARSYSPIN

The PARM text presented to the specified program is what would normally be presented to ARSLOAD. See [Chapter 24, "ARSLOAD," on page 185](#) for further details regarding ARSLOAD command line parameters.

Operand type:

String

Maximum Length:

8 characters

Case:

Converted to Upper Case

Default:

ARSLOAD

Example:

Retain the intermediate capture data set and specify IEFBR14 as the load program. The capture data set can then be used as input to the Content Manager OnDemand Report Wizard to more easily create indexing specifications.

```
CAPDSKEEP=YES  
LOADPGM=IEFBR14
```

MAXDORM

Specifies the maximum amount of time (in seconds) that ARSYSPIN is to wait for the arrival of additional Spool files which satisfy the current selection criteria. Once this period elapses, the current collection is regarded as complete.

This parameter is honored only when JOBBREAK=NO is specified; otherwise, if specified, it is ignored. See the description of the JOBBREAK and MAXSFC parameters for further details regarding how these parameters interact with one another.

Operand type:

Integer

Maximum Length:

15 digits

Default:

120

Example:

Indicate that Spool files for multiple jobs are to be gathered into a single collection, the approximate maximum number of Spool files is 150, and the maximum amount of time to wait for the arrival of additional Spool files is five minutes.

```
JobBreak=No  
MAXSFC=150  
MaxDorm=300
```

MAXJESDELAY

When requests for some JES services are performed, JES may not be able to successfully complete the requested service or function at the time of the receipt of the request. Delaying for a short time and then retrying the request generally results in successful request execution. The Extended Status Subsystem Request is an example of such a service.

The MAXJESDELAY control statement specifies the maximum amount of time (in seconds) over which ARSYSPIN should iterate on this category of JES service requests.

When this time period has elapsed and the requested JES service has still not succeeded, the request is regarded as having failed. Using the Extended Status Subsystem Request as an example, this failure results in the corresponding status information being regarded as unavailable.

MAXJESDELAY specifies the overall time period only; it does not specify the delay period between JES service request iterations (this value is internal to ARSYSPIN).

If the specified value is less than or equal to zero, the default value is used.

If the specified value is greater than the maximum permissible value (900), the maximum value is used.

Operand Type:

Integer

Maximum Length:

15 digits

Default:

60

Maximum:

900

Example:

Indicate that ARSYSPIN is to iterate for a maximum of 75 seconds when attempting JES service requests.

```
MaxJesDelay=75
```

MAXLPP

Specifies the size of a logical page in terms of lines. When this parameter is specified, ARSYSPIN maintains the notion of a logical page by "counting lines". If a page break does not occur within the data stream before the MAXLPP limit is reached, ARSYSPIN forces a page break. A page break within the data stream is indicated when a given Spool file possesses either ASA or Machine control characters and one of the following control characters is encountered:

X'F1'

ASA: Skip to Channel 1 and Print

X'8B'

Machine: Skip to Channel 1 Immediate

X'89'

Machine: Print and Skip to Channel 1

If this parameter is not specified or if an operand value of zero is specified, line counting is not performed.

Operand type:

Integer

Maximum Length:

15 digits

Default:

0

Example:

Indicate line counting is to be performed and the logical page size is 70 lines.

```
MAXLPP=70
```

MAXSFC

Specifies the maximum number of Spool files to be gathered into a single collection. Once this limit is reached or exceeded, the current collection is regarded as complete.

This parameter is honored only when JOBBREAK=NO is specified; otherwise, if specified, it is ignored. See the description of the JOBBREAK and MAXDORM parameters for further details regarding how these parameters interact with one another.

Note that, when possible, the Spool files for a given job are always contained in the same collection. Therefore, the specified limit should be regarded as an approximate upper bound and it is possible that the MAXSFC value may be exceeded.

The upper limit on the number of Spool files that may be contained in a single collection is 3071 (decimal). When this limit is reached, the current collection is regarded as complete, and the collection is subsequently processed as though the StorLimit processing action of TruncRetry had been specified, causing the following actions to be performed:

- ARSYSPIN User Exit 7 (StorLimit processing) is invoked, indicating that the TruncRetry processing option is intended to be performed. The exit may accept or alter this intended processing action.
- Assuming the TruncRetry processing option remains in effect, the current collection (which does not include the Spool file that caused the limit value to be reached) is processed by the load program and, on completion, a new collection is initiated, beginning with the Spool file that was being processed when the limit value was reached.
- If the User Exit changed the intended StorLimit processing action, that action is performed.

Operand type:

Integer

Maximum Length:

15 digits

Maximum Value:

3071

Default:

100

Example:

Indicate that Spool files for multiple jobs are to be gathered into a single collection, the approximate maximum number of Spool files is 150, and the maximum amount of time to wait for the arrival of additional Spool files is five minutes.

```
JobBreak=No
MAXSFC=150
MaxDorm=300
```

MINSPACE

Important: The function provided by this parameter is no longer supported. If specified, the statement is syntax-checked, but is otherwise ignored.

This parameter value is multiplied by 256 MB to determine the initial size of the capture data space.

Operand type:

Integer

Maximum Length:

1 digit

Maximum Value:

8

Default:

1

Example:

Specify the capture data space to be the maximum size (2GB - 1 page).

```
MINSPACE=8
```

NOSPINDELAY

Specifies that after reading a spool data set that was spun, no delay will be made while waiting for job status. This has the effect that the separator records for the spool file may not contain the job ending status. The lack of a delay may also affect what spool files are gathered by ARSYSPIN due to timing.

Operand type:

Boolean

Default:

False

Example:

Indicate no delay for spun data sets.

```
ODHOST=MyLibServer
```

ODHOST

Specifies the name of the Content Manager OnDemand library server. See [Chapter 24, “ARSLOAD,” on page 185](#) for a description of the ARSLOAD -h parameter for further details.

Operand type:

String

Maximum Length:

128 characters

Case

Mixed

Corresponding ARSLOAD parameter:

-h

Default:

null

Example:

Specify the name of the Content Manager OnDemand library server as MyLibServer.

```
ODHOST=MyLibServer
```

ODINSTANCE

Specifies the Content Manager OnDemand Instance name. See [Chapter 24, “ARSLOAD,” on page 185](#) for a description of the ARSLOAD -I parameter for further details.

Operand type:

String

Maximum Length:

20 characters

Case

Mixed

Corresponding ARSLOAD parameter

-I

Default:

null

Example:

Specify the Content Manager OnDemand Instance name as Production.

```
ODINSTANCE=Production
```

ODUSER

Specifies a Content Manager OnDemand User name (with administration authority) on whose behalf ARSLOAD is to execute. See [Chapter 24, “ARSLOAD,” on page 185](#) for a description of the ARSLOAD -u parameter for further details.

Operand type:

String

Maximum Length:

20 characters

Case

Mixed

Corresponding ARSLOAD parameter:

-u

Default:

null

Example:

Specify the OnDemand User Name as Admin.

```
ODUSER=Admin
```

ODUSERPW

Specifies the logon password for the Content Manager OnDemand User name specified by the ODUSER parameter. See [Chapter 24, “ARSLOAD,” on page 185](#) for a description of the ARSLOAD -p parameter for further details.

Operand type:

String

Maximum Length:

20 characters

Case

Mixed

Corresponding ARSLOAD parameter:

-p

Default:

null

Example:

Specify the Content Manager OnDemand User Name as Admin with a logon password of PASSWORD.

```
ODUSER=Admin  
ODUSERPW=PASSWORD
```

OUTSEP

Indicates if the ARSYSPIN generated separator records that surround captured Spool files are to be generated.

Operand type:

Boolean

Default:

True

Example:

Indicate that separator records are not to be produced.

```
OUTSEP=No
```

REPORTID

Provides a string that is placed in the separator records that surround each captured Spool file. This parameter is provided solely for compatibility with a similar function offered in Content Manager OnDemand for OS/390 Version 2.

Operand type:

String

Maximum Length:

8 characters

Case

Mixed

Default:

blanks

Example:

Specify a Report ID string of Sample.

```
REPORTID=Sample
```

RSADEXIT

Indicates if the Report Specifications Archive Definition exit is to be enabled. See *IBM Content Manager OnDemand for z/OS: Configuration Guide* for details regarding the operation and other characteristics of this logical exit point.

Operand type:

Boolean

Corresponding ARSLOAD parameter:

-E

Default:

False

Example:

Indicate that Report Specifications Archive Definition exit is to be enabled.

```
RSADEXIT=Yes
```

SELDEST

Specifies a JES destination ID to be used as a Spool file selection filtering criteria. See the description of the OUTPUT JCL statement DEST parameter in *z/OS MVS JCL Reference*, SA22-7597 for details regarding JES destination IDs.

Wild card characters may be specified.

If the SELDEST parameter is not specified, then selection filtering by destination ID is not performed.

Operand type:

String

Maximum Length:

18 characters

Case

Converted to Upper Case

Default:

null

Example:

Indicate that Spool files are to be selected which were to be routed to user SIMPSON on system POKVMB.

```
seldest=ODADMIN
```

SELFORM

Specifies a set of up to eight Forms names to be used as Spool file selection filtering criteria. See the description of the OUTPUT JCL statement FORMS parameter in *z/OS MVS JCL Reference*, SA22-7597 for details regarding Form names.

Wild card characters may be specified.

If the SELFORM parameter is not specified, then selection filtering by Form names is not performed.

Operand type:

Parenthesized string

Maximum Length:

Up to eight, 8-character comma separated forms names

Case

Converted to Upper Case

Default:

null

Example:

Indicate that Spool files are to be selected which have an associated Forms name of either GREENBAR or any string beginning with the characters XYZ.

```
SELFORM=(GREENBAR, XYZ*)
```

SELHOLD

Specifies whether Spool files are selected from the JES 'hold' queues.

Operand type:

Boolean

Default:

True

Example:

Indicate that Spool files are not to selected if in the 'held' queue.

```
SELHOLD=FALSE
```

SELWTR

Specifies a SYSOUT Writer name to be used as a Spool file selection filtering criteria. See the description of the OUTPUT JCL statement FORMS parameter in *z/OS MVS JCL Reference*, SA22-7597 for details regarding SYSOUT Writer names.

Wild card characters may be specified.

If the SELWTR parameter is not specified, then selection filtering by Writer name is not performed.

Operand type:

String

Maximum Length:

8 characters

Case

Converted to Upper Case

Default:

null

Example:

Indicate that Spool files are to be selected that have an associated Writer name of JCLWTR.

```
SELWTR=JCLWTR
```

STORLIMIT

The STORLIMIT control statement specifies the maximum amount of data (in bytes) to be placed into a given aggregate (the set of SYSOUT data to be stored into Content Manager OnDemand as a single entity).

The operand must be enclosed within parentheses.

An omitted subparameter is indicated by a comma acting as a placeholder. For example, STORLIMIT=(, TruncRetry)

The operand is specified in the form (limit size specification, action), where *limit size specification* is a positive decimal integer that may optionally be followed by a scaling character:

- M represents Mega units (1M=1024*1024=1 048 576)
- G represents Giga units (1G=1024*1024*1024=1 073 741 824)

If a scaling character is not supplied, the value is interpreted in Mega units. For example, a specification of 512 is interpreted as (and is synonymous with) 512M.

The string NOLIMIT may be specified to indicate that ARSYSPIN is not itself to impose any limit on the size of the capture data space (non-z/OS only), nor the quantity of 64-bit addressable storage to be used to hold the intermediate data.

An implicit limit on the size of the capture data space or the 64-bit addressable storage object can be imposed through several mechanisms that are outside the scope of ARSYSPIN control. Examples include:

- The architectural limit on the size of a data space is 2GB.
- SMF exit IEFUSI can be used to limit the quantity of 64-bit addressable storage used by an address space.

Therefore, it is possible that an implicit limit value may be imposed that is less than the specified limit value.

action specifies the action to be performed when the limit is reached, and may be one of the following:

- **ABEND** ARSYSPIN is to ABEND with a Reason Code indicating that the limit value has been reached. Prior to an ABEND, all the Spool files in the current aggregate (including the Spool file currently being processed) are placed into a System Hold state with Hold Reason Code 6 (Held by SYSOUT API).
- **TRUNCHOLD** The current collection is truncated at the end point of the most recent successfully captured Spool file. The Spool file currently being processed is placed into a System Hold state with Hold Reason Code 6 (Held by SYSOUT API).
- **TRUNCRETRY** The current collection is truncated at the end point of the most recent successfully captured Spool file. On completion of processing of the current collection, the Spool file that caused the limit value to be reached is reprocessed from its beginning. If this Spool file itself exceeds the limit value, the TRUNCHOLD action is performed.

Operand Type:

Parenthesized string

Default:

(NOLIMIT, ABEND)

Example:

Indicate ARSYSPIN is to limit the collection size to 768MB. When the size limit is reached, the TruncRetry action is to be performed.

```
StorLimit=(768M, TruncRetry)
```


Example:

Indicate ARSYSPIN is to use the default collection size limit value of NOLIMIT. When the effective size limit is reached, the TruncRetry action is to be performed.

```
StorLimit=(,TruncRetry)
```

Example:

Indicate that ARSYSPIN is to limit the collection size to 1GB. When the size limit is reached, the default ABEND action is to be performed.

```
StorLimit=(1GB)
```

TEMPPATH

Identifies the path to an HFS directory where temporary files created during the indexing and store processing are to be placed. See Chapter 24, “ARSLOAD,” on page 185 for a description of the ARSLOAD -c parameter for further details.

Note: If ARSLOAD is being used as the load program and TEMPPATH is not specified, the home directory associated with the user ID on whose behalf ARSYSPIN is executing is used as the effective TEMPPATH value.

Operand type:

String

Maximum Length:

255 characters

Case

Mixed

Corresponding ARSLOAD parameter:

-c

Default:

null

Example:

Indicate that temporary HFS files are to be placed in the /usr/tmp directory.

```
TEMPPATH=/usr/tmp
```

TEMPUNIT

Identifies the MVS Unit Name to be used when dynamically allocating intermediate data sets.

Operand type:

String

Maximum Length:

8 characters

Case

Converted to Upper Case

Default:

SYSALLDA

Example:

Specify an MVS Unit Name of SYSDA.

```
TEMPUNIT=SYSDA
```

USERSTRING

Specifies an arbitrary string that is presented to all ARSYSPIN user exits. The operand is accepted as is (that is, without case conversion). ARSYSPIN itself neither interprets nor modifies this string.

Operand type:

Quoted String

Maximum Length:

128 characters

Case

Mixed

Default:

null

Example:

```
USERSTRING="Hello World"
```

UXnn

The *UXnn* statement is used to specify the state of an ARSYSPIN logical exit point, and provides the name of a user written module to be associated with that exit point.

The *nn* portion of the keyword is a two digit decimal number corresponding to a logical exit point.

The operand must be specified in one of the following forms. (Note that the first suboperand is a boolean value as previously described.)

(NO)

Indicates the logical exit point is permanently disabled.

(NO,*name*)

Associates the module named *name* with the logical exit point, but the exit point is currently disabled. The exit point may subsequently be enabled or disabled via an MVS Modify command.

(YES,*name*)

Associates the module named *name* with the logical exit point, and the exit point is currently enabled. The exit point may subsequently be disabled or enabled via an MVS Modify command.

Restriction: Support for the changing of the state of an exit routine through the use of the MVS Modify command is not currently provided. This support may be provided in a subsequent PTF.

Operand type:

Parenthesized String

Operand Length:

The maximum length of the *name* suboperand is 8 characters.

Case

Converted to Upper Case

Default:

NO

Example:

Indicate User Exit 1 is never to be invoked.

```
UX01=(NO)
```

Example:

Indicate User Exit 2 is never to be invoked.

```
UX02=(NO)
```

Example:

Associate the module named MYUX03 with logical exit point 3, but currently that exit point is disabled and module MYUX03 will not be invoked.

```
UX03=(NO,MYUX03)
```

Example:

Enable logical exit point 3 and associate the module named MYUX03 with that exit point.

```
UX03=(YES,MYUX03)
```

ZSTRING

Specifies a mixed case string to be used as the operand of the ARSLOAD -Z command line argument. See [Chapter 24, "ARSLOAD," on page 185](#) for further details regarding the intended use of the ARSLOAD -Z command line argument.

Operand type:

Quoted String

Maximum Length:

128 characters

Case

Mixed

Corresponding ARSLOAD parameter:

-z

Default:

null

Example:

```
ZSTRING="ODFDBOwner"
```

ARSYSPIN user exits

Overview

Several logical exit points are provided to allow user written exit routines to modify the behavior of ARSYSPIN processing. The following exit points are provided:

<i>Table 18: Logical exit points that allow user written exit routines to modify the behavior of ARSYSPIN processing</i>	
Exit point	Exit routine
Initialization	Invoked during the initialization of the ARSYSPIN Subsystem Interface Task after all control statement processing has completed and before any interactions with JES take place.
Termination	Invoked during the termination of the ARSYSPIN Subsystem Interface Task.

Table 18: Logical exit points that allow user written exit routines to modify the behavior of ARSYSPIN processing (continued)

Exit point	Exit routine
Input Record Processing	Invoked each time a record has been read from the current input data source. The exit can: <ul style="list-style-type: none"> • alter, replace, discard or accept the record as is, • indicate if a logical End-of-Data condition is to be recognized, or • indicate that the current capture stream is to be regarded as complete.
Store Processing	Invoked immediately prior to the invocation of the load program, that is, the program identified by the LOADPGM initialization control statement. The exit can alter the majority of the parameters which effect store processing (for example, the names of the Content Manager OnDemand application and application group).
Separator Record Processing	Invoked when the separator records surrounding each captured Spool file are being produced. The exit can alter, replace, or suppress the ARSYSPIN standard separator records, as well as introduce its own separator records.
Collection Size (StorLimit) Limit Processing	Invoked when the current "collection" has reached its maximum size limit value. The exit can accept or override the intended StorLimit processing action.

Additional exit points may be provided in the future.

Processing environment

All exit routines:

- Must reside in an APF Authorized library, but must not possess the AC=1 attribute.
- Must be written to support re-entrant execution and must possess the re-entrant attribute.
- Can reside in either a PDS Load Library or a PDSE Program Object Library. The normal MVS search order (STEPLIB/JOBLIB, LPA, LNKLST) is used to locate exit routines.
- Cannot reside in the HFS.
- Should restore all registers, with the exception of General Register 15 (which must contain a return code) prior to exiting.
- Must exit by branching to the address contained in General Register 14 at entry to the exit routine.

The processing environment at entry to a given exit routine is as follows:

Dispatchable Unit Mode:

Enabled, Unlocked Task Mode

PSW Conditions:

- Problem State
- Primary ASC Mode
- 31-bit Addressing Mode
- Non-system Protect Key

Cross Memory Mode:

PASN=SASN=HASN

General Registers:**R1**

Points to the principal interface parameter list as mapped by ARSYUXPL.

R13

Points to a standard 18-word general register save area.

R14

Contains the address to which the exit routine must branch upon its completion, that is, the return address.

R15

Contains the exit routine entry point address.

If an exit routine has an ABEND or returns an undefined return code, the exit routine is disabled and is not called again.

A storage area intended to be shared among the exit routines is acquired by ARSYSPIN and is presented to each exit routine via a descriptor contained within the ARSYXPCM structure. This area is aligned on a 4K Page boundary and is guaranteed to be at least 8K bytes in length.

Data areas

The following table identifies the data areas presented to exit routines. Mapping macros for these data areas are provided in the SARSINST data set. See the assembly listings of these data areas for details regarding the content and intended usage of the fields within each of these structures.

<i>Table 19: ARSYSPIN exit routine data areas</i>		
Macro Name	DSECT Name	Description
ARSYUXPL	ARSYUXPL	Maps the principal interface parameter list pointed to by General Register 1 at entry to a given exit routine.
ARSYUXVD	ARSYUXVD	Maps the descriptor of an exit routine. Exit routine descriptors are contained within the ARSYUXVT structure.
ARSYUXVT	ARSYUXVT	Maps the User Exits Vector Table.
ARSYXOPA	ARSYXOPA	Maps the overall execution options structure.
ARSYXP	ARSYXP	Maps the interface data structures which are unique to each of the exit points.
ARSYXPCM	ARSYXPCM	Maps the interface data structure which is common to all exit points.

A mapping listing for reference can be produced by assembling the following source stream:

```

TITLE 'ARSYUXPL: SPIN User Exit Interface Parameter List'
ARSYUXPL DSECT=YES
*
TITLE 'ARSYUXVD: SPIN User Exit Vector Descriptor'
ARSYUXVD DSECT=YES
*
TITLE 'ARSYUXVT: SPIN User Exit Vector Table'
ARSYUXVT DSECT=YES
*
TITLE 'ARSYXOPA: SPIN Execution Options'
ARSYXOPA DSECT=YES
*
```

```

TITLE 'ARSYXP: SPIN User Exit Data Structures'
ARSYXP DSECT=YES
*
TITLE 'ARSYXPCM: SPIN User Exit Common Information Structure'
ARSYXPCM DSECT=YES
*
END ,

```

The following figure illustrates how the data areas are interconnected.

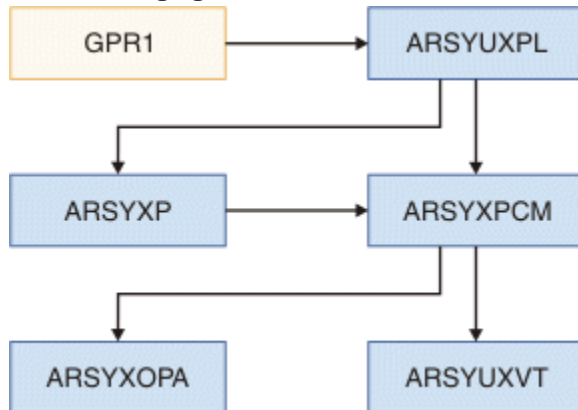


Figure 20: Interconnection of data areas in ARSYSPIN exit routines

Logical exit points

User Exit 1: Initialization

User Exit 1 is invoked during the initialization of the ARSYSPIN Subsystem Interface Task after all control statement processing has completed and before any interactions with JES take place. The initialization exit is not invoked if any errors are detected during the processing of the control statement stream. This exit point is intended to be used to establish relatively long lived structures or connections to other subsystems, such as DB2.

Note: If the Subsystem Interface Task terminates and is subsequently reinstated, User Exit 1 is re-driven.

User Exit 2: Termination

User Exit 2 is invoked during the termination of the ARSYSPIN Subsystem Interface Task. This exit point is intended to be used to perform clean up activities such as the releasing of storage areas acquired and the severing of subsystem connections established during the execution of Exit 1.

User Exit 3: Input Record Processing

User Exit 3 is invoked each time a record has been read from the current input data source.

Interface items of particular interest are:

- A pointer to a Read Only copy of the SAPI SSOB Extension (mapped by the SSS2 DSECT produced by the IAZSSS2 macro) corresponding to the SYSOUT data set being processed.
- A pointer to a Read Only copy of the Extended Status Job Queue Element (mapped by the STATJQTR DSECT produced by the IAZSSST macro) corresponding to the Job that produced the SYSOUT data set being processed.

The exit can:

- Alter, replace, discard, or accept the record as is.
- Indicate if a logical End-of-Data condition is to be recognized.
- Indicate that the current capture stream is to be regarded as complete. Note that when this condition is signaled by the exit routine, the current input record is not regarded as belonging to the current capture stream.

The effective disposition of the current record is returned as part of the feedback information from the exit. This disposition is independent of the exit return code and can be one of the following:

- All processing of the record is complete. Processing is to continue with the next logical record (if any additional records exist) from the current input source.
- The current record is to be presented to the Input Exit again.

The effective disposition of the current input source is returned as part of the feedback information from the exit. This disposition is independent of the exit return code and can be one of the following:

- Normal processing of the current input source is to continue.
- Logical End-of-Data; further processing of the current input source is to be discontinued. When input processing resumes, the first record from the next (if any) input source stream is processed

User Exit 5: Store Processing

User Exit 5 is invoked immediately prior to the invocation of the load program, that is, the program identified by the LOADPGM initialization control statement. The exit can alter the majority of the parameters which effect store processing (for example, the names of the Content Manager OnDemand application and application group). The exit can also indicate that the load program is not to be invoked.

User Exit 6: Separator Record Processing

User Exit 6 is invoked when ARSYSPIN is about to write a separator record to the output stream. If output separator records are not to be produced (as a result of specifying OUTSEP=NO in the control statement input stream), then User Exit 6 is not invoked. Interface items of particular interest are:

- A pointer to a Read Only copy of the SAPI SSOB Extension (mapped by the SSS2 DSECT produced by the IAZSS2 macro) corresponding to the SYSOUT data set being processed.
- A pointer to a Read Only copy of the Extended Status Job Queue Element (mapped by the STATJQTR DSECT produced by the IAZSSST macro) corresponding to the Job that produced the SYSOUT data set being processed.
- A pointer to the "standard", ARSYSPIN constructed, output separator record intended to be written.
- An indicator of which separator record type is being presented: the Begin record that precedes the SYSOUT data, or the End record that follows the SYSOUT data.

The exit routine may modify, replace, or delete the supplied separator record.

A "disposition" field allows the exit routine to request that it be driven again and be presented with the same ARSYSPIN constructed separator record data. The exit routine can use this mechanism to insert as many separator records into the output stream as it wants.

When the exit routine provides a separator record to be written to the output stream, it must ensure that the length of this record does not exceed the permissible maximum. The maximum permissible record length is provided in the field ARSYXP06_MAXRECL within the ARSYXP06 data structure; the minimum value in this field is 500 (decimal).

User Exit 7: StorLimit Action Processing

User Exit 7 is invoked when the current aggregate (or "collection") size has reached its maximum limit value.

The exit can accept or override the intended STORLIMIT action. The intended STORLIMIT action is that which was specified via the STORLIMIT control statement or via system defaults.

Interface items of particular interest are:

- A pointer to a Read Only copy of the SAPI SSOB Extension (mapped by the SSS2 DSECT produced by the IAZSS2 macro) corresponding to the SYSOUT data set being processed when the limit value was reached.
- A pointer to a Read Only copy of the Extended Status Job Queue Element (mapped by the STATJQTR DSECT produced by the IAZSSST macro) corresponding to the Job that produced the SYSOUT data set being processed.

- The intended STORLIMIT action.
- Several fields that reflect the current size of the aggregate in megabytes, records, and data sets.

Security system (SAF/RACF) considerations

Both JES2 and JES3 exploit SAF to provide the capability to restrict access to data stored in SPOOL data sets. In particular, the JESSPOOL resource class is used to protect SPOOL data sets (that is, SYSOUT data sets) which are to be processed by ARSYSPIN.

If the JESSPOOL resource class is active, then a profile with ALTER capability must be defined corresponding to the user ID on whose behalf ARSYSPIN is executing. If such a profile is not defined, JES2 and JES3 will reject the SPOOL access requests performed by ARSYSPIN. The rejection is generally accompanied by the following messages:

JES2

```
$HASP186 jobname NOT SELECTED BY devname DUE TO SECURITY POLICY
```

JES3

```
IAT7093 JOB jobname (jobid) NOT PROCESSED BY devname REASONS=JES(SEcurity).
```

The following RACF commands can be used to activate the JESSPOOL class and enable RACF protection of spool data sets:

```
SETROPTS CLASSACT(JESSPOOL)
SETROPTS GENERIC(JESSPOOL)
```

Profiles are not required in the JESSPOOL class for protection to be in effect because the default for the class is access denied when no profiles exist. It is recommended that generics be activated for the JESSPOOL class because the profile names are system generated.

The following RACF commands can be used to generally deny access to SPOOL resident data sets and grant access to the user ID associated with the execution of ARSYSPIN:

```
RDEFINE JESSPOOL ** UACC(NONE)
PERMIT ** CLASS(JESSPOOL) ID(userid) ACCESS(ALTER)
```

See these publications for more complete information regarding the SAF/RACF protection of JES SPOOL data sets:

<i>Table 20: Publications that contain more complete information about the SAF/RACF protection of JES SPOOL data sets</i>	
Publication	Topic
<i>z/OS JES2 Initialization and Tuning Guide</i>	Controlling Access to Data that Resides on Spool
<i>z/OS JES3 Initialization and Tuning Guide</i>	Controlling Access to Data that Resides on Spool
<i>z/OS SecureWay Security Server RACF System Programmer's Guide</i>	<ul style="list-style-type: none"> • Protecting General Resources • Providing Security for JES

ARSYSPIN files

ARSYSPIN intermediate file structure

The intermediate file written by ARSYSPIN possesses the following attributes:

RECFM

VBA. Spool files which contain Machine control characters are converted to ASA format during the capture process. Spool files which do not contain control characters are converted to ASA format with all lines containing a Single Space control character.

LRECL

The LRECL is the greater of the length of the separator record and the largest record length found in the Spool file data. The largest possible LRECL value is 32756 bytes.

BLKSIZE

A system determined block size is used. Generally, this is approximately one half of the track capacity of the unit to which the intermediate data set is written (for a 3390 device type, this is approximately 23 KB).

ARSYSPIN brackets each captured Spool file with a pair of separator records. A "Begin" separator precedes the Spool file and an "End" separator immediately follows the Spool file. The general layout of the intermediate output data set created by ARSYSPIN is depicted in the following figure..



Figure 21: Layout of the intermediate output data set created by ARSYSPIN

Mappings for the separator records are provided in several programming languages and are contained in the SARSINST data set.

Language	Member Name
S/390® Assembler	ARSYSEP
C	ARSYSEPH
COBOL	ARSYSEPB

The assembler mapping is shown in the figure and the table. Note that all offsets, lengths and values are shown in hexadecimal unless explicitly indicated otherwise.

```

**** ARSYSEP: Spool Interface Processor Separator Record
*
*
* Classification:
* Licensed materials - Property of IBM.
* This source part is "RESTRICTED MATERIALS OF IBM".
* External Classification: PSPI
*
* 5697-N93 OnDemand for z/OS
*
* (C) Copyright IBM Corp. 2001
*
* This source code is delivered "AS IS" with no warranties
* expressed or implied by the IBM Corporation.
*
*
* Function:
* Maps the logical records which exist immediately before
* and after a captured Spool data set.
*
*
* Method of Access:
* Not applicable.
*
*
* Serialization Requirements:
* None.
*
*
* Special Considerations:
* This record mapping is provided in several programming
* languages. These mappings must be kept in synch.
*
* Name Programming Language
* -----
* ARSYSEP S/390 Assembler
* ARSYSEPB COBOL
* ARSYSEPH C
*
*
* Change History:
* 04/Feb/2002 RJR Created for PQ57760.
*
****

```

Figure 22: S/390 Assembler mapping of separator records

Table 22: S/390 Assembler mapping of separator records					
Offset	Length	Value	Name	Type	Description
+0	1		ARSYSEP_CCHAR	Character	A spacing control character. When the output stream is written using Machine control characters, a No-Op (X'03') is used. Otherwise, an ASA Skip to Channel One and Write (X'F1') is used.
+1	1		ARSYSEP_SEPTYPE	Character	The separator record type.

Table 22: S/390 Assembler mapping of separator records (continued)

Offset	Length	Value	Name	Type	Description
		C'B'	ARSYSEP_SEPTBGN	Equate	The "Begin" record. This record occurs immediately before the captured data.
		C'E'	ARSYSEP_SEPTEND	Equate	The "End" record. This record occurs immediately after the captured data.
+2	8		ARSYSEP_REPTID	Character	The Report ID that was specified in the ReportID control statement.
+A	8		ARSYSEP_JOBNAME	Character	The name of the Job that created the Spool file.
+12	1	C'-'	ARSYSEP_SC01	Character	A constant.
+13	8		ARSYSEP_JOBID	Character	The JES assigned Job ID of the job that created the Spool file.
+1B	8		ARSYSEP_PROCSTEP	Character	The ProcStep name within the job that created the Spool file.
+23	8		ARSYSEP_STEP	Character	The Step name within the Job that created the Spool file.
+2B	8		ARSYSEP_DDNAME	Character	The DDNAME associated with the Spool file. This is the DDNAME used by the process that created the Spool file.
+33	8		ARSYSEP_WRITER	Character	The SYSOUT Writer name associated with the Spool file.
+3B	8		ARSYSEP_FORM	Character	The SYSOUT Form name associated with the Spool file.
+43	1		ARSYSEP_CLASS	Character	The SYSOUT Class associated with the Spool file.
+44	2C		ARSYSEP_JESDSN	Character	The JES assigned Spool file data set name.
+70	A		ARSYSEP_CRTDATE	Character	The creation date of the Spool file in the format: yyyy-mm-dd
+7A	F		ARSYSEP_CRTTIME	Character	The creation time of the Spool file in the format: hh.mm.ss.mmmmmm
+89	A		ARSYSEP_JMRDATE	Character	The date when the Job that created the Spool file was processed by the MVS JCL Converter. Format: yyyy-mm-dd
+93	F		ARSYSEP_JMRTIME	Character	The time when the Job that created the Spool file was processed by the MVS JCL Converter. Format: hh.mm.ss.mmmmmm

Table 22: S/390 Assembler mapping of separator records (continued)

Offset	Length	Value	Name	Type	Description
+A2	10		ARSYSEP_RECcnt	Character	The number of logical records contained in the captured Spool file, that is, the number of records that exist between the Begin and End separator records. This field is a string of decimal digits in character form, right justified with leading zero characters as necessary.
+B2	10		ARSYSEP_FILEsz	Character	The size of the captured Spool file expressed in kilobytes, rounded up to a kilobyte boundary. 1 kilobyte = 1024 bytes. This field is a string of decimal digits in character form, right justified with leading zero characters as necessary.
+C2	8		ARSYSEP_JENDSTAT	Character	The Job Ending State string. See Note 1 below.
+CA	3				Reserved.
+CD	E		ARSYSEP_TAG	Character	A constant tag field that can be used to identify separator records.
+CD	E	E0E07A 50C2E2 D7C9D5 C2507A E0E0	ARSYSEP_TAGB	Character	The tag value in the "Begin" separator record. In EBCDIC Code Pages 037 and 1047 this is the character string \\ :&BSPIN&: \\ \.
+CD	E	E0E07A 50C5E2 D7C9D5 C5507A E0E0	ARSYSEP_TAGE	Character	The tag value in the "End" separator record. In EBCDIC Code Pages 037 and 1047 this is the character string \\ :&ESPINE&: \\ \.
+DB	A		ARSYSEP_JENDCC	Character	Job Ending Completion Code or ABEND Codes. See Note 1 below.
+E5	3				Reserved.
		E8	ARSYSEP_STRSIZE	Character	The total size of the separator record.

Note 1

The Job Ending State string in field ARSYSEP_JENDSTAT is one of the following values:

Table 23: Possible values for the Job Ending State string in the field ARSYSEP_JENDSTAT	
String	Description
blanks	Normal completion.

Table 23: Possible values for the Job Ending State string in the field ARSYSEP_JENDSTAT (continued)

String	Description
CC	The job ended with a non-zero completion code. The ARSYSEP_JENDCC field contains the string "CC" immediately followed by the 8 decimal digit Completion Code value.
JCLERROR	The Job failed with a JCL error.
CANCEL	The Job was cancelled.
ABEND	At least one step of the Job ended with an ABEND. The ARSYSEP_JENDCC field contains the string Sxxx Udddd where xxx is the 3 hexadecimal digit System ABEND Code, and dddd is the 4 decimal digit User ABEND Code.
CNVTFail	A JCL Converter ABEND occurred while processing the Job.
SECURITY	A Security System violation occurred.
EOMFAIL	The Job failed during End of Memory processing.
UNKNOWN	The Job ending status could not be determined.
NOTAVAIL	The Job ending status could not be obtained.

ARSYSPIN error codes

ABEND codes

ABEND Code U-0013 or U0013 (X'000D') is issued when a failure or error condition occurs for which normal program execution cannot continue. An associated Reason Code (usually contained in general register 15 at the time of the ABEND) further describes the condition.

The Reason Code is an 8-digit hexadecimal value of the form **mmmmrrrr** where:

- **mmmm** (the high order 4 digits) is the Binary Coded Decimal representation of an associated message number. If these digits are 0000 no message has been issued.
- **rrrr** (the low order 4 digits) is a Reason Code that identifies the failure or error condition.

Unless otherwise indicated, the following items apply to all Reason Codes:

System Action

The system may create LOGREC records and may also create an SVC dump.

System Programmer Response

Search problem reporting databases for a solution to the problem. If no solution exists, gather the following diagnostic and problem determination materials and contact the IBM Support Center:

- Binary machine-readable dump data sets (e.g., SVC dump or SYSMDUMP data sets). (Print image formatted dumps written to data sets referenced by the SYSUDUMP and SYSABEND DD statements are, in general, inadequate for determining the source of the problem.)
- Formatted LOGREC records related to the failure. Refer to the *Environmental Record Editing and Printing Program (EREP) User's Guide* GC35-0151, and the *Environmental Record Editing and Printing Program (EREP) Reference* GC35-0152 for information on the selection and formatting of LOGREC records.
- Relevant sections of the MVS System Log.
- All related Job Logs and JCL listings.

Reason codes

<i>Table 24: Reason codes</i>		
Reason Code	Explanation	Detecting Module
0401	<p>The attempt to OPEN a DCB was not succesful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 points to an area consisting of the DD name, immediately followed by the DCB. • R14 contains the OPEN return code. 	ARSYSPIN
0402	<p>The attempt to create a capture Data Space was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the DSPSERV Reason Code. • R2 points to the DSPSERV interface parameter list. • R14 contains the DSPSERV Return Code. 	ARSYSPIN
0403	<p>The attempt to create an Access List Entry for the capture Data Space was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ALESERV Reason Code. • R2 points to the ALESERV interface parameter list. • R14 contains the ALESERV Return Code. 	ARSYSPIN
0404	<p>The attempt to delete the capture Data Space was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the DSPSERV Reason Code. • R2 points to the DSPSERV interface parameter list. • R14 contains the DSPSERV Return Code. 	ARSYSPIN
0405	<p>The attempt to delete the Access List Entry for the capture Data Space was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ALESERV Reason Code. • R2 points to the ALESERV interface parameter list. • R14 contains the ALESERV Return Code. 	ARSYSPIN
0470	<p>One or more support modules could not be LOAded.</p>	ARSYSPIN
0471	<p>An unexpected return code was received from ATTACH.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 points to the ATTACH interface parameter list. • R14 contains the ATTACH service Return Code. 	ARSYSPIN

Table 24: Reason codes (continued)

Reason Code	Explanation	Detecting Module
0472	<p>An unexpected return code was received from the MVS Lock Manager when attempting to obtain a lock.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 indicates which lock was being manipulated: <ul style="list-style-type: none"> – 1: Local – 2: CMS • R14 contains the Lock Manager Return Code. 	ARSYSPIN
0473	<p>An unexpected return code was received from the MVS Lock Manager when attempting to release a lock.</p> <p>At the time of ABEND registers are loaded as follows:R0 indicates which lock was being manipulated:</p> <ul style="list-style-type: none"> • R0 indicates which lock was being manipulated: <ul style="list-style-type: none"> – 1: Local – 2: CMS • R14 contains the Lock Manager Return Code. 	ARSYSPIN
0474	<p>The End of Task Exit was entered for a TCB that did not have a corresponding element on the ARSYSCAS_TCBQH subtask queue.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the ARS3QMP structure. • R2 contains the address of the TCB presented to the End of Task Exit routine. • R14 contains the ARS3QMGR service Return Code. 	ARSYSPIN
0476	<p>The attempt to establish an ESTAEX recovery environment was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ESTAEX Reason Code. • R14 contains the ESTAEX Return Code. 	ARSYSPIN
047A	<p>An unexpected return code was received from the ARS3QMGR service.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the ARS3QMGR structure. • R14 contains the ARS3QMGR service Return Code. 	ARSYSPIN

Table 24: Reason codes (continued)

Reason Code	Explanation	Detecting Module
047B	<p>An unexpected return code was received from the TCBTOKEN service.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the TCBTOKEN interface parameter list. • R14 contains the TCBTOKEN service Return Code. 	ARSYSPIN
047D	<p>An unexpected return code was received from an IEANTxx service.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 indicates which service was being requested: <ul style="list-style-type: none"> – 4: IEANTCR – 8: IEANTRT – C: IEANTDL • R14 contains the IEANTxx service Return Code. 	ARSYSPIN
047E	<p>A subtask did not terminate when requested to do so. The subtask is being forcibly terminated with CALLRTM.</p>	ARSYSPIN
047F	<p>During subsystem shutdown, not all subtasks could be terminated.</p>	ARSYSPIN
0480	<p>An IEALSQRY request was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the IEALSQRY interface parameter list. • R14 contains the IEALSQRY service Return Code. 	ARSYSPIN
0490	<p>A request has been made to interact with the JES3 subsystem. However, the current unit of work is not executing under the control of JES3.</p>	ARSYSSI
0491	<p>An unexpected return code was received from the MVS Subsystem Request Router.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the IEFSSREQ Return Code. • R14 contains the address of the related ARSYCBSI structure. 	ARSYSSI

Table 24: Reason codes (continued)

Reason Code	Explanation	Detecting Module
0492	<p>An unexpected return code was received from the MVS Lock Manager while attempting to obtain a lock.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 indicates which lock was being manipulated: <ul style="list-style-type: none"> – 1: Local – 2: CMS • R14 contains the Lock Manager Return Code. 	ARSYSSI
0493	<p>An unexpected return code was received from the MVS Lock Manager while attempting to release a lock.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 indicates which lock was being manipulated: <ul style="list-style-type: none"> – 1: Local – 2: CMS • R14 contains the Lock Manager Return Code. 	ARSYSSI
0494	<p>An unexpected return code was received from JES in response to a SAPI request.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the SSOBRETN value • R14 contains the address of the related ARSYCBSI structure. 	ARSYSSI
0496	<p>The attempt to establish an ESTAEX recovery environment was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ESTAEX Reason Code. • R14 contains the ESTAEX Return Code. 	ARSYSSI
049A	<p>An unexpected return code was received from the ARS3QMGR service.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the ARS3QMP structure. • R14 contains the ARS3QMGR service Return Code. 	ARSYSSI
049B	<p>An unexpected return code was received from the TCBTOKEN service.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the TCBTOKEN interface parameter list. • R14 contains the TCBTOKEN service Return Code. 	ARSYSSI

Table 24: Reason codes (continued)

Reason Code	Explanation	Detecting Module
049C	<p>An unexpected return code was received from the DSPSERV service while attempting to release data space pages.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the DSPSERV service Reason Code. • R2 contains the address of the DSPSERV interface parameter list. • R14 contains the DSPSERV service Return Code. 	ARSYSSI
049D	<p>An unexpected return code was received from ATTACH.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 points to the ATTACH interface parameter list. • R14 contains the ATTACH service Return Code. 	ARSYSSI
04A9	<p>A request to unallocate a transfer data set was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the contents of DYNALLOC Request Block field S99RSC. • R14 contains the address of the DYNALLOC Request Block (S99RB). 	ARSYSSI
04AA	<p>An attempt to OPEN a transfer file DCB was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the DCB address. 	ARSYSSI
04AB	<p>A request to DYNALLOC a transfer data set was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the contents of the DYNALLOC Request Block field S99RSC. • R14 contains the address of the DYNALLOC Request Block (S99RB). 	ARSYSSI
04AC	<p>A request to unallocate a Spool file was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the contents of the DYNALLOC Request Block field S99RSC. • R14 contains the address of the DYNALLOC Request Block (S99RB). 	ARSYSSI
04AD	<p>An attempt to CLOSE a Spool file ACB was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R14 contains the ACB address. 	ARSYSSI

Table 24: Reason codes (continued)

Reason Code	Explanation	Detecting Module
04AE	<p>An attempt to OPEN a Spool file ACB was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R14 contains the ACB address. 	ARSYSSI
04AF	<p>A request to dynamically allocate a Spool file was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the contents of DYNALLOC Request Block field S99RSC. • R14 contains the address of the DYNALLOC Request Block (S99RB). 	ARSYSSI
04B0	<p>An IEALSQRY request was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the address of the IEALSQRY interface parameter list. • R14 contains the IEALSQRY service Return Code. 	ARSYSSI
04B3	<p>The attempt to create an Access List Entry for the capture Data Space was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ALESERV Reason Code. • R2 points to the ALESERV interface parameter list. • R14 contains the ALESERV Return Code. 	ARSYSSI
04B5	<p>The attempt to delete the Access List Entry for the capture Data Space was unsuccessful.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ALESERV Reason Code. • R2 points to the ALESERV interface parameter list. • R14 contains the ALESERV Return Code. 	ARSYSSI
04B7	<p>The ARSYDSIE array has reached its size limit.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the ARSYDSIE array element count. 	ARSYSSI
04B8	<p>The maximum capture space size limit has been reached.</p> <p>At the time of ABEND registers are loaded as follows:</p> <ul style="list-style-type: none"> • R0 contains the size of the capture space in 4K units. • R14 contains a pointer to the last byte in the capture space that can be referenced. 	ARSYSSI

Table 24: Reason codes (continued)

Reason Code	Explanation	Detecting Module
04B9	The attempt to EXTEND the capture data space was unsuccessful. At the time of ABEND registers are loaded as follows: <ul style="list-style-type: none"> • R0 contains the DSPSERV Reason Code. • R2 points to the DSPSERV interface parameter list. • R14 contains the DSPSERV Return Code. 	ARSYSSI
04BA	The attempt to position the cursor over the Spool file currently being processed was unsuccessful. At the time of ABEND registers are loaded as follows: <ul style="list-style-type: none"> • R0 contains RPL field RPLFDBWD. • R2 contains the address of the RPL. • R14 contains the POINT Return Code. 	ARSYSSI
04BB	The attempt to obtain the file position RBA for the Spool file currently being process was unsuccessful. At the time of ABEND registers are loaded as follows: <ul style="list-style-type: none"> • R0 contains the SHOWCB Reason Code. • R2 contains the address of the RPL. • R14 contains the SHOWCB Return Code. 	ARSYSSI

ARSSPVIN: Sample APKACIF exit routine

Important: The ARSSPVIN exit is provided for compatibility with prior releases of Content Manager OnDemand. In most circumstances, the content of the separator records constructed by ARSYSPIN provides the necessary data items for the construction of job completion indexing values, and using ARSSPVIN is not necessary.

If you want, you can use ARSYSPIN exits to augment the data stream. In particular, the ARSYSPIN Input Exit (UX03) and Separator Exit (UX06) provide substantially more information about the job that produced the Spool file being processed than what is available at the time when APKACIF (or another indexer program) is driven by Content Manager OnDemand. In addition, the processing overhead of driving ARSYSPIN exit routines is lower than that associated with indexer exit routines. For this reason, IBM recommends that you do not use ARSSPVIN.

ARSSPVIN is a sample APKACIF input exit that is provided with ARSYSPIN to introduce additional index values into the data stream, using a "trailer" record. Trailer records are inserted at the end of the JESMSG LG data, and reflect the highest severity condition (this may be a step completion code, an ABEND code, or other type of problem such as a JCL error) that has been observed in messages contained within these Spool files.

This exit depends on the separator records constructed by ARSYSPIN for identification of the appropriate Spool file (e.g., JESMSG LG or JSYSMSG). When either of these Spool files are being processed, each record is examined to look for specific message identifiers or other specific character strings to select records from which condition code or ABEND code information is extracted. On reaching the end of the Spool file data, the trailer record is inserted into the data stream.

The following table provides the structure of the record produced by ARSSPVIN. All offsets and lengths are shown in hexadecimal, unless otherwise stated.

Table 25: Trailer record format for the ARSSPVIN sample APKACIF input exit

Offset	Length	Value	Type	Description
+0	1	C"	Character	An ASA Space One Line and Write control character.
+1	12	<ul style="list-style-type: none"> • 4CC1D9 • E2E8E2 • D7C9D5 • 40E399 • 818993 • 85996E 	Character	In EBCDIC Code Pages 037 and 1047 this is the character string <ARSYSPIN Trailer>
+13	4		Character	The Condition Code value.
+17	1	C"	Character	A Blank.
+18	10		Character	The Job start timestamp, in the format yyyy-mm-dd hh.mm.
+28	1	C"	Character	A Blank
+29	10		Character	The Job end timestamp, in the format yyyy-mm-dd hh.mm.

Special considerations for APKACIF exits written in COBOL

The ARSSPVIN sample APKACIF input exit is written as a COBOL main program. In order to prevent the Language Environment from creating and destroying the COBOL runtime environment each time ARSSPVIN is called, a CEEUOPT CSECT must be assembled and link-edited with the COBOL object code. Constructing a CEEUOPT CSECT is documented in *z/OS Language Environment Customization SA22-7564*. A sample CEEUOPT CSECT is included in data set CEE.SCEESAMP(CEEUOPT). You can use this sample as a model, but you must be sure that the following option is specified:

```
RTEREUS=(ON)
```

IBM also recommends that the ALL31(ON) option be specified, but this is not required.

In addition, you must be sure that the resulting module is link-edited as **NOT RE-ENTRANT** and **NOT REUSABLE**. This is required to allow the local variables in the COBOL exit code to retain their values across multiple invocations. The following is a sample CEEUOPT CSECT.

```
CEEUOPT CSECT ,
CEEUOPT AMODE ANY
CEEUOPT RMODE ANY

CEEEOPT
  ABPERC=(NONE) , +
  ABTERMENC=(ABEND) , +
  AIXBLD=(OFF) , +
  ALL31=(ON) , +
  ANYHEAP=(16K, 8K, ANYWHERE, FREE) +
  BELOWHEAP=(8K, 4K, FREE) , +
  CBLOPTS=(ON) , +
  CBLPSHPOP=(ON) , +
  CBLQDA=(OFF) , +
  CEEDUMP=(60, SYSOUT=*, FREE=END, SPIN=UNALLOC) , +
  CHECK=(ON) , +
  COUNTRY=(US) , +
  DEBUG=(OFF) , +
  DEPTHCONDLMT=(10) , +
```

```

DYNDUMP=(*USERID,NODYNAMIC,TDUMP),      +
ENVAR=(' '),                              +
ERRCOUNT=(0),                            +
ERRUNIT=(6),                              +
FILEHIST=(ON),                            +
FILETAG=(NOATOCVT,NOAUTOTAG),            +
HEAP=(32K,32K,ANYWHERE,KEEP,8K,4K),      +
HEAPCHK=(OFF,1,0,0,0),                   +
HEAPOOLS=(OFF,8,10,32,10,128,10,156,10,1024,10,2048, +
10,0,10,0,10,0,10,0,10,0,10,0,10),      +
INFOMSGFILTER=(OFF,,,,),                +
INQPCOPN=(ON),                           +
INTERRUPT=(OFF),                         +
LIBSTACK=(4K,4K,FREE),                   +
MSGFILE=(SYSOUT,FBA,121,0,NOENQ),        +
MSGQ=(15),                                +
NATLANG=(ENU),                           +
NOAUTOTASK=,                             +
NOTEST=(ALL,* ,PROMPT,INSPREF),          +
NOUSRHDLR=(' '),                         +
OCSTATUS=(ON),                          +
PC=(OFF),                                 +
PLITASKCOUNT=(20),                      +
POSIX=(OFF),                             +
PROFILE=(OFF,' '),                       +
PRTUNIT=(6),                             +
PUNUNIT=(7),                             +
RDRUNIT=(5),                             +
RECPAD=(OFF),                            +
RPTOPTS=(OFF),                           +
RPTSTG=(OFF),                            +
RTEREUS=(ON),                            +
SIMVRD=(OFF),                            +
STACK=(128K,128K,ANYWHERE,KEEP,512K,128K), +
STORAGE=(NONE,NONE,NONE,OK),            +
TERMTHDACT=(TRACE,,96),                 +
THREADHEAP=(4K,4K,ANYWHERE,KEEP),       +
THREADSTACK=(OFF,4K,4K,ANYWHERE,KEEP,128K,128K), +
TRACE=(OFF,4KDUMP,LE=0),                +
TRAP=(ON,SPIE),                          +
UPSI=(00000000),                        +
VCTRSAVE=(OFF),                          +
XPLINK=(OFF),                             +
XUFLOW=(AUTO)                            +

```

END ,

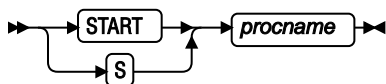
Chapter 35. Operator commands

You can use z/OS operator commands to monitor and control the operations of the ARSSOCKD server. The following sections provide a reference to commonly used commands for experienced system programmers.

START command

Use the START command to dynamically start an OnDemand server. The abbreviated version of the command is S.

Syntax



Parameters

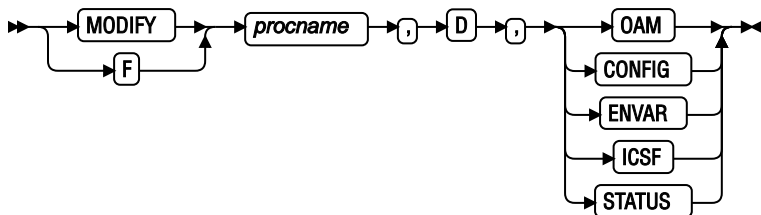
procname

The name of a member in a cataloged procedure library that runs the ARSSOCKD or ARSOBJD program.

MODIFY command

Use the MODIFY command to display information about the OnDemand server. The abbreviated version of the command is F.

Syntax



Parameters

procname

The member name of the cataloged procedure that is used to start the OnDemand server.

D,OAM

Display the status of the OAM threads.

D,CONFIG

Display the configuration of the server instance.

D,ENVAR

Display the environment variables for the server instance.

D,ICSF

Display whether you can encrypt data with the 128-bit AES cipher.

Examples (D,OAM)

The following example shows the output that is returned when you run the MODIFY D,OAM command to display the status of the OAM threads:

```
F ARSSOCKD,D,OAM

ARS0375I 13.21.14 DISPLAY OAM
DB2      DB1A      PLAN CBRIDBS      GENERAL      8 SLOW      1
TCB@     STATUS  COLLECTION/OBJECT
007CE6B0 IGDE     ARS.COLLECT
                JJA.L57.FAA*
                START=13.16.59.963 STOP=13.17.00.021
007CE408 IGST     ARS.COLLECT
                JJA.L57.FAA1
                START=13.16.39.001 STOP=13.16.39.009

007CE160 IG
007CAE88 IG
007CABE0 IG
007CA938 IG
007CA690 IG
007CA3E8 IG
007CA140 IS
```

Examples (D,CONFIG)

The following example shows the output that is returned when you run the MODIFY D,CONFIG command to display the configuration of the server instance.

Note: In the following example, the <CFG> are the operative lines of the ars.cfg from the SRVR_OD_CFG file. The section started by the <CACHE> are the cache files from the SRVR_SM_CFG file.

```
F ARSSOCKD,D,CONFIG

ARS0391I 3.21.14 DISPLAY CONFIG
<@SRV@_ARCHIV9A>
PROTOCOL=2
PORT=1456
SRVR_INSTANCE=ARSDBA9A
SRVR_INSTANCE_OWNER=ARSUSE9A
SRVR_OD_CFG=/etc/ars/ars9a.cfg
SRVR_SM_CFG=/etc/ars/ars9a.cache
HOST=SVLSPIZ4
<CFG>
ARS_LANGUAGE=ENU
ARS_NUM_DBSRVR=4
ARS_TMP=/ars/ars9a/tmp
ARS_PRINT_PATH=/ars/ars9a/tmp
DB_ENGINE=DB2
ARS_DB_TABLESPACE=ARSTSPAC
ARS_NUM_OAMSRVR=4
ARS_OAM_DB1A=DB1A
ARS_OAM_PLAN=CBRIDBS
ARS_STORAGE_MANAGER=TSM
LIBPATH=/etc/ars/tsm2
DSMI_CONFIG=/etc/ars/tsm2/dsm.opt
DSMI_DIR=/etc/ars/tsm2
DSMI_LOG=/ars/ars9a/tmp
ARS_ORIGINAL_CODEPAGE=0037
<CACHE>
SMFS1=/ars/ars9a/cache
SMFS_NUM=1
```


Examples (D,ENVAR)

The following example shows the output that is returned when you run the MODIFY D,ENVAR command to display the environment variables for the server instance.

```
F ARSSOCKD,D,ENVAR

ARS0392I 3.21.14 DISPLAY ENVAR
_XEE_HEAP_MANAGER=CEL4MCHK
ARS_ORIGINAL_CODEPAGE=0037
ARS_LANGUAGE=ENU
ARS_NUM_DBSRVR=4
ARS_TMP=/ars/ars9a/tmp
ARS_PRINT_PATH=/ars/ars9a/tmp
DB_ENGINE=DB2
ARS_DB_TABLESPACE=ARSTSPAC
ARS_NUM_OAMSRVR=4
ARS_OAM_DB1A=DB1A
ARS_OAM_PLAN=CBRIDBS
ARS_STORAGE_MANAGER=TSM
LIBPATH=/etc/ars/tsm2
DSMI_CONFIG=/etc/ars/tsm2/dsm.opt
DSMI_DIR=/etc/ars/tsm2
DSMI_LOG=/ars/ars9a/tmp
_EDC_ADD_ERRN02=1
```

Example (D,ICSF)

The following example shows the output that is returned when you run the MODIFY D,ICSF command to display whether you can encrypt data with the 128-bit AES cipher.

```
F ARSSOCKD,D,ICSF

ARS0438I 14.11.51 DISPLAY ICSF
CSFIQF RC=0, RSN=00000000, AES=1, FMID=HCR7780
```

RACF access authority and resource names

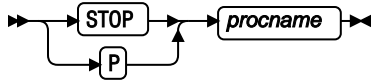
OnDemand verifies the authority of the operator to issue the command by issuing the CMDAUTH macro. When an operator issues an OnDemand command with a RACF profile, OnDemand determines the resource-name that matches the command and passes that resource-name to RACF or other external security products through the CMDAUTH macro. RACF uses the resource-name to locate the profile for the command in the OPERCMDS class and verifies that the operator is allowed to issue the command by checking the access list in the profile. If RACF authorizes the access, OnDemand processes the command; if RACF denies the access, OnDemand rejects the command. For detailed information on setting up the OPERCMDS CLASS, see the Administering the Use of Operator Commands section in the *Security Server RACF Security Administrator's Guide*.

Command/keyword	Authority	Resource-Name
DISPLAY,CONFIG	READ	ARS.DISPLAY.CONFIG
DISPLAY,ENVAR	READ	ARS.DISPLAY.ENVAR
DISPLAY,ICSF	READ	ARS.DISPLAY.ICSF
DISPLAY,OAM	READ	ARS.DISPLAY.OAM
DISPLAY,STATUS	READ	ARS.DISPLAY.STATUS

STOP command

Use the STOP command to stop start an OnDemand server that is running. The abbreviated version of the command is P.

Syntax



Parameters

procname

The name of the procedure to stop. This name is the same name that was used to start the server by using the START command.

Appendix A. Appendixes to the Administration Guide

Accessibility features

This product includes a number of features that make it more accessible for people with disabilities. These features include:

- The ability to operate all features using the keyboard instead of the mouse.
- Support for enhanced display properties
- Options for audio and visual alert cues
- Compatibility with assistive technologies
- Compatibility with operating system accessibility features
- Accessible documentation formats

Keyboard input and navigation

Keyboard input

The Content Manager OnDemand clients can be operated using only the keyboard. Menu items and controls provide access keys that allow users to activate a control or select a menu item directly from the keyboard. These keys are self-documenting, in that the access keys are underlined on the control or menu where they appear.

Keyboard focus

In Windows-based systems, the position of the keyboard focus is highlighted, indicating which area of the window is active and where the user's keystrokes will have an effect.

Features for accessible display

The clients have a number of features that enhance the user interface and improve accessibility for users with low vision. These enhancements include support for high-contrast settings and customizable font properties.

High-contrast mode

The clients support the high-contrast-mode option that is provided by the operating system. This feature assists users who require a higher degree of contrast between background and foreground colors.

Font settings

In Windows-based systems, you can specify display settings that determine the color, size, and font for the text in menus and dialog windows. The client allows you to select the font for the document list.

Non-dependence on color

You do not need to distinguish between colors in order to use any function of this product.

Alternative alert cues

In Windows-based systems, the SoundSentry feature can be used to provide visual feedback for general application and system alerts such as warning beeps.

Compatibility with assistive technologies

The clients are compatible with screen reader applications such as Narrator and Via Voice. The clients have properties required for these accessibility applications to make onscreen information available to visually impaired users.

Accessible documentation

Documentation for the Content Manager OnDemand product is available in HTML format. This allows users to view documentation according to the display preferences set in their browsers. It also allows the use of screen readers and other assistive technologies.

ARSSUPPORT utility

You can use ARSSUPPORT, a Java based tool to gather diagnostic information such as log entries. This tool is especially helpful when you need to report problems to IBM service.

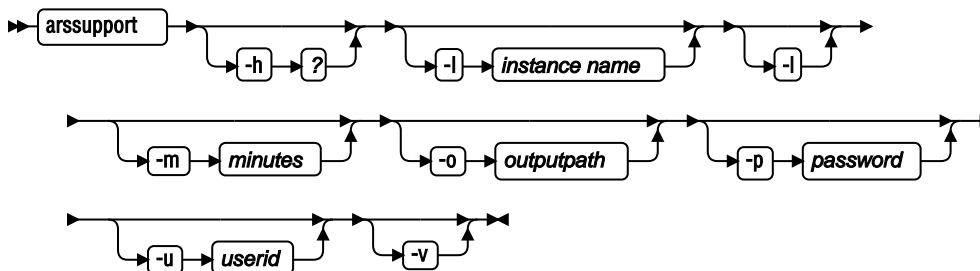
The ARSSUPPORT utility is delivered in the `arssupport.jar` file. To invoke the utility, use this command:

```
java -jar arssupport.jar
```

Prerequisites

- Ensure that you have Java runtime version 1.5 or higher to run this program.
- Ensure that you are logged on to the operating system using an ID that has the root authority on UNIX System Services.
- Run ARSSUPPORT from the Unix System Services command prompt.
- To retrieve system log entries, ensure that the Content Manager OnDemand server is running.
- The data are collected from the computer where ARSSUPPORT is run.

Syntax



Parameters

-h

Use this parameter to display help and usage information about this tool.

-I instance name

Specify the instance name to collect the instance information. If you do not specify this option, ARCHIVE is used as the instance name.

-l

Specify this parameter to retrieve system log entries. If you do not specify this option, the log entries for the past 60 minutes are retrieved. Ensure that you use this parameter with the -u parameter.

-m minutes

Specify how many past minutes of the system log entries to retrieve from the server. The maximum is 6000 minutes.

-o outputpath

Specify the output directory name. If the output directory is not specified, the output directory is the current directory.

-p password

Password that you use to access the server.

-u userid

The user ID that you use to access the server. If the -l parameter is specified, the -u parameter is required.

-v

Verbose output while running.

ARSSUPPORT generates information about a Content Manager OnDemand server including information about its configuration and system environment. ARSSUPPORT archives all files into one compressed file, arssupport.zip, and places this file in the odsupport subdirectory of the output directory.

Examples

```
java -jar arssupport.jar -h
java -cp ./arssupport.jar com/ibm/cm/od/arssupport -o c:\pmr11256 -l -u admin
java -cp ./arssupport.jar com/ibm/cm/od/arssupport -o . -I instance -l -m 28 -u admin
```

System control tables

This section contains a description of each Content Manager OnDemand system table, including column names and data types. This section also contains a description of the application group data tables (also known as dynamic tables).

The system tables are created when the Content Manager OnDemand database is created by using the ARSDB program. The system tables are updated during normal operation in response to dynamic SQL statements.

Warning: The Content Manager OnDemand system tables should not be modified or queried directly. Doing so may effect the operation or performance of the system. The system tables are subject to change in future versions of Content Manager OnDemand.

List of tables

- [“Application group table \(ARSAG\)” on page 288](#)
- [“Application group to folder field mapping table \(ARSAG2FOL\)” on page 289](#)
- [“Application group field table \(ARSAGFLD\)” on page 290](#)
- [“Application group field alias table \(ARSAGFLDALIAS\)” on page 291](#)
- [“Application group composite index table \(ARSAGINDEX\)” on page 291](#)
- [“Application group permissions table \(ARSAGPERMS\)” on page 292](#)
- [“Annotation table \(ARSANN\)” on page 292](#)
- [“Application table \(ARSAPP\)” on page 293](#)
- [“User logical views table \(ARSAPPUSR\)” on page 294](#)
- [“Cabinet table \(ARSCAB\)” on page 295](#)
- [“Cabinet to folder mapping table \(ARSCAB2FOL\)” on page 295](#)
- [“Cabinet permissions table \(ARSCABPERMS\)” on page 296](#)
- [“CFS-CMOD work table \(ARSCFSODWORK\)” on page 296](#)
- [“Folder table \(ARSFOL\)” on page 297](#)
- [“Folder field table \(ARSFOLFLD\)” on page 297](#)
- [“Folder user fields table \(ARSFOLFLDUSR\)” on page 298](#)
- [“Folder permissions table \(ARSFOLPERMS\)” on page 299](#)
- [“FTI work table \(ARSFTIWORK\)” on page 299](#)
- [“Group table \(ARSGROUP\)” on page 300](#)
- [“Hold table \(ARSHOLD\)” on page 301](#)
- [“Hold mapping table \(ARSHOLDMAP\)” on page 301](#)
- [“Hold permissions table \(ARSHOLDPERMS\)” on page 302](#)

- [“Hold work table \(ARSHOLDWORK\)” on page 302](#)
- [“Load table \(ARSLOAD\)” on page 303](#)
- [“Named query table \(ARSNAMEQ\)” on page 303](#)
- [“Node table \(ARSNODE\)” on page 304](#)
- [“Printer table \(ARSPRT\)” on page 312](#)
- [“Printer options table \(ARSPRTOPTS\)” on page 312](#)
- [“Printer user table \(ARSPRTUSR\)” on page 313](#)
- [“Resources table \(ARSRES\)” on page 313](#)
- [“Segment table \(ARSSEG\)” on page 314](#)
- [“Storage set table \(ARSSET\)” on page 315](#)
- [“System parameters table \(ARSSYS\)” on page 316](#)
- [“User table \(ARSUSER\)” on page 317](#)
- [“Users in groups table \(ARSUSRGRP\)” on page 319](#)
- [“User group ID table \(ARSUSRGRPID\)” on page 320](#)
- [“Application group data table” on page 320](#)

Application group table (ARSAG)

This table maintains a catalog of application groups.

Table 27: Application group table (ARSAG)

Column Name	Data Type	Description
name	VARCHAR(60)	Application group name
description	VARCHAR(120)	Application group description
agid	INTEGER	Application group identifier
agid_name	VARCHAR(8)	Application group internal identifier
db_seg	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Segment type (Database Organization)
ann_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Annotation update
db_exp_date	INTEGER	Expire database indexes
db_mgrt_date	INTEGER	Migrate database indexes
sm_cache_doc	INTEGER	Storage manager cache definition
resgrp	INTEGER	Resource group identifier
load_id	INTEGER	Load Id counter
log	INTEGER	Log mask
load_id_suffix	VARCHAR(8)	Load Id suffix
mgrt_cache_date	INTEGER	Copy cache to archive storage
reimport_exp_date	INTEGER	Number of days to keep imported table before re-expiring
sid	INTEGER	Storage set identifier

Table 27: Application group table (ARSAG) (continued)

Column Name	Data Type	Description
expire_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Expiration type
seg_rows	INTEGER	Rows per segment (Maximum Rows)
seg_id	INTEGER	Segment Id counter
obj_size	INTEGER	Size of storage object
type	INTEGER	Type of application group (Data Management; Create Tablespace Type)
upd_userid	VARCHAR(128)	Last user to update the application group
upd_date	BIGINT	Date and time application group was last updated
last_doc_date	BIGINT	Migrate last document date
migr_srvr_str	VARCHAR(254)	Migrate server string
sm_cache_delta	INTEGER	Delta value of the expiration (number of days) of cached data
database_name	VARCHAR(8)	Internal database alias name
sm_cache_res	INTEGER	Storage manager resources cache definition
fti_server	VARCHAR	Full text index server
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time application group was last updated
last_doc_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Migrate last document date
last_load_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time application group was last loaded
last_query_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time application group was last queried
last_retr_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time application group was last retrieved

Table 28: Indexes for application group table (ARSAG)

Index name	Columns in index	Type of index
arsag_name_idx	name	Unique index

Application group to folder field mapping table (ARSAG2FOL)

This table maintains a catalog of mappings between application group fields and folder fields. This table also contains a catalog of applications within application groups that can be queried from a folder.

Table 29: Application group to folder field mapping table (ARSAG2FOL)

Column Name	Data Type	Description
fid	INTEGER	Folder identifier
agid	INTEGER	Application group identifier
aid	INTEGER	Application within the application group
folder_field	SMALLINT	Folder field identifier
appgrp_field1	SMALLINT	Application group field identifier
appgrp_field2	SMALLINT	Application group field identifier

Table 30: Indexes for application group to folder field mapping table (ARSAG2FOL)

Index name	Columns in index	Type of index
arsag2fol_idx	fid, agid, aid, folder_field	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Application group field table (ARSAGFLD)

This table maintains a catalog of application group fields.

Table 31: Application group field table (ARSAGFLD)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
field	SMALLINT	Application group field identifier
name	VARCHAR(18)	Application group field name
type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Application group field type
qual	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Application group field type qualifier
mask	INTEGER	Application group field mask
string_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Type of string field
string_len	SMALLINT	Length of string field
string_case	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Case of string field
leading	VARCHAR(8)	Leading characters to trim from string field
strip	VARCHAR(8)	Any characters to trim from string field

Table 31: Application group field table (ARSAGFLD) (continued)

Column Name	Data Type	Description
trailing	VARCHAR(8)	Trailing characters to trim from string field
map_field	SMALLINT	Map field

Table 32: Indexes for application group field table (ARSAGFLD)

Index name	Columns in index	Type of index
arsagfld_idx	agid, field	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Application group field alias table (ARSAGFLDALIAS)

This table maintains a catalog of application group field aliases (Database values and Displayed values).

Table 33: Application group field alias table (ARSAGFLDALIAS)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
field	SMALLINT	Application group field identifier
internal	VARCHAR(254)	Internal (Database) value
external	VARCHAR(254)	External (Displayed) value

Table 34: Indexes for application group field alias table (ARSAGFLDALIAS)

Index name	Columns in index	Type of index
arsagfldalias_idx	agid, field	Cluster index for DB2; Index for Oracle and SQL Server

Application group composite index table (ARSAGINDEX)

This table maintains application group composite indexes on application group fields.

Table 35: Application group composite index table (ARSAGINDEX)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
iid	SMALLINT	Index identifier
type	INTEGER	Application group identifier
sort_order	SMALLINT	Specifies the order of the application group fields
ag_fld	SMALLINT	Application group field identifier
sort_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Specifies that index entries should be kept in ascending or descending order of the column values

<i>Table 36: Indexes for application group composite index table (ARSAGINDEX)</i>		
Index name	Columns in index	Type of index
arsagindex_idx	agid	Cluster index for DB2; Index for Oracle and SQL Server

Application group permissions table (ARSAGPERMS)

This table maintains permissions specifications for application groups. Each row contains two primary elements: who (id) and what (agid) and specifies the user's authority for the application group, the user's document and annotation permissions, and a query restriction, if any.

Table 37: Application group permissions table (ARSAGPERMS)

Column Name	Data Type	Description
id	INTEGER	User identifier
agid	INTEGER	Application group identifier
id_perms	SMALLINT	User authority for the application group
doc_perms	SMALLINT	Document permissions
ann_perms	SMALLINT	Annotation permissions
q_restrict	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Query restriction; user-defined and validated SQL string

<i>Table 38: Indexes for application group permissions table (ARSAGPERMS)</i>		
Index name	Columns in index	Type of index
arsagperms_idx	agid, id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Annotation table (ARSANN)

This table maintains a catalog of annotations.

Table 39: Annotation table (ARSANN)

Column Name	Data Type	Description
id	INTEGER	User identifier
userid	VARCHAR(128)	User name
agid	INTEGER	Application group identifier
doc_name	VARCHAR(11)	Document name
doc_off	INTEGER	Document offset
doc_len	INTEGER	Document length
comp_off	INTEGER	Compression offset
comp_len	INTEGER	Compression length
time_stamp	BIGINT	Time annotation was created

Table 39: Annotation table (ARSANN) (continued)

Column Name	Data Type	Description
page	INTEGER	Page (within document) on which annotation exists
type	INTEGER	Type of annotation
x_off	INTEGER	x offset on page
y_off	INTEGER	y offset on page
reserved	VARCHAR(32) FOR BIT DATA	(first byte is annotation color)
text_buf	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	The text of the annotation
ann_rowid	INTEGER	Annotation identifier
image_buf	BLOB(1048576) (DB2), VARBINARY(MAX) (SQL Server), BLOB (Oracle)	Any images that are attached to the annotation
table_name	VARCHAR(18)	Segment in which annotation exists
doc_exp_date	BIGINT	Expiration date of document
doc_exp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Expiration date of document
time_stamp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time annotation table was created or last modified

Table 40: Indexes for annotation table (ARSANN)

Index name	Columns in index	Type of index
arsann_idx	agid, id, doc_name, doc_off, doc_len, comp_off, comp_len	Cluster index for DB2; Index for Oracle and SQL Server

Application table (ARSAPP)

This table maintains a catalog of applications.

Table 41: Application table (ARSAPP)

Column Name	Data Type	Description
name	VARCHAR(60)	Application name
description	VARCHAR(120)	Application description
agid	INTEGER	Application group identifier
aid	INTEGER	Application identifier
doc_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Document type
doc_comp_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Default document compression type

Table 41: Application table (ARSAPP) (continued)

Column Name	Data Type	Description
res_comp_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Default resource compression type
idx_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Indexer
comp_obj_size	INTEGER	Compressed object size
pid	INTEGER	Default printer identifier
alias	VARCHAR(254)	Application identifier within application group
fixed_view	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Default and fixed logical view information
indexer	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Indexing parameters
preprocessor	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Parameters used to preprocess rows before loading
upd_userid	VARCHAR(128)	Userid of last user to update application
upd_date	BIGINT	Date and time application was last updated (in old style)
db_exp_date	INTEGER	Expiration date
appl_type	INTEGER	Application type
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time application was last updated

Table 42: Indexes for application table (ARSAPP)

Index name	Columns in index	Type of index
arsapp_idx	agid, aid, name	Unique cluster index for DB2; Unique index for Oracle and SQL Server

User logical views table (ARSAPPUSR)

This table maintains a catalog of logical views.

Table 43: User logical views table (ARSAPPUSR)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
aid	INTEGER	Application identifier
id	INTEGER	User identifier

Table 43: User logical views table (ARSAPPUSR) (continued)

Column Name	Data Type	Description
user_view	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Logical view information

Table 44: Indexes for user logical views table (ARSAPPUSR)

Index name	Columns in index	Type of index
arsappusr_idx	agid, aid, id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Cabinet table (ARSCAB)

This table maintains a catalog of cabinet definitions.

Table 45: Cabinet table (ARSCAB)

Column Name	Data Type	Description
name	VARCHAR(60)	Cabinet name
description	VARCHAR(120)	Cabinet description
cid	INTEGER	Cabinet identifier
upd_userid	VARCHAR(128)	Userid of last user to update the cabinet
upd_date	BIGINT (DB2, SQL Server), INT (Oracle)	Date and time cabinet was last updated (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time cabinet was last updated

Table 46: Indexes for cabinet table (ARSCAB)

Index name	Columns in index	Type of index
arscab_name_idx	name	Unique index
arscab_cid_idx	cid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Cabinet to folder mapping table (ARSCAB2FOL)

This table maintains a catalog of mappings between cabinets and folders.

Table 47: Cabinet to folder mapping table (ARSCAB2FOL)

Column Name	Data Type	Description
cid	INTEGER	Cabinet identifier
fid	INTEGER	Folder identifier

<i>Table 48: Indexes for cabinet to folder mapping table (ARSCAB2FOL)</i>		
Index name	Columns in index	Type of index
arscab2fol_idx	cid, fid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Cabinet permissions table (ARSCABPERMS)

This table maintains permissions specifications for cabinets.

Table 49: Cabinet permissions table (ARSCABPERMS)

Column Name	Data Type	Description
id	INTEGER	User identifier
cid	INTEGER	Cabinet identifier
perms	INTEGER	User permissions for the cabinet

<i>Table 50: Indexes for cabinet permissions table (ARSCABPERMS)</i>		
Index name	Columns in index	Type of index
arscabperms_idx	cid, id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

CFS-CMOD work table (ARSCFSODWORK)

This table is a catalog of work between Content Manager OnDemand and CFS-CMOD.

Table 51: CFS-CMOD work table

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
name	VARCHAR(11)	Load identifier
type	INTEGER	Type of work
status	INTEGER	Status of work
time_stamp	BIGINT (DB2, SQL Server), INT (Oracle)	Timestamp of work (in old style)
misc_comp_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Compression type of misc
misc_info	BLOB(10485760) (DB2), VARBINARY(MAX) (SQL Server), BLOB (Oracle)	Work information
time_stamp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Timestamp of work

Table 52: Indexes for CFS-CMOD work table

Index name	Columns in index	Type of index
arscfsodwork_idx	agid, name	Cluster index for DB2; Index for Oracle and SQL Server

Folder table (ARSFOL)

This table maintains a catalog of folders.

Table 53: Folder table (ARSFOL)

Column Name	Data Type	Description
name	VARCHAR(60)	Folder name
description	VARCHAR(120)	Folder description
fid	INTEGER	Folder identifier
ann_search	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Annotation display
upd_userid	VARCHAR(128)	Userid of last user to update folder
upd_date	BIGINT	Date and time folder was last updated (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time folder was last updated

Table 54: Indexes for folder table (ARSFOL)

Index name	Columns in index	Type of index
arsfol_name_idx	name	Unique index
arsfol_fid_idx	fid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Folder field table (ARSFOLFLD)

This table maintains a catalog of folder fields.

Table 55: Folder field table (ARSFOLFLD)

Column Name	Data Type	Description
fid	INTEGER	Folder identifier
field	SMALLINT	Folder field identifier
name	VARCHAR(60)	Folder field name
description	VARCHAR(120)	Folder field description
type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Folder field type

Table 55: Folder field table (ARSFOLFLD) (continued)

Column Name	Data Type	Description
qual	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Folder field type qualifier
map_type	SMALLINT	Type of field mapping

Table 56: Indexes for folder field table (ARSFOLFLD)

Index name	Columns in index	Type of index
arsfolfid_idx	fid, field	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Folder user fields table (ARSFOLFLDUSR)

This table maintains a catalog of folder fields for users that have specific field information for a folder.

Table 57: Folder user fields table (ARSFOLFLDUSR)

Column Name	Data Type	Description
id	INTEGER	User identifier
fid	INTEGER	Folder identifier
field	SMALLINT	Folder field identifier
query_order	SMALLINT	Order of fields in search criteria
dspl_order	SMALLINT	Order of fields in document list
sort_order	SMALLINT	Default sort order
sort_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Type of sort
usr_mask	SMALLINT	Field setup
ops	SMALLINT	Search operators
def_op	SMALLINT	Default search operator
def_value1	VARCHAR(254)	Default search value
def_value2	VARCHAR(254)	Default search value for Between, Not Between
min_int	BIGINT	Smallest integer allowed
max_int	BIGINT	Largest integer allowed
datefmt	VARCHAR(60)	Format of date strings in search criteria
datedef_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Default date units for interval
datedef_arg	INTEGER	Default date value for interval
min_float	FLOAT	Smallest decimal allowed

Table 57: Folder user fields table (ARSFOLFLDUSR) (continued)

Column Name	Data Type	Description
max_float	FLOAT	Largest decimal allowed
decimal_pt_pos	SMALLINT	Precision (right of decimal point)
string_wild	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Wildcard
dspl_datefmt	VARCHAR(60)	Format of date strings in document list
min_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Minimum date allowed
max_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Maximum date allowed
min_str	VARCHAR(254)	Minimum decimal float allowed
max_str	VARCHAR(254)	Maximum decimal float allowed

Table 58: Indexes for folder user fields table (ARSFOLFLDUSR)

Index name	Columns in index	Type of index
arsfolfldusr_idx	fid, id, field	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Folder permissions table (ARSFOLPERMS)

This table maintains permissions specifications for folders. Each row contains two primary elements: the user (*id*) and the folder (*fid*) and specifies the user's authority for the folder, fields, and named queries.

Table 59: Folder permissions table (ARSFOLPERMS)

Column Name	Data Type	Description
id	INTEGER	User identifier
fid	INTEGER	Folder identifier
perms	SMALLINT	User permissions for the folder
hits	INTEGER	Maximum hits returned from a query

Table 60: Indexes for folder permissions table (ARSFOLPERMS)

Index name	Columns in index	Type of index
arsfolperms_idx	fid, id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

FTI work table (ARSFTIWORK)

This table maintains a catalog of FTI work.

Table 61: FTI work table (ARSFTIWORK)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
name	VARCHAR(11)	Load name
type	INTEGER	Work type
status	INTEGER	Work status
time_stamp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Timestamp of work
misc_comp_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Compression type of misc
misc_info	BLOB(10485760) (DB2), VARBINARY(MAX) (SQL Server), BLOB (Oracle)	Work information

Table 62: Indexes for FTI work table (ARSFTIWORK)

Index name	Columns in index	Type of index
arsftiwork_idx	agid, name	Index

Group table (ARSGROUP)

This table maintains a catalog of groups.

Table 63: Group table (ARSGROUP)

Column Name	Data Type	Description
gid	INTEGER	Group identifier
id	INTEGER	Owner of the group; can be a user or a group
name	VARCHAR(128)	Name of group
description	VARCHAR(120)	Group description
upd_userid	VARCHAR(128)	Userid of last user to update group
upd_date	BIGINT	Date and time group was last updated (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time group was last updated

Table 64: Indexes for group table (ARSGROUP)

Index name	Columns in index	Type of index
arsgroup_gid_idx	gid	Unique cluster index for DB2; Unique index for Oracle and SQL Server
arsgroup_name_idx	name	Unique index

Hold table (ARSHOLD)

This table is a catalog of holds.

Table 65: ARSHOLD table

Column Name	Data Type	Description
name	VARCHAR(60)	Name of hold
description	VARCHAR(120)	Description of hold
hid	INTEGER	Hold identifier
upd_userid	VARCHAR(128)	User ID that created the hold
upd_date	BIGINT (DB2, SQL Server), INT (Oracle)	Timestamp when the hold was created (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Timestamp when the hold was created

Table 66: Indexes for ARSHOLD table

Index name	Columns in index	Type of index
arshold_name_idx	name	Unique index
arshold_hid_idx	hid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Hold mapping table (ARSHOLDMAP)

This table maintains a catalog of documents to holds.

Table 67: Hold mapping table (ARSHOLDMAP)

Column Name	Data Type	Description
hid	INTEGER	Hold identifier
agid	INTEGER	Application group identifier
aid	INTEGER	Application identifier
hold_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Hold type
name	VARCHAR(11)	Document object name
doc_off	INTEGER	Document offset
doc_len	INTEGER	Document length
comp_off	INTEGER	Document compression offset
comp_len	INTEGER	Document compression length
res_id	INTEGER	Document resource identifier

<i>Table 68: Indexes for hold mapping table (ARSHOLDMAP)</i>		
Index name	Columns in index	Type of index
arsholdmap_idx	hid, agid, aid, name, doc_off, doc_len, comp_off, comp_len	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Hold permissions table (ARSHOLDPERMS)

This table maintains a catalog of hold permissions.

Table 69: Hold permissions table (ARSHOLDPERMS)

Column Name	Data Type	Description
id	INTEGER	User/Group identifier
hid	INTEGER	Hold identifier
perms	INTEGER	Permission

<i>Table 70: Indexes for hold permissions table (ARSHOLDPERMS)</i>		
Index name	Columns in index	Type of index
arsholdperms_idx	hid, id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Hold work table (ARSHOLDWORK)

This table maintains a catalog of hold work.

Table 71: Hold work table (ARSHOLDWORK)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
aid	INTEGER	Application identifier
name	VARCHAR(11)	Load name
type	INTEGER	Work type
status	INTEGER	Work status
time_stamp	BIGINT (DB2, SQL Server), INT (Oracle)	Timestamp of work (in old style)
time_stamp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Timestamp of work

<i>Table 72: Indexes for hold work table (ARSHOLDWORK)</i>		
Index name	Columns in index	Type of index
arsholdwork_idx	agid, aid, name	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Load table (ARSLOAD)

Use this table to track loads for expiration by load. This table maintains a record of all successful loads to application groups with the "expire by load" expiration type.

Table 73: Load table (ARSLOAD)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
pri_nid	SMALLINT	Primary storage node identifier
sec_nid	SMALLINT	Secondary storage node identifier
name	VARCHAR(11)	Name of load
start	BIGINT (DB2, SQL Server), INT (Oracle)	Start date in segment
stop	BIGINT (DB2, SQL Server), INT (Oracle)	Stop date in segment
exp_date	BIGINT (DB2, SQL Server), INT (Oracle)	Expiration date
aid	INTEGER	Application ID
total_docs	INTEGER	Number of documents in load
start_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Start date
stop_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Stop date
exp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Expire date

Table 74: Indexes for load table (ARSLOAD)

Index name	Columns in index	Type of index
arsload_idx	agid, exp_date	Cluster index for DB2; Index for Oracle and SQL Server
arsload_name_idx	agid, name	Index
arsload_dt_idx	agid, exp_dt	Index

Named query table (ARSNAMEQ)

This table maintains a catalog of named queries.

Table 75: Named query table (ARSNAMEQ)

Column Name	Data Type	Description
fid	INTEGER	Folder identifier
id	INTEGER	User identifier
name	VARCHAR(60)	Name of named query
buf	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Named query

<i>Table 76: Indexes for named query table (ARSNAMEQ)</i>		
Index name	Columns in index	Type of index
arsnameq_idx	fid, id, name	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Node table (ARSNODE)

This table maintains a catalog of storage nodes.

Table 77: Node table (ARSNODE)

Column Name	Data Type	Description
sid	INTEGER	Storage set identifier
nid	SMALLINT	Storage node identifier
name	VARCHAR(60)	Name of storage node
description	VARCHAR(120)	Description of storage node
logon	VARCHAR(128)	Storage manager logon userid
passwd	VARCHAR(128)	Storage manager logon password
server	VARCHAR(254)	Name of OnDemand object server
status	SMALLINT	Current location of storage set
sec_nid	SMALLINT	Secondary storage node
upd_userid	VARCHAR(128)	Userid of last user to update storage node
upd_date	BIGINT	Date and time storage node was last updated (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time storage node was last updated

<i>Table 78: Indexes for node table (ARSNODE)</i>		
Index name	Columns in index	Type of index
arsnode_idx	sid, nid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

ODF distribution definition table (ARSDFDCT)

This table maintains a catalog of distributions.

Table 79: Distribution table (ARSDFDCT)

Column Name	Data Type	Description
DCT_RECIPIENT	VARCHAR(60)	Recipient/list name for this distribution
DCT_DIST_NAME	VARCHAR(60)	Distribution name
DCT_DIST_METHOD	VARCHAR(8)	Distribution method

Table 79: Distribution table (ARSDFDCT) (continued)

Column Name	Data Type	Description
DCT_STATUS	CHAR(1)	Status
DCT_JNAME	VARCHAR(8)	Job name
DCT_FORMDEF	VARCHAR(8)	Form definition name
DCT_PAGEDEF	VARCHAR(8)	Page definition name
DCT_DEST	VARCHAR(19)	Destination
DCT_FORMSID	VARCHAR(8)	Forms ID name
DCT_CLASS	CHAR(1)	Class
DCT_WRITER	VARCHAR(8)	Writer name
DCT_OUTLIM	INTEGER	Output limit
DCT_FCB	VARCHAR(4)	FCB
DCT_UCS	VARCHAR(4)	UCS
DCT_ACCOUNT	VARCHAR(60)	Account information
DCT_CONT_WAIT	CHAR(1)	Continue/wait indicator
DCT_CONT_MAXTRIES	INTEGER	Maximum number of times to continue print processing
DCT_MANIFEST	CHAR(1)	Manifest Indicator
DCT_RPT_BREAK	CHAR(1)	Report break indicator
DCT_LOCATION	CHAR(1)	Location
DCT_CUSTVR	VARCHAR(55)	Customer variables
DCT_EXITINFO	VARCHAR(400)	Exit program information
DCT_DATASET	VARCHAR(254)	GDG dataset base name
DCT_UPD_USERID	VARCHAR(128)	Last user to update the distribution
DCT_UPD_DATE	BIGINT	Date and time distribution was last updated

Table 80: Indexes for distribution definition table (ARSDFDCT)

Index name	Columns in index	Type of index
ARSDFDCT_IDX1	DCT_RECIPIENT, DCT_DIST_NAME	Unique cluster index for DB2; Unique index for Oracle and SQL Server
ARSDFDCT_IDX2	DCT_DIST_NAME	Unique index
ARSDFDCT_IDX3	DCT_JNAME	Index

ODF distribution request table (ARSDFDRT)

This table maintains a catalog of distributions that have been scheduled for processing.

Table 81: Distribution request table (ARSDFDRT)

Column Name	Data Type	Description
DRT_RECIPIENT	VARCHAR(60)	Recipient name
DRT_DIST_NAME	VARCHAR(60)	Distribution name
DRT_ORIGINATOR	VARCHAR(8)	Identifier who scheduled the distribution
DRT_CREATE_DT	TIMESTAMP	Timestamp when the distribution request was created
DRT_PROCESS_DT	TIMESTAMP	Timestamp when the distribution request was processed
DRT_COMPLETE_DT	TIMESTAMP	Timestamp when the distribution request completed
DRT_STATUS	INTEGER(4)	Status of distribution request
DRT_CONT_ATTEMPTS	INTEGER(4)	Number of continuation attempts made
DRT_REPRINT	INTEGER(4)	Reprint indicator

Table 82: Indexes for distribution request table (ARSDFDRT)

Index name	Columns in index	Type of index
ARSDFDRT_IDX1	DRT_RECIPIENT, DRT_DIST_NAME	Unique cluster index for DB2; Unique index for Oracle and SQL Server
ARSDFDRT_IDX2	DRT_DIST_NAME	Unique index

ODF document status table (ARSDFDST)

This table maintains a catalog of OnDemand documents that are available for distribution.

Table 83: Document status table (ARSDFDST)

Column Name	Data Type	Description
DST_RECIPIENT	VARCHAR(60)	Recipient name
DST_DIST_NAME	VARCHAR(60)	Distribution name
DST_REPORTID	VARCHAR(60)	Report ID
DST_APPLGRP_NAME	VARCHAR(60)	Application group name
DST_APPL_NAME	VARCHAR(60)	Application name
DST_AGID	INTEGER(4)	Application group ID
DST_AID	INTEGER(4)	Application ID
DST_LOADID	VARCHAR(128)	Load ID
DST_LOAD_SIZE	BIGINT(8)	Number of bytes loaded
DST_RID	INTEGER(4)	Resource ID
DST_RES_SIZE	BIGINT(8)	Resource size

Table 83: Document status table (ARSDFDST) (continued)

Column Name	Data Type	Description
DST_CREATE_DT	TIMESTAMP	Timestamp when the document entry was created
DST_PROCESS_DT	TIMESTAMP	Timestamp when the document distribution was processed
DST_COMPLETE_DT	TIMESTAMP	Timestamp when the document distribution completed
DST_STATUS	INTEGER(4)	Status

Table 84: Indexes for document status table (ARSDFDST)

Index name	Columns in index	Type of index
ARSDFDST_IDX1	DST_RECIPIENT, DST_DIST_NAME, DST_REPORTID	Unique cluster index for DB2; Unique index for Oracle and SQL Server

ODF print processor table (ARSDFPPT)

This table maintains a catalog of report bundles that are submitted for print processing.

Table 85: Print processor table (ARSDFPPT)

Column Name	Data Type	Description
PPT_RECIPIENT	VARCHAR(60)	Recipient name
PPT_DIST_NAME	VARCHAR(60)	Distribution name
PPT_SEQUENCE	INTEGER(4)	Sequence number
PPT_REPORTID	VARCHAR(60)	Report ID
PPT_LISTID	VARCHAR(60)	List ID
PPT_APPLGRP_NAME	VARCHAR(60)	Application group name
PPT_APPL_NAME	VARCHAR(60)	Application name
PPT_AGID	INTEGER(4)	Application Group ID
PPT_AID	INTEGER(4)	Application ID
PPT_LOADID	VARCHAR(128)	Load ID
PPT_LOAD_SIZE	BIGINT(8)	Number of Bytes Loaded
PPT_RID	INTEGER(4)	Resource ID
PPT_RES_SIZE	DB2 TIMESTAMP	Resource size
PPT_LOCATION	CHAR(1)	Location
PPT_FILE_DIR	VARCHAR(254)	File directory
PPT_EMAIL	VARCHAR(254)	E-mail address
PPT_TRANSFORM	INTEGER(4)	Transform indicator; 0=None, 1=UTF-8, 3=PDF
PPT_CREATE_DT	TIMESTAMP	Timestamp when the report bundle request was created

Table 85: Print processor table (ARSDFPPT) (continued)

Column Name	Data Type	Description
PPT_PROCESS_DT	TIMESTAMP	Timestamp when the report bundle was processed
PPT_COMPLETE_DT	TIMESTAMP	Timestamp when the report bundle completed
PPT_STATUS	INTEGER(4)	Status
PPT_LINE_COUNTER	BIGINT(8)	Line counter
PPT_PAGE_COUNTER	INTEGER(4)	Page counter
PPT_EXCPT_MSG_NUM	INTEGER(4)	Exception message number
PPT_REPRINT	INTEGER(4)	Reprint indicator
PPT_JNAME	VARCHAR(8)	Job name
PPT_FORMDEF	VARCHAR(8)	Form definition name
PPT_PAGEDEF	VARCHAR(8)	Page definition name
PPT_FORMSID	VARCHAR(8)	Forms ID
PPT_DEST	VARCHAR(19)	Destination
PPT_CLASS	CHAR(1)	Class
PPT_WRITER	VARCHAR(8)	Writer name
PPT_FCB	VARCHAR(4)	FCB
PPT_UCS	VARCHAR(4)	UCS
PPT_ACCOUNT	VARCHAR(60)	Account
PPT_BURST	CHAR(1)	Burst
PPT_COPYM	VARCHAR(8)	COPYM
PPT_FLASH	VARCHAR(6)	Flash
PPT_CUSTVR	VARCHAR(55)	Customer variables
PPT_EXITINFO	VARCHAR(400)	Exit information
PPT_OUTLIM	INTEGER(4)	Output limit
PPT_NCOPY	INTEGER(4)	Number of copies
PPT_STARTPAGE	BIGINT(8)	First page of range of pages to print
PPT_ENDPAGE	BIGINT(8)	Last page of range of pages to print
PPT_SQL	CLOB(4)	SQL query select statement

Table 86: Indexes for print processor table (ARSDFPPT)

Index name	Columns in index	Type of index
ARSDFPPT_IDX1	PPT_RECIPIENT, PPT_DIST_NAME, PPT_SEQUENCE	Unique cluster index for DB2; Unique index for Oracle and SQL Server

ODF recipient list table (ARSDFLIS)

This table maintains a catalog of recipient lists. The table contains one row per recipient in a particular recipient list.

Table 87: Recipient list table (ARSDFLIS)

Column Name	Data Type	Description
LIS_LISTID	VARCHAR(60)	Recipient list name
LIS_RECIPIENT	VARCHAR(60)	Recipient name

Table 88: Indexes for recipient list table (ARSDFLIS)

Index name	Columns in index	Type of index
ARSDFLIS_IDX1	LIS_LISTID, LIS_RECIPIENT	Unique cluster index for DB2; Unique index for Oracle and SQL Server
ARSDFLIS_IDX2	LIS_RECIPIENT	Unique index

ODF recipient/list table (ARSDFUOT)

This table maintains a catalog of recipients and recipient lists.

Table 89: Recipient/list table (ARSDFUOT)

Column Name	Data Type	Description
UOT_RECIPIENT	VARCHAR(60)	Recipient or recipient list name
UOT_IS_LIST	INTEGER	Name is a recipient list
UOT_HDR1	VARCHAR(60)	Header line 1
UOT_HDR2	VARCHAR(60)	Header line 2
UOT_HDR3	VARCHAR(60)	Header line 3
UOT_HDR4	VARCHAR(60)	Header line 4
UOT_HDR5	VARCHAR(60)	Header line 5
UOT_HDR6	VARCHAR(60)	Header line 6
UOT_HDR7	VARCHAR(60)	Header line 7
UOT_HDR8	VARCHAR(60)	Header line 8
UOT_BANNER	CHAR(1)	Banner
UOT_ACCOUNT	VARCHAR(60)	Account
UOT_ADDRESS1	VARCHAR(60)	Address line 1
UOT_ADDRESS2	VARCHAR(60)	Address line 2
UOT_ADDRESS3	VARCHAR(60)	Address line 3
UOT_ADDRESS4	VARCHAR(60)	Address line 4
UOT_BUILDING	VARCHAR(60)	Building
UOT_DEPT	VARCHAR(60)	Department
UOT_NAME	VARCHAR(60)	Name of recipient

Table 89: Recipient/list table (ARSDFUOT) (continued)

Column Name	Data Type	Description
UOT_ROOM	VARCHAR(60)	Room
UOT_TITLE	VARCHAR(60)	Title of recipient
UOT_EMAIL	VARCHAR(254)	Email address of recipient
UOT_DESC	VARCHAR(120)	Description
UOT_UPD_USERID	VARCHAR(128)	Last user to update the recipient or recipient list
UOT_UPD_DATE	BIGINT	Date and time recipient or recipient list was last updated

Table 90: Indexes for recipient/list table (ARSDFUOT)

Index name	Columns in index	Type of index
ARSDFUOT_IDX1	UOT_RECIPIENT	unique index

ODF report bundle definition table (ARSDFBDT)

This table maintains a catalog of report bundle definitions.

Table 91: Report bundle definition table (ARSDFBDT)

Column Name	Data Type	Description
BDT_RECIPIENT	VARCHAR(60)	Recipient name
BDT_DIST_NAME	VARCHAR(60)	Distribution name
BDT_SEQUENCE	INTEGER	Sequence number
BDT_REPORTID	VARCHAR(60)	Report ID
BDT_STATUS	CHAR(1)	Status; A=Active, I= Inactive
BDT_REPORTBUILD	CHAR(1)	Report build; F=Full Report, Q=Query specified
BDT_JNAME	VARCHAR(8)	Job name
BDT_FORMDEF	VARCHAR(8)	Form definition name
BDT_PAGEDEF	VARCHAR(8)	Page definition name
BDT_DEST	VARCHAR(19)	Destination
BDT_FORMSID	VARCHAR(8)	Forms ID
BDT_CLASS	CHAR(1)	Class
BDT_WRITER	VARCHAR(8)	Writer name
BDT_OUTLIM	INTEGER	Output limit
BDT_FCB	VARCHAR(4)	FCB
BDT_UCS	VARCHAR(4)	UCS
BDT_WAIT_IGNORE	CHAR(1)	Wait/ignore indicator
BDT_BURST	CHAR(1)	Burst

Table 91: Report bundle definition table (ARSDFBDT) (continued)

Column Name	Data Type	Description
BDT_COPYM	VARCHAR(8)	COPYM
BDT_FLASH	VARCHAR(6)	Flash
BDT_NCOPY	INTEGER	Number of copies
BDT_LOCATION	CHAR(1)	Location
BDT_CUSTVR	VARCHAR(55)	Customer variables
BDT_EXITINFO	VARCHAR(400)	Exit information
BDT_DATASET	VARCHAR(254)	GDG dataset base name
CLOB	VARCHAR(30000)	CLOB query select statement
BDT_UPD_USERID	VARCHAR(128)	Last user to update the report bundle
BDT_UPD_DATE	BIGINT	Date and time report bundle was last updated

Table 92: Indexes for report bundle definition table (ARSDFBDT)

Index name	Columns in index	Type of index
ARSDFBDT_IDX1	BDT_RECIPIENT, BDT_DIST_NAME, BDT_SEQUENCE	Unique cluster index for DB2; Unique index for Oracle and SQL Server
ARSDFBDT_IDX2	BDT_DIST_NAME	Unique index
ARSDFBDT_IDX3	BDT_REPORTID	Unique index
ARSDFBDT_IDX4	BDT_JNAME	Unique index

ODF report ID table (ARSDFCRT)

This table maintains a catalog of report IDs.

Table 93: Report ID table (ARSDFCRT)

Column Name	Data Type	Description
CRT_REPORTID	VARCHAR(60)	Report ID name
CRT_STATUS	CHAR(1)	Status
CRT_APPLGRP_NAME	VARCHAR(60)	Application group name
CRT_APPL_NAME	VARCHAR(60)	Application name
CRT_REFERENCE	VARCHAR(254)	Reference
CRT_UPD_USERID	VARCHAR(128)	Last user to update the report ID
CRT_UPD_DATE	BIGINT	Date and time report ID was last updated

<i>Table 94: Indexes for report ID table (ARSDFCRT)</i>		
Index name	Columns in index	Type of index
ARSDFCRT_IDX1	CRT_REPORTID	Unique index
ARSDFCRT_IDX2	CRT_APPLGRP_NAME, CRT_APPL_NAME	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Printer table (ARSPRT)

This table maintains a catalog of printers.

Table 95: Printer table (ARSPRT)

Column Name	Data Type	Description
pid	INTEGER	Printer identifier
name	VARCHAR(60)	Name of printer
description	VARCHAR(120)	Description of printer
queue	VARCHAR(60)	Name of queue that is associated with the printer
type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Type of printer
upd_userid	VARCHAR(128)	Userid of last user to update printer
upd_date	BIGINT	Date and time printer was last updated (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time printer was last updated

<i>Table 96: Indexes for printer table (ARSPRT)</i>		
Index name	Columns in index	Type of index
arsprt_name_idx	name	Unique index
arsprt_pid_idx	pid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Printer options table (ARSPRTOPTS)

This table maintains a catalog of printing options (parameters) for printers.

Table 97: Printer options table (ARSPRTOPTS)

Column Name	Data Type	Description
id	INTEGER	Printer identifier
name	VARCHAR(60)	Name of printer options
type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Type of printer

Table 97: Printer options table (ARSPRTOPTS) (continued)

Column Name	Data Type	Description
buf	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	Printing options

Table 98: Indexes for printer options table (ARSPRTOPTS)

Index name	Columns in index	Type of index
arsprtopts_idx	id, name	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Printer user table (ARSPRTUSR)

This table maintains a catalog of user permissions for printers.

Table 99: Printer user table (ARSPRTUSR)

Column Name	Data Type	Description
pid	INTEGER	Printer identifier
id	INTEGER	User or group identifier

Table 100: Indexes for printer user table (ARSPRTUSR)

Index name	Columns in index	Type of index
arsprtusr_idx	pid, id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Resources table (ARSRES)

This table maintains a catalog of AFP resources.

Table 101: Resources table (ARSRES)

Column Name	Data Type	Description
rid	INTEGER	Resource identifier
agjd	INTEGER	Application group identifier
pri_nid	SMALLINT	Primary storage node identifier
sec_nid	SMALLINT	Secondary storage node identifier
comp_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Method of compression
add_date	BIGINT	Date and time (in old style) that the resource was loaded into the system
comp_size	INTEGER	Compressed size
decomp_size	INTEGER	Decompressed size

Table 101: Resources table (ARSRES) (continued)

Column Name	Data Type	Description
exp_date	BIGINT (DB2, SQL Server), INT (Oracle)	Expiration date (in old style)
hold	SMALLINT	Whether or not on hold
add_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time that the resource was loaded into the system
exp_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Expiration date

Table 102: Indexes for resources table (ARSRES)

Index name	Columns in index	Type of index
arsres_idx	agid, rid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Segment table (ARSSEG)

This table maintains a catalog of segments of application group data. There is one row for each segment of application group data.

The segments of application group data are also known as dynamic tables that are created by OnDemand.

If there is more than one application group in a folder: OnDemand builds a query to the arsseg table for each application group that is specified, assuming that you have not limited your query to a specific application group. For example, if you have eight application groups, you search the arsseg table eight different times to find the tables applicable to your search.

Table 103: Segment table (ARSSEG)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
table_name	VARCHAR(18)	Application group segment table name
start_date	BIGINT	Segment start date (in old style)
stop_date	BIGINT	Segment stop date (in old style)
post_date	BIGINT	Date table was created (in old style)
closed_date	BIGINT	Date table was closed (in old style)
reimported_date	BIGINT	Date table was imported to the database (in old style)
last_update	BIGINT	Date table was last updated (in old style)
last_backup	BIGINT	Date of last table backup (in old style)
last_stats	BIGINT	Date statistics last collected (in old style)

Table 103: Segment table (ARSSEG) (continued)

Column Name	Data Type	Description
mask	INTEGER	Location
ins_rows	INTEGER	Inserted rows
upd_rows	INTEGER	Updated rows
del_rows	INTEGER	Deleted rows
mod_rows	INTEGER	Modified rows
max_rows	INTEGER	Maximum number of rows
database_name	VARCHAR (8)	The database name of the segment
fti_server	VARCHAR(254)	Full text index server
start_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Segment start date
stop_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Segment stop date
post_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date table was created
closed_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date table was closed
reimported_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date table was imported to the database
last_update_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date table was last updated
last_backup_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date of last table backup
last_stats_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date statistics last collected

Table 104: Indexes for segment table (ARSSEG)

Index name	Columns in index	Type of index
arsseg_name_idx	agid, table_name	Unique cluster index for DB2; Unique index for Oracle and SQL Server
arsseg_idx	agid, start_date, stop_date	Index
arsseg_cdate_idx	agid, closed_date	Index
arsseg_ssidx	agid, start_dt, stop_dt	Index
arsseg_cdt_idx	agid, closed_dt	Index

Storage set table (ARSSET)

This table maintains a catalog of storage sets.

Table 105: Storage set table (ARSSET)

Column Name	Data Type	Description
sid	INTEGER	Storage set identifier
name	VARCHAR(60)	Name of storage set
description	VARCHAR(120)	Description of storage set
load_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Type of load
upd_userid	VARCHAR(128)	Userid of last user to update storage set
upd_date	BIGINT	Date and time storage set was last updated (in old style)
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time storage set was last update

Table 106: Indexes for storage set table (ARSSET)

Index name	Columns in index	Type of index
arsset_name_idx	name	Unique index
arsset_sid_idx	sid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

System parameters table (ARSSYS)

This table contains a single row that specifies the current system parameters.

Table 107: System parameters table (ARSSYS)

Column Name	Data Type	Description
id	INTEGER	Identifier counter
name	VARCHAR(8)	Internal application group identifier
cdid	INTEGER	CD-ROM identifier
passwd_exp	INTEGER	Password expiration value
time_out	BIGINT (DB2, SQL Server), INT (Oracle)	System inactivity timeout value
syslog_mask	INTEGER	Message logging options
exitlog_mask	INTEGER	User exit logging options
passwd_min_len	SMALLINT	Minimum password length
nid	SMALLINT	Internal storage node identifier
sys_mask	INTEGER	Case of userids and passwords
buf_resv	CLOB(32700) (DB2), VARCHAR(MAX) (SQL Server), CLOB (Oracle)	reserved

Table 107: System parameters table (ARSSYS) (continued)

Column Name	Data Type	Description
upd_userid	VARCHAR(128)	Userid of last user to update system parameters
upd_date	BIGINT (DB2, SQL Server), INT (Oracle)	Date and time system parameters were last updated (in old style)
sys_mask2	INTEGER	Types of annotations
min_gjd	INTEGER	Group ID starting value
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time system parameters were last updated
failed_logins	INTEGER	Number of times user can attempt login
user_inactivity	INTEGER	Disable user that has not logged in after number of days
prev_passwds	INTEGER	Number of previous passwords to check before user can reuse a password
life_of_annot	INTEGER	Number of days OnDemand keeps an annotation
lockout_mins	INTEGER	Number of minutes to prevent user from attempting login
notify_pwd_days	INTEGER	Notify user for a number of days before password expires
svr_trace_on	INTEGER	Enable or disable tracing for the Content Manager OnDemand system
svr_trace_str	VARCHAR(254)	Specify parameters as instructed by IBM Software Support

Indexes: The system parameters table is not indexed

User table (ARSUSER)

This table maintains a catalog of users.

Table 108: User table (ARSUSER)

Column Name	Data Type	Description
uid	INTEGER	
userid	VARCHAR(128)	User name
passwd	VARCHAR(128)	User's password
description	VARCHAR(120)	Description of user
admin	SMALLINT	User type and authority
last_update	BIGINT (DB2, SQL Server), INT (Oracle)	Date password last set
pid	INTEGER	Default printer identifier

Table 108: User table (ARSUSER) (continued)

Column Name	Data Type	Description
acct_info	VARCHAR(60)	User-defined accounting information
name	VARCHAR(60)	User's name
company	VARCHAR(60)	User's organization
title	VARCHAR(60)	User's title
addr1	VARCHAR(60)	User's address
addr2	VARCHAR(60)	User's address
addr3	VARCHAR(60)	User's address
addr4	VARCHAR(60)	User's address
dept	VARCHAR(60)	User's department
building	VARCHAR(60)	User's building or location
room	VARCHAR(60)	User's office or room
phone	VARCHAR(32)	User's phone number
fax	VARCHAR(32)	User's fax number
cover_page	VARCHAR(60)	Default fax cover page
time_out	BIGINT (DB2, SQL Server), INT (Oracle)	User's inactivity timeout value
email	VARCHAR(254)	User's e-mail address
upd_userid	VARCHAR(128)	Userid of last user to update user
upd_date	BIGINT (DB2, SQL Server), INT (Oracle)	Date and time user was last updated (in old style)
passwd_exp	INTEGER	Password expiration
uclass	VARCHAR(1)	Printer class value
banner	VARCHAR(1)	Printer banner
dest	VARCHAR(8)	Printer dest value
node	VARCHAR(8)	The system name
writer	VARCHAR(8)	Printer writer value
pagedef	VARCHAR(8)	Printer PageDef
formdef	VARCHAR(8)	Printer FormDef
forms	VARCHAR(8)	Printer Forms
routing	VARCHAR(24)	Printer routing field
misc390	VARCHAR(254)	Reserved for future use
upd_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time user was last updated
failed_logins	INTEGER	Number of failed logins

Table 108: User table (ARSUSER) (continued)

Column Name	Data Type	Description
last_login_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time user logged into Content Manager OnDemand
last_pw_chng_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time user changed the password
prev_pwd2	VARCHAR(128)	Password previous to current password
prev_pwd3	VARCHAR(128)	Password previous to prev_pwd2
prev_pwd4	VARCHAR(128)	Password previous to prev_pwd3
prev_pwd5	VARCHAR(128)	Password previous to prev_pwd4
prev_pwd6	VARCHAR(128)	Password previous to prev_pwd5
prev_pwd7	VARCHAR(128)	Password previous to prev_pwd6
prev_pwd8	VARCHAR(128)	Password previous to prev_pwd7
prev_pwd9	VARCHAR(128)	Password previous to prev_pwd8
prev_pwd10	VARCHAR(128)	Password previous to prev_pwd9
lockout_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time user was locked out of OnDemand
last_action_dt	TIMESTAMP (DB2, Oracle), DATETIME2(6) (SQL Server)	Date and time user performed the most recent action

Table 109: Indexes for user table (ARSUSER)

Index name	Columns in index	Type of index
arsuser_id_idx	uid	Unique cluster index for DB2; Unique index for Oracle and SQL Server
arsuser_name_idx	userid	Unique index

Users in groups table (ARSUSRGRP)

This table maintains the association of groups with individual users.

Table 110: Users in groups table (ARSUSRGRP)

Column Name	Data Type	Description
gid	INTEGER	Group identifier
uid	INTEGER	User identifier

Table 111: Indexes for users in groups table (ARSUSRGRP)

Index name	Columns in index	Type of index
arsusrgrp_idx	gid, uid	Unique cluster index for DB2; Unique index for Oracle and SQL Server

User group ID table (ARSUSRGRPID)

This table contains the user permissions for a specific user. The information is contained on the User Permissions page of a user.

Table 112: User group ID table (ARSUSRGRPID)

Column Name	Data Type	Description
id	INTEGER	User identifier
owner_id	INTEGER	Identifier (uid or gid) with the permissions specified by the perms field
perms	INTEGER	User authority for the group

Table 113: Indexes for user group ID table (ARSUSRGRPID)

Index name	Columns in index	Type of index
arsusrgrpid_idx	id, owner_id	Unique cluster index for DB2; Unique index for Oracle and SQL Server

Application group data table

An application group data table contains one row for each document that is stored in the application group. The name of an application group data table is `owner.applgrpInternalIdn`, where `owner` is the value of the `SRVR_INSTANCE_OWNER` parameter for the instance from the `ARS.INI` file, `applgrpInternalId` is the identifier that is assigned to the application group when the application group is created, and `n` is an incremental value. For example, `arsdbas3.rba1`, `arsdbas3.rba2`, and so on. The application group identifier is listed on the Advanced Storage Management page of the application group.

The application group data tables are also known as dynamic tables that are created by OnDemand.

A row in an application group data table contains from 1 to 128 user-defined fields followed by approximately 40 bytes of Content Manager OnDemand control information (fields).

Table 114: Application group data table

Column Name	Data Type	Description
user_field_1	varies	First user-defined application group field
user_field_n	varies	Last user-defined application group field
doc_name	VARCHAR(11)	Document name
doc_off	INTEGER	Document offset
doc_len	INTEGER	Document length
comp_off	INTEGER	Compression offset
comp_len	INTEGER	Compression length
annot	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Annotation flag
comp_type	CHAR(1) FOR BIT DATA (DB2), BINARY(1) (SQL Server), RAW(1) (Oracle)	Compression type
resource	INTEGER	Resource identifier
pri_nid	SMALLINT	Primary storage node identifier
sec_nid	SMALLINT	Secondary storage node identifier

Table 115: Indexes for application group data table

Index name	Columns in index	Types of index
doc_name	doc_name	Index
usr_index_1...usr_index_n		

Note:

1. This table is indexed on one or more of the user-defined fields, as represented by user_field_1 ... user_field_n in the table.
2. The index doc_name is used only when the Expiration Type of the application group is set to Load or Storage Manager.

Tablespaces for Content Manager OnDemand system tables

Table 116: Tablespaces for Content Manager OnDemand system tables

Table	Tablespace	LOCKSIZE	BUFFERPOOL
ARSAG	ARSAGT	ROW	BP0
ARSAG2FOL	ARSAG2FT	ANY	BP0
ARSAGFLD	ARSAGFLT	ANY	BP0
ARSAGFLDALIAS	ARSAGFAT	ANY	BP0
ARSAGINDEX	ARSAGIDT	ANY	BP0
ARSAGPERMS	ARSAGPET	ANY	BP32K
ARSANN	ARSANNT	ANY	BP32K
ARSAPP	ARSAPPT	ANY	BP32K
ARSAPPUSR	ARSAPPUT	ANY	BP32K
ARSCAB	ARSCABT	ANY	BP0
ARSCAB2FOL	ARSCABFT	ANY	BP0
ARSCABPERMS	ARSCABPT	ANY	BP0
ARSCFSODWORK	ARSCFSWT	ANY	BP0
ARSFOL	ARSFOLT	ANY	BP0
ARSFOLFLD	ARSFOLFT	ANY	BP0
ARSFOLFLDUSR	ARSFOLUT	ANY	BP0
ARSFOLPERMS	ARSFOLPT	ANY	BP0
ARSGROUP	ARSGROUT	ANY	BP0
ARSHOLD	ARSHLDT	ANY	BP0
ARSHOLDMAP	ARSHLDMT	ANY	BP0
ARSHOLDPERMS	ARSHLDPT	ANY	BP0
ARSHOLDWORK	ARSHLDWT	ANY	BP0
ARSLOAD	ARSLOADT	ANY	BP0
ARSNAMEQ	ARSNAMEQ	ANY	BP32K
ARSNODE	ARSNODET	ANY	BP0
ARSPRT	ARSPRTT	ANY	BP0
ARSPRTOPTS	ARSPRTOT	ANY	BP0
ARSPRTUSR	ARSPRUST	ANY	BP0

Table 116: Tablespaces for Content Manager OnDemand system tables (continued)

Table	Tablespace	LOCKSIZE	BUFFERPOOL
ARSRES	ARSREST	ANY	BP0
ARSSEG	ARSSEGT	ROW	BP0
ARSSET	ARSSETT	ANY	BP0
ARSSYS	ARSSYST	ANY	BP32K
ARSUSER	ARSUSERT	ANY	BP0
ARSUSRGRP	ARSUSRGT	ANY	BP0
ARSUSRGRPID	ARSUSGIT	ANY	BP0

Note:

1. LOCKSIZE ANY usually means PAGE, though DB2 reserves the right to change it to LOCKSIZE ROW in the future.
2. All table spaces are defined as a segmented tablespace with SEGSIZE 64, with the exception of ARSSYS, which has a SEGSIZE of 8. ARSSYS only has a single row.
3. ARSPRTOPTS is not used. It is assigned to a 4K buffer pool, with a minimum PRIQTY.

Objects and data model used in the Content Manager OnDemand XML file

XML objects in the Content Manager OnDemand XML file

You can use an XML interface to import data into a Content Manager OnDemand system. Before importing data, you need to have an XML file that contains all the data that needs to be imported. The XML file can be either created during a previous XML export process or written new. This section helps you to understand the objects in the XML file.

The following diagram illustrates the overall layout of the Content Manager OnDemand XML file.

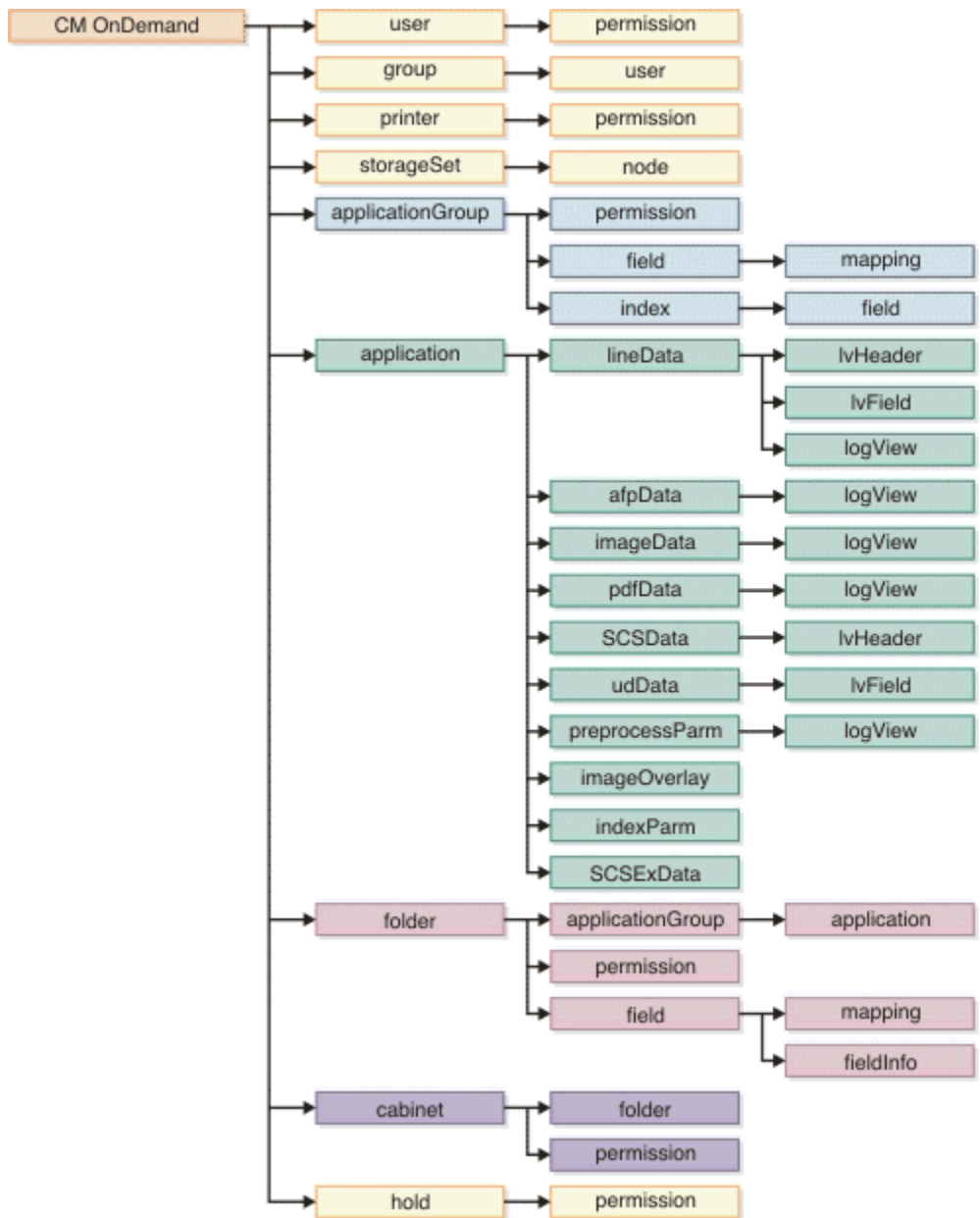


Figure 23: Overall layout of the Content Manager OnDemand XML file

The following tables provide detailed descriptions about the objects in the Content Manager OnDemand XML file. If you create these objects in an input XML file, ensure that you include them in the following order:

1. systemParameters
2. user
3. group
4. printer
5. storageSet
6. applicationGroup
7. application
8. folder
9. cabinet
10. hold

11. odfRecipient
12. odfRecipientList
13. odfReportId
14. odfDistribution with odfReportBundles

For each object, the following information is provided:

Name

The name of the attribute. An attribute name in bold indicates that the attribute is a "key" attribute, which is used to uniquely identify the object. For all top-level objects, this is the Name attribute.

Data Type

The XML data type of the attribute. Types in italics indicates an XML data type that is defined in the Content Manager OnDemand XML schema file.

Required

Whether this attribute is required. Dependent means that the attribute may be required based on the settings of other attributes.

Default Value

The value that is used internally if the optional attribute is not specified in the XML file.

Possible Values

Lists any limitations on the value for attributes. If there are no limitations, this is blank.

Updateable

Whether this attribute can be updated.

For each child object, the following information is provided:

Name

The name of the child object. An attribute name in bold indicates that the attribute is a "key" attribute which is used to uniquely identify the object.

Min/Max Number

Indicates the minimum and maximum number of children of this type that are allowed.

Deleteable

Whether this child object can be deleted.

Updateable

Whether this child object can be updated.

For all XML objects, the data that is associated with an object is only in the form of attribute data and child objects.

Important: The capitalization of the object names is important and should be used exactly as shown in the data tables.

System Parameters object

Table 117: Attributes for the systemParameters object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
allowPrivateUser	Allows the user to add private annotations to a document.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> • "No" • "Yes" 	Yes
allowPublic	Allows the user to add public annotations.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> • "No" • "Yes" 	Yes
allowGraphicAnnot	Allow the user to add graphic annotations.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> • "No" • "Yes" 	Yes

Table 117: Attributes for the systemParameters object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
allowTextAnnot	Allow the user to add text annotations.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	Yes
defaultAnnot	Specify the default viewing scope for all annotations.	<i>String</i>	No	""	<ul style="list-style-type: none"> "Public" "Private to User" "Private to Group" 	
disableOrLockUser	Specify whether to disable or lock out a user after the user attempts and fails to log in for the specified number of times.	<i>String</i>	No	""	<ul style="list-style-type: none"> "Never" "Disable User" "Lock Out User" 	Yes
enableComments	Display the System Log Comments window when you perform an add, update, or delete operation.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
enableLDAP	Specify whether you want to use LDAP (Lightweight Directory Access Protocol) authentication in your OnDemand server.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
failedLogins	Specify the number of times a user can attempt to log in.	<i>integer</i>	No	""	An integer between 1 and 10	Yes
inactivityTO	Specify the number of minutes between server requests that must pass, after which OnDemand notifies user to log off.	<i>integer</i>	No	""	An integer between 1 and 1440.	
lifeOfAnnotations	Specify the number of days OnDemand keeps annotations	<i>integer or string</i>	No	""	<ul style="list-style-type: none"> "Never expire" An integer between 1 and 365 	Yes
lockoutMinutes	Specify the number of minutes to prevent a user from attempting to log in.	<i>integer</i>	No	""	An integer between 1 and 1440	Yes
logAGMsgs	Specify whether OnDemand saves a message in the system log when a user queries or retrieves application group data and other types of application group events.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	

Table 117: Attributes for the systemParameters object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
logFailedLogin	Specify whether OnDemand saves a message in the system log when there is an unsuccessful log on attempt.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
logLogin	Specify whether OnDemand saves a message in the system log when a user logs on the server.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
logLogoff	Specify whether OnDemand saves a message in the system log when a user logs off the server.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
logUEAGMsgs	Specify whether the Content Manager OnDemand user exit saves a message in the system log when a user queries or retrieves application group data and other types of application group events.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
logUEFailedLogin	Specify whether the Content Manager OnDemand user exit saves a message in the system log when there is an unsuccessful log on attempt.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
logUELogin	Specify whether the Content Manager OnDemand user exit saves a message in the system log when a user logs on the server.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
logUELogoff	Specify whether the Content Manager OnDemand user exit saves a message in the system log when a user logs off the server.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
notifyPasswordDays	Notify users that their password will expire within the specified number of days.	<i>integer or string</i>	No	""	<ul style="list-style-type: none"> "Never notify" An integer from 1 - 30 	Yes

Table 117: Attributes for the systemParameters object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
passwordAge	Specify the number of days after which a password must be changed.	<i>integer or string</i>	No	""	<ul style="list-style-type: none"> "Never Expires" "Always Expires" An integer from 1 - 365 	
passwordLen	Specify whether a password is required. If a password is required, specify the minimum length of a password.	<i>integer or string</i>	No	""	<ul style="list-style-type: none"> "Permit Blank" An integer from 1 - 128 	
previousPasswords	Specify whether users can reuse a password.	<i>integer or string</i>	No	""	<ul style="list-style-type: none"> "Never check" An integer from 1 - 10 	Yes
pwdCaseSensitive	Specify whether a password must be case sensitive.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
requireComments	Requires you to enter one or more characters in the Comments field when you perform an add, update, or delete operation.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
serverTraceOn	Enable or disable tracing for the Content Manager OnDemand system.	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "No" "Yes" 	Yes
serverTraceString	Specify parameters as instructed by IBM Software Support.	<i>String</i>	No	""		Yes
uidCaseSensitive	Specify whether the user ID must be case sensitive.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> "No" "Yes" 	
updateDate		<i>String</i>	No	""		
userLoginInactivity	Specify whether to disable users that do not login after the specified number of days.	<i>integer or string</i>	No	""	<ul style="list-style-type: none"> "Never disable" An integer between 1 and 2147483647 	Yes

User object

Table 118: Attributes for the user object

Name	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the user	<i>nameString</i>	Yes		<ul style="list-style-type: none"> User name "_ALL" 	Yes

Table 118: Attributes for the user object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
uid	User ID value	Integer	No	0	<ul style="list-style-type: none"> Integer value 1–1,080,000 0 to indicate system generated values 	No
description	Description of the user	<i>descString</i>	No	""		Yes
password	User's password	<i>passwordString</i>	No	""		Yes
userType	Type of user	<i>userTypeString</i>	No	"User"	<ul style="list-style-type: none"> "User" "User Admin" "AG/Folder/Cabinet Admin" "Hold Admin" "System Admin" "System Admin (Read-Only)" 	Yes
timeOut	Time out length in minutes	<i>timeOutType</i>	No	"Use System Value"	<ul style="list-style-type: none"> "Use System Value" "No Limit" Any positive integer value 	Yes
fullName	User's full name	<i>userMiscString</i>	No	""		Yes
acctInfo	Accounting for the user	<i>userMiscString</i>	No	""		Yes
company	Name of the company	<i>userMiscString</i>	No	""		Yes
title	User's title of job responsibility	<i>userMiscString</i>	No	""		Yes
addr1	User's address	<i>userMiscString</i>	No	""		Yes
addr2	Address, line number 2	<i>userMiscString</i>	No	""		Yes
addr3	Address, line number 3	<i>userMiscString</i>	No	""		Yes

Table 118: Attributes for the user object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
addr4	Address, line number 4	<i>userMiscString</i>	No	""		Yes
dept	User's department name or number	<i>userMiscString</i>	No	""		Yes
building	User's building name or number	<i>userMiscString</i>	No	""		Yes
room	User's office name or number	<i>userMiscString</i>	No	""		Yes
phone	User's phone number	<i>phoneString</i>	No	""		Yes
fax	User's fax number	<i>phoneString</i>	No	""		Yes
email	User's email address	<i>emailString</i>	No	""		Yes
coverPage	Default Fax cover page for the user	<i>userMiscString</i>	No	""		Yes
printer	User's default printer	<i>nameString</i>	No	"NONE"		Yes
createCabinetsAuth	Whether this user or group has authority to create cabinets	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
createFoldersAuth	Whether this user have authority to create folders	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
createHoldsAuth	Whether this user has authority to create holds	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
createUsersAuth	Whether this user has authority to create users	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
createGroupsAuth	Whether this user have authority to create group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Table 118: Attributes for the user object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
createAppGroupsAuth	Whether this user have authority to create application groups	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
disableUser	Specifies whether this user should be disabled	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
passwordExp	Specifies when the user's password expires	<i>pwExpValue</i>	No	"Use System Value"	<ul style="list-style-type: none"> "Use System Value" "Never Expires" An integer between 1 and 365 	Yes
printerClass	The printer class	<i>char</i>	No	""		Yes
printerBanner	Specifies whether there should be a printer banner	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
printerDest	The printer destination	<i>valString</i>	No	""		Yes
printerNode	The printer node	<i>valString</i>	No	""		Yes
printerWriter	The printer writer	<i>valString</i>	No	""		Yes
printerPagedef	The printer pagedef	<i>valString</i>	No	""		Yes
printerFormdef	The printer formdef	<i>valString</i>	No	""		Yes
printerForms	The printer forms	<i>valString</i>	No	""		Yes
printerRouting	The printer routing	<i>routingString</i>	No	""		Yes

Table 119: Child object under the user object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes

Table 120: Attributes for the permission object under the user object

Attributes	Description	Datatype	Required	Default Value	Possible Values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
user	The name of a user that can view and maintain this user	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> User name 	No
group	The name of a group that can view and maintain this user	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> Group name 	No
adminAuthority	Whether this user or group has authority to update and delete the user	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Remember: The "_ALL" value can only be used in an ARSXML export operation.

In the permission child, you must specify either a user or a group object, however, you cannot specify both.

Group object

Table 121: Attributes for the group object

Name	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the group	<i>nameString</i>	Yes		<ul style="list-style-type: none"> Group name "_ALL" 	Yes
gid	Group ID value	Integer	No	0	<ul style="list-style-type: none"> Integer value greater than or equal to 1080001 0 to indicate a system value 	No
description	Description of this group	<i>descString</i>	No	""		Yes
ownerUser	Name of the user that is the owner of this group	<i>nameString</i>	No	"NONE"		Yes

Table 121: Attributes for the group object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
ownerGroup	Name of the group that is the owner of this group	<i>nameString</i>	No	"NONE"		Yes

Note: You can specify an ownerUser or an ownerGroup attribute, however, you cannot specify both.

Table 122: Child object under the group object

Name	Minimum number	Maximum number	Deleteable	Updateable
user	0	Unbounded	Yes	No

Table 123: Attributes for the user object under the group object

Attributes	Description	Datatype	Required	Default Value	Possible Values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" 	N/A
name	The name of a user that is a member of this group	<i>nameString</i>	Yes			No

Remember: The "_ALL" value can only be used in an ARSXML export operation.

Printer object

Table 124: Attributes for the printer object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the printer	<i>nameString</i>	Yes		<ul style="list-style-type: none"> Printer name "_ALL" 	Yes
description	A description of the printer	<i>descString</i>	No	""		Yes
queue	The printer queue	<i>queueString</i>	Yes			Yes
printerType	The type of printer	<i>printerTypeString</i>	No	"Printer"	<ul style="list-style-type: none"> "Printer" "Fax" "Printer With Information" 	Yes

Table 125: Child objects under the printer object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	No

Table 126: Attributes for the permission object under the printer object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" 	N/A
user	The user to give permission to	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> User name 	No
group	The group to give permission to	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> Group name "*PUBLIC" 	No

Remember: The "_ALL" value can only be used in an ARSXML export operation.

Storage set object

Table 127: Attributes for the storageSet object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the storage set	<i>nameString</i>	Yes		<ul style="list-style-type: none"> Storage set name "_ALL" 	Yes
description	A description of the storage set	<i>descString</i>	No	""		Yes
storageType	The load type of the storage set	<i>storageTypeString</i>	No	"Fixed"	<ul style="list-style-type: none"> "Fixed" "Local" 	Yes

Table 128: Child objects under the storageSet object

Name	Minimum number	Maximum number	Deleteable	Updateable
node	1	Unbounded	Yes (unless this is the only object left)	Yes

Table 129: Attributes for the node object under the storageSet object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
name	Name of the storage set node	<i>nameString</i>	Yes			No

Table 129: Attributes for the node object under the storageSet object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
description	A description of the storage set node	<i>descString</i>	No	""		Yes
server	The server where the node resides	<i>string</i>	No	"*ONDEMAND"		Yes
logon	Client node name	<i>nameString</i>	Dependent	""		Yes
password	Client node password	<i>passwordString</i>	Dependent	""		Yes
loadData	Whether this is a load data node	<i>yesNoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
accessMethod	Determines the type of archive storage for the primary node	<i>accessString</i>	No	"TSM"	<ul style="list-style-type: none"> • "Cache" • "ASM" • "OAM" • "TSM" • "VSAM" 	Yes
configFile	Name of the configuration file for HDFS, ICOS, S3, Swift, and TSM access methods	<i>string</i>	No	""		Yes
reloadHoldData	Whether this node should be used to reload hold data	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Remember: The "_ALL" value can only be used in an ARSXML export operation.

The logon and password attributes might be required based on the type of node.

Application group object

Table 130: Attributes for the applicationGroup object

Name	Description	Data Type	Required	Default value	Possible Values	Updateable
autoFTI	All data loaded into the Content Manager OnDemand server is processed by the full text indexer.	<i>yesnoString</i>	No	""	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
name	Name of the application group	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • Application group name • "_ALL" 	Yes
database	Name of the database	<i>nameString</i>	No	""		Yes
description	Description of the application group	<i>descString</i>	No	""		Yes
cacheData	Whether to store data in cache	<i>yesnoString</i>	No	Yes	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
docDataLen	How many days to cache the data for	<i>docDataLenValue</i>	No	90	<ul style="list-style-type: none"> • "No Cache" • "Search Cache" • A positive integer 	Yes
storageSet	Name of the storage set to use for this application group	<i>nameString</i>	No	""		Yes
CFSOD	Use CFSOD	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Table 130: Attributes for the applicationGroup object (continued)

Name	Description	Data Type	Required	Default value	Possible Values	Updateable
declareCFSOD	Use to declare CFSOD	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
expirationType	Determines how data is deleted from the application group	expTypeString	No	"Load"	<ul style="list-style-type: none"> • "Load" • "Segment" • "Document" • z/OS only option: "Storage Manager" 	No
expirationDate	The number of days that Content Manager OnDemand keeps documents, resources, and index data in the application group	expDateValue	No	90	<ul style="list-style-type: none"> • "Never Expire" • A positive integer 1-99999 	Yes
federateCFSOD	Federate CFSOD	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
FTI	Whether to enable Full Text Indexing for the application group	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
encryption	Whether to enable encryption for the application group	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	No
lastLoadDate	The date and time of the last load	String	No	""		No
lastQueryDate	The date and time of the last query	String	No	""		No
lastRetrieveDate	The date and time of the last retrieve	String	No	""		No
logRetrieval	Whether document retrieval messages should be logged	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logBulkRetrieval	Whether document bulk retrieval messages should be logged	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logDBQueriesAfter	Whether database queries (after the query) should be logged	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logDBQueriesBefore	Whether document database queries should be logged	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logServerPrinting	Whether document server printing messages should be documented	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logIndexAdd	Whether document index add messages should be documented	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logIndexUpdate	Whether document index update messages should be documented	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logIndexDelete	Whether document index delete messages should be documented	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logObjectStore	Whether object store messages should be documented	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logObjectRetrieve	Whether object retrieve messages should be logged	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
logAnnotAdd	Whether annotation add messages should be logged	yesnoString	No	"Yes"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Table 130: Attributes for the applicationGroup object (continued)

Name	Description	Data Type	Required	Default value	Possible Values	Updateable
logAnnotUpdate	Whether annotation update messages should be logged	yesnoString	No	"Yes"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
logAnnotDelete	Whether annotation delete messages should be logged	yesnoString	No	"Yes"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
maxDBRows	The maximum number of database rows for this application group	Integer	No	10000000	<ul style="list-style-type: none"> Positive integer "Single table for all loads" 	Yes
loadsPerTable	The amount of loads per table	loadsString	No	"Multiple"	<ul style="list-style-type: none"> "Multiple" 	No
dbTablespaceType	The type of database table space to use for this application group	dbtString	No	"SMS"	<ul style="list-style-type: none"> "None" "SMS" "Automatic Storage" "Use Tablespace" 	Yes
dbCompress	For databases that support compression, Content Manager OnDemand compresses the tables or table spaces. If the database does not support compression, Content Manager OnDemand ignores this setting.	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	No
annot	Annotation flags in the document database tables	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	No
enhancedRetManagement	Use enhanced retention management	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
impliedHold	Use implied hold	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
objectSize	The object size in kilobytes	Integer	No	10,000	Positive integer	Yes
migrateCache	When to migrate data from cache	migrateString	No	"Load"	<ul style="list-style-type: none"> "Load" "Never" "Migration" A positive integer 	Yes
migrateIndex	When to migrate indexes from cache	migrateIndString	No	"Never"	<ul style="list-style-type: none"> "Never" A positive integer 	Yes
migrateIndexKeep	How long to keep imported migrated indexes	Integer	No	0	Nonnegative	Yes
parameterMarkers	Whether to use parameter markers	yesnoString	No	"Yes"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
serverFTI	IP address of the full text indexer server	String	No	""		Yes
useFilegroups	Specifies whether this application group uses file groups	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
updateExistingTables	Specifies whether tables should be updated if an index changes	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 130: Attributes for the applicationGroup object (continued)

Name	Description	Data Type	Required	Default value	Possible Values	Updateable
applQueryName	Name of the application that has all the application groups you want to export. When you specify applQueryName during export, the value for the name attribute is ignored. If you specify applQueryName, you cannot specify folderQueryName.	string	No	""		No
folderQueryName	Name of folder that has the application groups you want to export. When you specify folderQueryName during export, the values for the name attribute is ignored. If you specify folderQueryName, you cannot specify applQueryName.	string	No	""		No

Important:

1. The storageSet value can be updated only if it is currently set to "".
2. The default and possible dbTablespaceType values depend on the database that is used.

Remember: The "_ALL" value can only be used in an ARSXML export operation.

Table 131: Child objects for the applicationGroup object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes
field	1	128	No	Yes
index	0	Unbounded	Yes	No

Table 132: Attributes for the index object under the applicationGroup object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	Task to perform	taskString	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" 	N/A
name	Name of the index	indexName	Yes		A positive integer	No
cluster	Only for DB2. Whether it is a cluster index	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	No

Table 133: Child objects under the index object

Name	Minimum number	Maximum number	Deleteable	Updateable
field	2	16	Yes	No

Table 134: Attributes for the field object under the index object

Attributes	Description	Data Type	Required	Default	Possible values	Updateable
name	Name of the field	string	Yes			No

Table 134: Attributes for the field object under the index object (continued)

Attributes	Description	Data Type	Required	Default	Possible values	Updateable
order	The field order	<i>orderString</i>	No	"Ascending"	<ul style="list-style-type: none"> "Ascending" "Descending" 	No

Table 135: Attributes for the permission object under the applicationGroup object

Attributes	Description	Data Type	Required	Default	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
user	Provide permissions to the application group to this user	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> User name "*PUBLIC" 	No
group	Provide permissions to the application group to this group	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> Group name "*PUBLIC" 	No
docAddPerm	Whether this user or group should be allowed to add documents to this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docDeletePerm	Whether this user or group should be allowed to remove documents from this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docUpdatePerm	Whether this user or group should be allowed to update documents in this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docPrintPerm	Whether this user or group should be allowed to print documents from this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docCopyPerm	Whether this user or group should be allowed to copy documents from this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docCFSODPerm	Whether this user or group should be allowed to add CFS-CMOD documents to this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docFTIPerm	Whether this user or group should be allowed to add full text indexer documents to this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
docHoldPerm	Whether this user or group should be allowed to add holds to documents for this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
annotViewPerm	Whether this user or group be allowed to view annotation in this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
annotAddPerm	Whether this user or group be allowed to add annotation to this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 135: Attributes for the permission object under the applicationGroup object (continued)

Attributes	Description	Data Type	Required	Default	Possible values	Updateable
annotDeletePerm	Whether this user or group be allowed to delete annotation from this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
annotUpdatePerm	Whether this user or group be allowed to update annotation in this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
annotPrintPerm	Whether this user or group be allowed to print annotation from this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
annotCopyPerm	Whether this user or group be allowed to copy annotation from this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
authority	Specifies the authority given	<i>authagString</i>	No	"Access"	<ul style="list-style-type: none"> "Access" "Administrator" "Logical Views" 	Yes
queryRes	A query restriction for the user or group	<i>queryString</i>	No	""		Yes

Important: Either a user or a group must be specified, but not both.

Table 136: Attributes for the field object under the applicationGroup object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on the object	<i>taskString</i>	No	Same as the task attribute of the parent	<ul style="list-style-type: none"> "add" "update" 	N/A
name	Database field name	<i>nameString</i>	Yes			No
type	Index type	<i>fieldTypeString</i>	No	<ul style="list-style-type: none"> "Filter" Index 	<ul style="list-style-type: none"> "Filter" "Index" "Not in Database" 	Yes, but only from "Filter" to "Index" or from "Index" to "Filter"
CFSOD	Whether this field is a CFSOD field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 136: Attributes for the field object under the applicationGroup object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
dataType	The data type of the field	<i>dataTypeString</i>	No	"String"	<ul style="list-style-type: none"> • "Big Int" • "Date" • "Date (native)" • "Date/Time" • "Date/Time (native)" • "Date/Time (TZ)" • "Date/Time (TZ) (native)" • "Decimal" • "Integer" • "Small Int" • "String" 	No
segment	If the expiration type is segment, value of this field is used by Content Manager OnDemand to determine when to delete data from the application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	No
expireDate	Whether this field is used for the expiration date	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	No
lockdown	Whether this is a lockdown field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
log	Whether to use the system log	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
userExit	Whether to send data to the user exit	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
partition	Used to partition the index data across the multiple nodes	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
cluster	Only for DB2. Determines the clustering index for the application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
reference	Only for z/OS. Controls which reports are distributed from Content Manager OnDemand Distribution Facility.	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
stringCase	The case to use if the data type is string	<i>caseString</i>	No	"Upper"	<ul style="list-style-type: none"> • "Upper" • "Lower" • "Mixed" 	No

Table 136: Attributes for the field object under the applicationGroup object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
stringType	The type of string	<i>stringType</i>	No	"Fixed"	<ul style="list-style-type: none"> "Fixed" "Variable" 	No
stringLength	The length of the string	Integer	No	0	<ul style="list-style-type: none"> For fixed strings: 1–254 For variable length strings: 1–2000 	No
stringEmbedded	The embedded characters to remove from the string	string	No	""		Yes
stringLeading	The leading characters to remove from the string	string	No	""		Yes
stringTrailing	The trailing characters to remove from the string	string	No	""		Yes
appIDField	Whether this field is the application ID field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
pageCount	Specifies whether this is a page count field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
documentSize	Specifies whether this is a document size field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
updateable	Whether this field can be updated	<i>yesnoString</i>	No	"Yes"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
FTI	Whether to index this field with the Full Text Indexer	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
uniqueID	Whether this is a unique ID field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	No
hashSHA256	Whether this is a Hash SHA-256 field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	No
encryption	Whether this is an encryption field (used internally by the encryption process)	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	No

The following restrictions apply to the Not in Database value:

- Do not use a field whose type attribute has a definition of Not in Database to hold information such as a report ID that applies to all documents that are returned in the hit list.
- The Not in Database value allows you to create a segment field that is based off the start/end dates that are found in the segment table without capturing another date in the application group data table.
- You can query on a value by using a folder field that is mapped to the application group field whose type attribute has a definition of "Not In Database". The system performs a check to determine

whether the value matches the segment range. However, the value cannot be displayed because it is not stored anywhere.

- Use the "Not In Database" value for a database field that has been defined as segment field.
- You can use the "Not In Database" value to segment a report that is based on a value that you do not want to capture in the database as an index or filter. In other words, you only needed the field when you were indexing the report.

Table 137: Child objects under the field object

Name	Minimum number	Maximum number	Deleteable	Updateable
mapping	0	Unbounded	Yes	Yes
nls	0	26	Yes	Yes

Table 138: Attributes for the mapping object under field

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "update" • "delete" 	N/A
dbValue	The database value	<i>dbString2</i>	Yes			No
displayedValue	The displayed value	<i>dvString</i>	Yes			Yes

Table 139: Attributes for the nls object under field

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "update" • "delete" 	N/A

Table 139: Attributes for the nls object under field (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
language	The natural language	string	Yes	""	<ul style="list-style-type: none"> • "English" • "French" • "German" • "Italian" • "Japanese" • "Norwegian" • "Portuguese (Brazil)" • "Spanish" • "Chinese (Simplified)" • "Chinese (Traditional)" • "Danish" • "Finnish" • "Swedish" • "Korean" • "Dutch" • "Arabic" • "Czech" • "Greek" • "Hebrew" • "Croatian" • "English" • "Hungarian" • "Polish" • "Russian" • "Slovakian" 	Yes

Table 140: Child objects under the nls object

Name	Minimum number	Maximum number	Deleteable	Updateable
mapping	0	Unbounded	Yes	Yes

Table 141: Attributes for the mapping object under nls

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
dbValue	The database value	string	Yes			No
displayedValue	The displayed value	string	Yes			Yes

Application object

Table 142: Attributes for the application object

Name	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the application	nameString	Yes		<ul style="list-style-type: none"> • Application name • "_ALL" 	Yes
description	Description of the application	descString	No	""		Yes

Table 142: Attributes for the application object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
appGroup	The application group that this application belongs to	<i>nameString</i>	Yes			No
identifier	The application group identifier	<i>dbString2</i>	Yes	""	Determined by the application group	Yes
dataType	The data type of this application	<i>appTypeString</i>	No	<ul style="list-style-type: none"> • "AFP" • "SCS" 	<ul style="list-style-type: none"> • "AFP" • "Line" • "BMP" • "GIF" • "PCX" • "TIFF" • "PDF" • "JPEG" • "PNG" • "User Defined" • "OD Defined" • "E-Mail" • "None" • "SCS" • SCS-Extended • Global DJDE • XML 	No
indexer	The name of the indexer program to use with this application	<i>indexerString</i>	No	"None"	<ul style="list-style-type: none"> • "None" • "ACIF" • "Generic" • "PDF" • "OS/390" • "OS/400" • "XML" 	Yes
expirationDate	Use this value to override the expirationDate attribute value specified in the application group	<i>expDateValueA</i>	No	"Use AppGroup Value"	<ul style="list-style-type: none"> • "Use AppGroup Value" • A positive integer 1-99999 	Yes
dataCompression	The compression technique that is used for the data	<i>compressionString</i>	No	"OD77"	<ul style="list-style-type: none"> • "OD77" • "OD77Lite" • "OD77HW" • "OD77LiteHW" • "LZW12" • "LZW16" • "Disable" • "None" 	Yes

Table 142: Attributes for the application object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
resCompression	The compression technique that is used for the resources	<i>compressionString</i>	No	"OD77"	<ul style="list-style-type: none"> • "OD77" • "OD77Lite" • "OD77HW" • "OD77LiteHW" • "LZW12" • "LZW16" • "Disable" • "None" 	Yes
resourceComparison	The number of resource comparisons	<i>resCompValue</i>	No	50	0–9999	Yes
largeObject	Whether this application a Large Object	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
size	The number of pages or size of the object in kilobytes that is based on the setting of the largeObject attribute	Integer	No	100	Positive integer representing either object size or number of pages based on setting of largeObject	Yes
pageIdentifiers	Whether page identifiers should be used	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
postProcessorString	The post processor string	<i>postProcString</i>	No	""		Yes
printAmount	The default print option to specify which page or pages to print	<i>printAmString</i>	No	"All Pages"	<ul style="list-style-type: none"> • "All Pages" • "Current Pages" 	Yes
defaultPrinter	Default printer	<i>nameString</i>	No	"*NONE"		Yes
cicsJCLParms	Values that are used by the CICS or ESA client program to print documents	string	No	"None"		Yes
printParms	Print parameters that are used by the server print manager	string	No	""		Yes
afpToLine	Options that OnDemand uses to convert AFP data to line data	string	No	""		Yes
usePreview	Whether to use the preview user exit	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
previewParms	Whether the preview parameters should be used	string	No	""		Yes

Table 142: Attributes for the application object (continued)

Name	Description	Data Type	Required	Default value	Possible values	Updateable
agQueryName	Name of the application group that has all the applications you want to export. When you specify agQueryName during export, the values for the name and appGroup attributes are ignored. If you specify agQueryName, you cannot specify folderQueryName.	string	No	""		No
folderQueryName	Name of folder that has the applications you want to export. When you specify folderQueryName during export, the values for the name and appGroup attributes are ignored. If you specify folderQueryName, you cannot specify agQueryName.	string	No	""		No

Note:

1. If the application group contains more than one field object, identifier is required.
2. The identifier object can be updated to an unused field value only.

Table 143: Child objects under the application object

Name	Minimum number	Maximum number	Deleteable	Updateable
lineData	0	1	No	Yes
SCSData	0	1	No	Yes
afpData	0	1	No	Yes
pdfData	0	1	No	Yes
imageData	0	1	No	Yes
udData	0	1	No	Yes
indexParm	0	1	Yes	Yes
preprocessParm	0	Dependent	No	Yes
imageOverlay	0	Unbounded	Yes	No

Important: The udData child is required if the application dataType is User Defined.

Table 144: Attributes for the SCSDData object under the application object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
codePage	The code page to use	Integer	No	37	Nonnegative integer	Yes
prMode	Processing Mode	<i>prmodeString</i>	No	"None"	<ul style="list-style-type: none"> • "None" • "SOSI1" • "SOSI2" • "SOSI3" 	Yes

Table 145: Child objects under the SCSDData object

Name	Minimum number	Maximum number	Deleteable	Updateable
lvHeader	0	1	No	Yes
lvField	0	30	Yes	Yes
logView	0	20	Yes	Yes

See Table 151 on page 350, Table 152 on page 350, and Table 153 on page 351 for descriptions of the lvHeader, lvField, and logView objects.

Table 146: Attributes for the SCSExData object under the application object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
codePage	The code page to use	Integer	No	37	Nonnegative integer	Yes
orientation	The orientation or rotation angle, in degrees, of the document	<i>orientString</i>	No	0	<ul style="list-style-type: none"> • 0 • 90 • 180 • 270 	Yes
paperSize	The output paper size	<i>paperString</i>	No	"Autosize"	See the list that follows this table for possible values for the paperSize attribute.	Yes

Possible values for the paperSize attribute are:

- A3
- A4 Wide
- A4
- A5
- Autosize
- B4 Wide
- B4

- B5
- Cut Sheet
- Euro Fanfold
- Executive 1
- Executive 2
- Executive 3
- Folio
- Ledger
- Legal Wide
- Legal
- Letter Wide
- Letter
- Narrow
- Quarto
- Short
- Statement
- Stationery
- Tabloid
- Wide
- 1403W
- 1403WS
- 3800N
- 3800NS
- 3800W
- 3800WS

Table 147: Child object under the SCSExData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	20	Yes	Yes

Table 148: Attributes for the logView object under the SCSExData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" • "update" 	N/A
name	Name of this logical view	<i>nameString</i>	Yes			Yes
user	User for this logical view	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • User name • "**PUBLIC" 	No
defaultView	Whether this is the default view	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Table 148: Attributes for the logView object under the SCSExData object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
zoom	Zoom Factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected Area Color	saColorString	No	"Yellow"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom" 	Yes
bgColor	Background	bgColorString	No	"White"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Green Bar" • "Custom" 	Yes

Table 149: Attributes for the lineData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
lineCount	The line count	Integer	No	0	Nonnegative integer	Yes
codePage	The code page to use	Integer	No	500	Positive integer	Yes
recFM	The record format	recfmString	No	"Variable"	<ul style="list-style-type: none"> • "Fixed" • "Variable" • "Stream" 	Yes
LRECL	The record length	Integer	No	133	Positive integer	Yes
delimiter	The line delimiter (specified in hex)	hexString	No	"0A"	Hex character string	Yes
prMode	Processing mode	prmodeString	No	"None"	<ul style="list-style-type: none"> • "None" • "SOSI1" • "SOSI2" • "SOSI3" 	Yes

Table 149: Attributes for the lineData object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
trc	Whether the data contains table reference characters	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
CCType	The type of the carriage control character	yesnoString	No	"ANSI"	<ul style="list-style-type: none"> "ANSI" "Machine" "None" 	Yes

Table 150: Child objects under the lineData object

Name	Minimum number	Maximum number	Deleteable	Updateable
lvHeader	0	1	No	Yes
lvField	0	30	Yes	Yes
logView	0	20	Yes	Yes

Table 151: Attributes for the lvHeader object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
pageHdrRows	Number of page header rows	Integer	No	0	Nonnegative integer	Yes
fieldHdrRows	Number of field header rows	Integer	No	0	Nonnegative integer	Yes
validationRow	The validation row	Integer	Dependent	0	Nonnegative integer	Yes
validationCol	The validation column	Integer	Dependent	0	Nonnegative integer	Yes
validationString	The validation string	String	Dependent	""		Yes

Table 152: Attributes for the lvField object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	taskString	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
name	The name of the logical view field	nameString	Yes			Yes
dbName	The application group database name	nameString	No	"*NONE"	database field name or "*NONE"	Yes

Table 152: Attributes for the lvField object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
startCol	The start column for this field definition	Integer	Yes	0	An integer value between 1 and 999	Yes
endCol	The end column for this field definition	Integer	Yes	0	an integer value between 1 and 999	Yes

Table 153: Attributes for the logView object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" • "update" 	N/A
name	Name of this logical view	<i>nameString</i>	Yes			Yes
user	user for this logical (or PUBLIC) view	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • User name • "**PUBLIC" 	No
defaultView	Whether this is the default view	<i>yesNoString</i>	No	"No"	<ul style="list-style-type: none"> • Yes • No 	Yes
zoom	Zoom factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected area color	<i>saColorString</i>	No	"Yellow"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom" 	Yes

Table 153: Attributes for the logView object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
bgColor	Background color	<i>bgColorString</i>	No	"White"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Green Bar" • "Custom" 	Yes
headingColor	Heading color	<i>hdgColorString</i>	No	"BG Color"	<ul style="list-style-type: none"> • "BG Color" • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom" 	Yes
textFidelity	Text fidelity	<i>fidelityString</i>	No	"Line"	<ul style="list-style-type: none"> • "Line" • "Word" • "Char" • "Draft" • "240-pel" 	Yes
overstrike	Text overstrike option	<i>overstrikeString</i>	No	"Separate Lines"	<ul style="list-style-type: none"> • "Separate Lines" • "Merge Lines" • "Overstrike Lines" 	Yes
fieldOrder	A string showing the order in which the fields are to be displayed	String	No	""		Yes
lockHeader	Whether the header lines should be locked in place during scrolling	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Table 153: Attributes for the logView object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
lockField	Number of fields that you want to lock in place	Integer	No	0	Nonnegative integer	Yes

Table 154: Attributes for the afpData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
orientation	The orientation or rotation angle, in degrees, of the document	<i>orientString</i>	No	0	<ul style="list-style-type: none"> • "0" • "90" • "180" • "270" 	Yes
paperSize	The output paper size	<i>paperString</i>	No	"Autosize"	See the list that follows this table for possible values for the paperSize attribute.	Yes

The following are possible values for the paperSize attribute:

- A3
- A4 Wide
- A4
- A5
- Autosize
- B4 Wide
- B4
- B5
- Cut Sheet
- Euro Fanfold
- Executive 1
- Executive 2
- Executive 3
- Folio
- Ledger
- Legal Wide
- Legal
- Letter Wide
- Letter
- Narrow
- Quarto
- Short
- Statement

- Stationery
- Tabloid
- Wide
- 1403W
- 1403WS
- 3800N
- 3800NS
- 3800W
- 3800WS

Table 155: Child objects under the *afpData* object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	20	Yes	Yes

Table 156: Attributes for the *logView* object under the *afpData* object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • delete • "update" 	N/A
name	Name of this logical view	<i>nameString</i>	Yes			Yes
user	User for this logical view (or PUBLIC)	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • User name • "*PUBLIC" 	No
defaultView	Whether this is the default view	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • Yes • No 	Yes
zoom	Zoom factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected area color	<i>saColorString</i>	No	"Yellow"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom" 	Yes
bgColor	Background color	<i>bgColorString</i>	No	"White"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Green Bar" • "Custom" 	Yes

Table 156: Attributes for the logView object under the afpData object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
imageColor	Image color	imgColorString	No	"Black"	<ul style="list-style-type: none"> • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Magenta" • "Cyan" • "Custom" 	Yes
textFidelity	Text fidelity	fidelityString	No	"Line"	<ul style="list-style-type: none"> • "Line" • "Word" • "Char" • "Draft" • "240-pel" 	Yes
imageIntensity	Image intensity	intensityString	No	"Normal"	<ul style="list-style-type: none"> • "Normal" • "Light" • "None" 	Yes
copyGroup	The copy group for the document	Integer	No	1	Nonnegative integer	Yes

Table 157: Attributes for the pdfData object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
orientation	The orientation or rotation angle, in degrees, of the document	orientString	No	0	<ul style="list-style-type: none"> • 0 • 90 • 180 • 270 	Yes

Table 158: Child objects under the pdfData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	Unbounded	Yes	Yes

Table 159: Attributes for the logView object under pdfData

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	taskString	No	"add"	<ul style="list-style-type: none"> • "add" • delete • "update" 	N/A
name	Name of this logical view	nameString	Yes			Yes
user	User for this logical view (or PUBLIC)	nameString	Yes		<ul style="list-style-type: none"> • User name • "*PUBLIC" 	No

Table 159: Attributes for the logView object under pdfData (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
defaultView	Whether this is the default view	yesnoString	No	"No"	<ul style="list-style-type: none"> • Yes • No 	Yes
zoom	Zoom factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected area color	saColorString	No	"Yellow"	<ul style="list-style-type: none"> • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom" 	Yes

Table 160: Attributes for the imageData object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
orientation	The orientation or rotation angle, in degrees, of the document	orientString	No	0	<ul style="list-style-type: none"> • 0 • 90 • 180 • 270 	Yes

Table 161: Child objects under the imageData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	20	Yes	Yes

Table 162: Attributes for the logView object under imageData

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	taskString	No	"add"	<ul style="list-style-type: none"> • "add" • delete • "update" 	N/A
name	Name of this logical view	nameString	Yes			Yes
user	User for this logical view (or PUBLIC)	nameString	Yes		<ul style="list-style-type: none"> • User name • "**PUBLIC" 	No
defaultView	Whether this is the default view	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
zoom	Zoom factor	Integer	No	100	Integer value 10 – 1000	Yes
contrast	Image contrast value	Integer	No	128	Integer value 0–255	Yes

Table 162: Attributes for the logView object under imageData (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
brightness	Image brightness value	Integer	No	128	Integer value 0–255	Yes
greyScale	Whether the image should use grey scales	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 163: Attributes for the udData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
fileExt	File extension	fileExtString	Yes			Yes

Table 164: Attributes for the indexParm object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	taskString	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	n/a
<character data>	The character data that represents the index parameters for this application	string	Yes			Yes

Important: The index parameters are not represented by any attribute data. They are represented by the character data field between the <indexParm> and </indexParm> fields.

Table 165: Attributes for the preprocessParm object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
dbName	Application group database name	dbString	Yes			No
loadIDName	Load ID name	namestring	No	dbName		Yes
defaultValue	The default value	string	No	Based on data type		Yes
strip	The embedded characters to remove	string	No	Based on data type		Yes
leading	The leading characters to remove	string	No	""		Yes
trailing	The trailing characters to remove	string	No	""		Yes
divideBy	The value to divide by if the field is numeric	Integer	No	1 or 100 based on data type	Positive integer	Yes
format	The date or time format to use	string	No	""		Yes

Table 166: Attributes for the imageOverlay object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" 	N/A
id	The name of the image overlay that is added to the application	string	Yes			No
fileName	The full path name of the DLL file that contains the image overlay bitmap that is added to the application	string	Yes			No

Remember: The "_ALL" value can only be used in an ARSXML export operation.

Folder object

Table 167: Attributes for the folder object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the folder	<i>nameString</i>	Yes		<ul style="list-style-type: none"> Folder name "_ALL" 	Yes
description	A description of the folder	<i>descString</i>	No	""		Yes
searchType	The note search type	<i>searchTypeString</i>	No	"Retrieve"	<ul style="list-style-type: none"> "Retrieve" "Hit List" "Note" 	Yes
displayDocHold	Whether the client displays the hold status for each document	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
displayDocLocation	Whether the client should show the storage location of each document in the document list	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
queryUnmappedFields	Whether this user or group should be allowed to add CFS-CMOD documents to this application group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
applQueryName	Name of the application that has all the folders you want to export. When you specify applQueryName during export, the value for the name attribute is ignored. If you specify applQueryName, you cannot specify agQueryName.	string	No	""		No

Table 167: Attributes for the folder object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
agQueryName	Name of the application group that has all the folders you want to export. When you specify agQueryName during export, the value for the identifier attributes is ignored. If you specify agQueryName, you cannot specify applQueryName.	string	No	""		No

Table 168: Child objects under the folder object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes
applicationGroup	1	Unbounded	Yes	No
field	1	128	No	Yes
nls	0	26	Yes	Yes
folderNls	0	26	Yes	Yes

Table 169: Attributes for the permission object under the folder object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	taskString	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
user	Provide permission to the folder of this user	nameString	Yes	""	<ul style="list-style-type: none"> User name 	No
group	Provide permission to the folder of this group	nameString	Yes	""	<ul style="list-style-type: none"> Group name "*PUBLIC" 	No
adminAuthority	Whether the user or group should have administrator authority	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
accessAuthority	Whether the user or group should have access authority	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
fieldsAuthority	Whether the user or group should have authority to change fields	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
publicNQAuthority	Whether the user or group should have public named query authority	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
privateNQAuthority	Whether the user or group should have private named query authority	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
viewNQAuthority	Whether the user or group should have named query view authority	yesnoString	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 169: Attributes for the permission object under the folder object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
maxHits	The maximum number of hits allowed	<i>maxHitsType</i>	No	"No Limit"	<ul style="list-style-type: none"> "No Limit" "None" A positive integer value 	Yes
secondaryFolder	Whether this is a secondary folder	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
userGroupFields	Whether user/group fields should be created for this user/group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
fullReportBrowse	Whether full report browse should be allowed for this user/group	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
serverBasedSorting	Whether sorting should be done on the server	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Important: You must specify either a user or a group object, however, you cannot specify both.

Table 170: Attributes for the applicationGroup object under the folder object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" 	N/A
name	Name of this object	<i>nameString</i>	Yes		Application group name	No

Table 171: Child objects under the applicationGroup object under the folder object

Name	Minimum number	Maximum number	Deleteable	Updateable
application	0	Unbounded	Yes	No

Table 172: Attributes for the application object under the applicationGroup object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" 	N/A
name	Name of this object	<i>nameString</i>	Yes		Application name	No

Table 173: Attributes for the field object under the folder object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
name	Name of the folder field	<i>nameString</i>	Yes			Yes

Table 173: Attributes for the field object under the folder object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
description	A description of the folder field	<i>descString</i>	No	""		Yes
fieldType	Field type of the folder object	<i>fieldString</i>	No	"String"	<ul style="list-style-type: none"> • "Ann Color Search" • "Ann Text Search" • "Application Group" • "Big Int" • "Date" • "Date (native)" • "Date/Time" • "Date/Time (native)" • "Date/Time (TZ)" • "Date/Time (TZ) (native)" • "DecFloat (16)" • "DecFloat (34)" • "Decimal" • "Document Type" • "FTI Highlight" • "FTI Score" • "FTI Search" • "FTI Summary" • "Hold" • "Integer" • "Segment" • "Small Int" • "String" • "Text Search" • "Time" 	No
mappingType	Mapping type of the folder field	<i>mappingString</i>	No	"Single"	<ul style="list-style-type: none"> • "Single" • "Range" • "Operator Or" 	No
applicationGroup	The Application Group to use for a segment field	<i>nameString</i>	No	""	""	Yes

Table 174: Child objects under the field object under the folder object

Name	Minimum number	Maximum number	Deleteable	Updateable
mapping	0	Unbounded	Yes	No
fieldInfo	1	Unbounded	Yes	Yes

Table 175: Attributes for the mapping object under the field object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" 	N/A

Table 175: Attributes for the mapping object under the field object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
dbName	The database field name	<i>nameString</i>	Yes			No
appGroup	The application group name	<i>nameString</i>	Yes			No

Table 176: Attributes for the fieldInfo object under the field object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" • "update" 	N/A
user	The user that is associated with this field definition	<i>nameString</i>	Yes		User name	Yes
group	The group that is associated with this field definition	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • Group name • "**PUBLIC" 	Yes
defaultValue	The default value	string	No			Yes
defaultValue2	The second default value	string	No			Yes
displayOrder	The order of this field within the hit list	Integer	No	1	0 – Number of fields	Yes
queryOrder	The order of this field during a query	Integer	No	1	0 – Number of fields	Yes
sortOrder	The sort position of this field, if any	Integer	No	0	0 – Number of fields	Yes
sortType	Whether the documents should be sorted in an ascending or descending order	<i>sortTypeString</i>	No	"Ascending"	<ul style="list-style-type: none"> • "Ascending" • "Descending" 	Yes
equal	Whether the equal operator should be available	<i>compString</i>	No	"Default"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
notEqual	Whether the not equal operator should be available	<i>compString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
lessThan	Whether the less than operator should be available	<i>compString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
ItOrEqual	Whether the less than or equal operator should be available	<i>compString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes

Table 176: Attributes for the fieldInfo object under the field object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
greaterThan	Whether the greater than operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
gtOrEqual	Whether the greater than or equal operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
in	Whether the in operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
notIn	Whether the not in operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
between	Whether the between operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
notBetween	Whether the not between operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
like	Whether the like operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
notLike	Whether the not like operator should be available	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" • "Default" 	Yes
default	Determines whether the field contains a default value when the user opens the folder	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
fixed	Determines whether the value that you type in the default fields can be changed by the user	compString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
required	Determines whether the user must specify a search value for the field in order to type a query	yesnoString	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes

Table 176: Attributes for the fieldInfo object under the field object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
viewTitle	Determines whether OnDemand client programs display the field name on the title bar of the viewing window when the user selects a document for viewing	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
internal	Determines whether OnDemand lists the displayed values or the database values for the folder search field	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
wildCard	Determines whether OnDemand uses wildcards with field search values	<i>wildString</i>	No	"Append"	<ul style="list-style-type: none"> "Prepend" "Append" "Both" "None" 	Yes
min	For integer, DecFloat(16), DecFloat(34), decimal, date, and time fields, determines the minimum value that the user can type in the search field	Integer	No	Based on datatype		Yes
max	For integer, DecFloat(16), DecFloat(34), decimal, date, and time fields, determines the maximum value that the user can type in the search field	Integer	No	Based on datatype		Yes
decimalPrecision	Specifies the decimal precision for decimal values	Integer	No	2	Nonnegative integer	Yes
dateDisplayFormat	Determines the format that is used to display dates and times	string	No	""	Date or time format	Yes
dateDefaultFormat	Determines the format that OnDemand uses to validate date and time values	string	No	""	Date or time format	Yes
dateInterval	For date and time fields, determines whether the interval refers to the next or last interval	<i>intervalString</i>	No	"Last"	<ul style="list-style-type: none"> "Last" "Next" 	Yes

Table 176: Attributes for the fieldInfo object under the field object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
dateIntLength	For date and time fields, determines the number of units that OnDemand uses to compute a default search range	integer	No	0	Nonnegative integer	Yes
dateIntType	Units of measure for dateIntLength	intTypeString	No	"Days"	<ul style="list-style-type: none"> • "Days" • "Months" • "Years" • "Hours" • "Minutes" • "Seconds" 	Yes

Table 177: Attributes for the nls object under the folder object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object.	taskString	No	"add"	<ul style="list-style-type: none"> • "add" • "update" 	No
language	The language employed by this object.	langString	Yes		<ul style="list-style-type: none"> • "English" • "French" • "German" • "Italian" • "Japanese" • "Norwegian" • "Portuguese (Brazil)" • "Spanish" • "Chinese (Simplified)" • "Chinese (Traditional)" • "Danish" • "Finnish" • "Swedish" • "Korean" • "Dutch" • "Arabic" • "Czech" • "Greek" • "Hebrew" • "Croatian" • "Hungarian" • "Polish" • "Russian" • "Slovakian" • "Slovenian" • "Turkish" 	No

Table 178: Child objects under the nls object under the folder object

Name	Minimum number	Maximum number	Deleteable	Updateable
field	0	128	No	Yes

Table 179: Attributes for the field object under the nls object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the folder field.	<i>stringLeng1-60</i>	Yes			No
langName	Name of the folder field written in the NLS language.	<i>stringLeng1-60</i>	Yes			Yes
langDescription	Description of the folder field written in the NLS language.	<i>stringLeng1-120</i>	No			Yes

Table 180: Attributes for the folderNls object under the folder object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object.	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" • "update" 	N/A
language	The language employed by this object.	<i>langString</i>	Yes		<ul style="list-style-type: none"> • "English" • "French" • "German" • "Italian" • "Japanese" • "Norwegian" • "Portuguese (Brazil)" • "Spanish" • "Chinese (Simplified)" • "Chinese (Traditional)" • "Danish" • "Finnish" • "Swedish" • "Korean" • "Dutch" • "Arabic" • "Czech" • "Greek" • "Hebrew" • "Croatian" • "Hungarian" • "Polish" • "Russian" • "Slovakian" • "Slovenian" • "Turkish" 	No

Table 180: Attributes for the folderNls object under the folder object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
langName	Name of the folder written in the NLS language.	<i>stringLeng1-60</i>	Yes			Yes
langDescription	Description of the folder written in the NLS language.	<i>stringLeng1-120</i>	No			Yes

Note:

1. You must specify either a user or a group, however, you cannot specify both.
2. The default values for displayOrder and queryOrder are based on their position within the file.

Remember: The "_ALL" value can only be used in an ARSXML export operation.

Cabinet object

Table 181: Attributes for the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the cabinet	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • Cabinet name • "_ALL" 	Yes
description	A description of the cabinet	<i>descString</i>	No	""		Yes

Table 182: Child objects under the cabinet object

Name	Minimum number	Maximum number	Deleteable	Updateable
folder	0	Unbounded	Yes	No
permission	0	Unbounded	Yes	Yes
cabinetNls	0	26	Yes	Yes

Table 183: Attributes for the folder object under the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the folder	<i>nameString</i>	Yes			No
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "delete" 	N/A

Table 184: Attributes for the permission object under the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
user	Name of the user	<i>nameString</i>	No		<ul style="list-style-type: none"> • User name 	No

Table 184: Attributes for the permission object under the cabinet object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
group	Name of the group	<i>nameString</i>	No		<ul style="list-style-type: none"> Group name "*PUBLIC" 	No
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A
accessAuthority	Whether this user or group has access authority	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
adminAuthority	Whether this user or group has administrative authority	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 185: Attributes for the cabinetNls object under the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object.	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A

Table 185: Attributes for the cabinetNls object under the cabinet object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
language	The language employed by this object.	<i>langString</i>	Yes		<ul style="list-style-type: none"> • "English" • "French" • "German" • "Italian" • "Japanese" • "Norwegian" • "Portuguese (Brazil)" • "Spanish" • "Chinese (Simplified)" • "Chinese (Traditional)" • "Danish" • "Finnish" • "Swedish" • "Korean" • "Dutch" • "Arabic" • "Czech" • "Greek" • "Hebrew" • "Croatian" • "Hungarian" • "Polish" • "Russian" • "Slovakian" • "Slovenian" • "Turkish" 	No
langName	Name of the cabinet written in the NLS language.	<i>stringLeng1-60</i>	Yes			Yes
langDescription	Description of the cabinet written in the NLS language.	<i>stringLeng1-120</i>	No			Yes

Remember: The "_ALL" value can only be used in an ARSXML export operation.

Hold object

Table 186: Attributes for the hold object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the hold	<i>nameString</i>	Yes		<ul style="list-style-type: none"> • Hold name • "_ALL" 	Yes
description	A description of the hold	<i>descString</i>	No	""		Yes

Table 187: Child objects under the hold object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes
holdNls	0	26	Yes	Yes

Table 188: Attributes for the permission object under the hold object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
user	Name of a user that can view and maintain this hold	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> User name 	No
group	Name of the group that can view and maintain this hold	<i>nameString</i>	Yes	""	<ul style="list-style-type: none"> Group name 	No
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	Yes
adminAuthority	Whether this user or group has administrative authority	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
accessAuthority	Whether this user or group has access authority	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> "Yes" "No" 	Yes

Table 189: Attributes for the holdNls object under the hold object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object.	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" "update" 	N/A

Table 189: Attributes for the holdNls object under the hold object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
language	The language employed by this object.	<i>langString</i>	Yes		<ul style="list-style-type: none"> • "English" • "French" • "German" • "Italian" • "Japanese" • "Norwegian" • "Portuguese (Brazil)" • "Spanish" • "Chinese (Simplified)" • "Chinese (Traditional)" • "Danish" • "Finnish" • "Swedish" • "Korean" • "Dutch" • "Arabic" • "Czech" • "Greek" • "Hebrew" • "Croatian" • "Hungarian" • "Polish" • "Russian" • "Slovakian" • "Slovenian" • "Turkish" 	No
langName	Name of the hold written in the NLS language.	<i>stringLeng1-60</i>	Yes			Yes
langDescription	Description of the hold written in the NLS language.	<i>stringLeng1-120</i>	No			Yes

Remember: The "_ALL" value can only be used in an ARSXML export operation.

ODF recipient object

Table 190: Attributes for the odfRecipient object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the recipient	<i>stringLen1-60</i>	Yes			No, see newName
newName	New value for name of the recipient	<i>stringLen1-60</i>	No			Yes
description	Description of the recipient	<i>stringLen120</i>	No			Yes

Table 190: Attributes for the *odfRecipient* object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
fullName	Full name of the recipient	<i>stringLen60</i>	No			Yes
title	Title of the recipient	<i>stringLen60</i>	No			Yes
email	E-mail address of the recipient	<i>stringLen254</i>	No			Yes
addr1	Address line 1 of the recipient	<i>stringLen60</i>	No			Yes
addr2	Address line 2 of the recipient	<i>stringLen60</i>	No			Yes
addr3	Address line 3 of the recipient	<i>stringLen60</i>	No			Yes
addr4	Address line 4 of the recipient	<i>stringLen60</i>	No			Yes
dept	Department	<i>stringLen60</i>	No			Yes
building	Building	<i>stringLen60</i>	No			Yes
account	Account information	<i>stringLen60</i>	No			Yes
room	Room	<i>stringLen60</i>	No			Yes
banner	Banner page?	<i>boolean</i>	No	"false"	<ul style="list-style-type: none"> • "true" • "false" 	Yes
header1	Header line 1	<i>stringLen60</i>	No			Yes
header2	Header line 2	<i>stringLen60</i>	No			Yes
header3	Header line 3	<i>stringLen60</i>	No			Yes
header4	Header line 4	<i>stringLen60</i>	No			Yes
header5	Header line 5	<i>stringLen60</i>	No			Yes
header6	Header line 6	<i>stringLen60</i>	No			Yes
header7	Header line 7	<i>stringLen60</i>	No			Yes
header8	Header line 8	<i>stringLen60</i>	No			Yes

ODF recipient list object

Table 191: Attributes for the *odfRecipientList* object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the recipient list	<i>stringLen1-60</i>	Yes			No, see <i>newName</i>
newName	New name of the recipient list	<i>stringLen1-60</i>	No			Yes

Table 191: Attributes for the *odfRecipientList* object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
description	A description of the recipient list	<i>stringLen120</i>	No			Yes

Table 192: Child objects under the *odfRecipientList* object

Name	Minimum number	Maximum number	Deleteable	Updateable
<i>odfRecipient</i>	0	Unbounded	Yes	No

Table 193: Attributes for the *odfRecipient* object under the *odfRecipientList* object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskStringA D</i>	No	"add"	<ul style="list-style-type: none"> "add" "delete" 	N/A
name	The name of a recipient that is a member of this recipient list	<i>stringLen1-60</i>	Yes			No

ODF report ID object

Table 194: Attributes for the *odfReportId* object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the report ID	<i>stringLen1-60</i>	Yes			No
applicationGroup	Application group associated with this report ID	<i>stringLen1-60</i>	No			No
application	Application associated with this report ID	<i>stringLen1-60</i>	No			No
status	Status	<i>statusString</i>	No	"Active"	<ul style="list-style-type: none"> "Active" "Inactive" 	Yes
reference	Reference string	<i>stringLen0-254</i>	No			Yes

ODF distribution object with report bundle object

Table 195: Attributes for the *odfDistribution* object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
name	Name of the distribution	<i>stringLen1-60</i>	Yes			No
recipient	Recipient for the distribution	<i>stringLen1-60</i>	Yes			No

Table 195: Attributes for the odfDistribution object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
status	Status	<i>statusString</i>	No	"Active"	<ul style="list-style-type: none"> "Active" "Inactive" 	Yes
location	Location	<i>locationString</i>	No	"Print"	<ul style="list-style-type: none"> "Print" "File" "Print Notify" "E-mail" "File Notify" "None" 	Yes
transform	Transform	<i>transformString</i>	No	"None"	<ul style="list-style-type: none"> "None" "UTF8" "PDF" 	Yes
fileNameLocation	Filename location	<i>stringLen254</i>	No			
exitInfo	Exit program information	<i>stringLen400</i>	No			Yes
custVariables	Customer variables	<i>stringLen55</i>	No			Yes
account	Account	<i>stringLen60</i>	No			Yes
cwIndicator	Continue or wait indicator	<i>cwIndString</i>	No		<ul style="list-style-type: none"> "Continue" "Wait" 	Yes
maxTries	Maximum number of attempts to continue processing the distribution	<i>integer0-1440</i>	No			Yes
manifest	Create manifest	<i>manifestString</i>	No	"Manifest"	<ul style="list-style-type: none"> "Manifest in JES spool file dataset" "No Manifest" "Manifest" 	Yes
reportBreak	Report break	<i>yesnoString</i>	No	"Yes"	<ul style="list-style-type: none"> "Yes" "No" 	Yes
distMethod	Distribution method	<i>distMethod String</i>	No	"Loaded"	<ul style="list-style-type: none"> "Loaded" "External" "All Ready" "Time of Print" "Time of Day" 	Yes
time	Time associated with distribution method	<i>stringLen5</i>	No			Yes
destination	Destination	<i>stringLen19</i>	No			Yes
writer	Writer name	<i>stringLen8</i>	No			Yes
forms	Forms	<i>stringLen8</i>	No			Yes
formDef	Form definition name	<i>stringLen8</i>	No			Yes
pageDef	Page definition name	<i>stringLen8</i>	No			Yes
job	Job name	<i>stringLen8</i>	No			Yes
class	Class	<i>stringLen1</i>	No			Yes

Table 195: Attributes for the *odfDistribution* object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
FCB	FCB	<i>stringLen4</i>	No			Yes
UCS	UCS	<i>stringLen4</i>	No			Yes
outputLimit	Maximum number of lines of output	<i>outputLimitValue</i>	No	"Unlimited"	<ul style="list-style-type: none"> "1" to "1677215" "Unlimited" 	Yes

Table 196: Child objects under the *odfDistribution* object

Name	Minimum number	Maximum number	Deleteable	Updateable
odfReportBundle	0	Unbounded	Yes	Yes

Table 197: Attributes for the *odfReportBundle* object under the *odfDistribution* object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> "add" "update" "delete" 	N/A
reportId	Report ID	<i>stringLen1-60</i>	Yes			Yes
sequence	The sequence number of this report bundle within the distribution	<i>positiveInteger</i>	Yes			No, see new sequence
newSequence	New value for sequence number	<i>positiveInteger</i>	No			Yes
reportBuild	Report build	<i>rptBuildString</i>	No	"Full"	<ul style="list-style-type: none"> "Full" "Query" 	Yes
status	Status	<i>statusString</i>	No	"Active"	<ul style="list-style-type: none"> "Active" "Inactive" 	Yes
waitIndicator	Wait or ignore indicator	<i>wiIndString</i>	No	"Wait"	<ul style="list-style-type: none"> "Wait" Do Not Wait" 	Yes
location	Location	<i>locationStringBundle</i>	No		<ul style="list-style-type: none"> "Print" "File" "Print Notify" "E-mail" "File Notify" "None" 	Yes
transform	Transform	<i>transformStringBundle</i>	No	"None"	<ul style="list-style-type: none"> "None" "UTF8" "PDF" 	Yes
fileNameLocation	FileName location	<i>stringLen254</i>	No			
exitInfo	Exit program information	<i>stringLen400</i>	No			Yes
custVariables	Customer variables	<i>stringLen55</i>	No			Yes
destination	Destination	<i>stringLen19</i>	No			Yes
writer	Writer name	<i>stringLen8</i>	No			Yes

Table 197: Attributes for the *odfReportBundle* object under the *odfDistribution* object (continued)

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
forms	Forms	<i>stringLen8</i>	No			Yes
formDef	Form definition name	<i>stringLen8</i>	No			Yes
pageDef	Page definition name	<i>stringLen8</i>	No			Yes
job	Job name	<i>stringLen8</i>	No			Yes
class	Class	<i>stringLen1</i>	No			Yes
FCB	FCB	<i>stringLen4</i>	No			Yes
UCS	UCS	<i>stringLen4</i>	No			Yes
burst	Burst	<i>yesnoString</i>	No	"No"	<ul style="list-style-type: none"> • "Yes" • "No" 	Yes
flash	Flash	<i>stringLen6</i>	No			Yes
copies	Number of copies	<i>integer1-255</i>	No	"1"		Yes
COPYM	COPYM	<i>stringLen8</i>	No			Yes
outputLimit	Maximum number of lines of output	<i>outputLimitValue</i>	No	"Unlimited"		Yes

Table 198: Child objects under the *odfReportBundle* object

Name	Minimum number	Maximum number	Deleteable	Updateable
sql	0	1	Yes	Yes

Table 199: Attributes for the *sql* object under the *odfReportBundle* object

Attributes	Description	Data Type	Required	Default value	Possible values	Updateable
task	The task to perform on this object	<i>taskString</i>	No	"add"	<ul style="list-style-type: none"> • "add" • "update" • "delete" 	N/A
character data	The character data that represents the SQL string to select records for this report bundle	<i>stringLen30000</i>	No			Yes

Data field limitations

Table 200: Data field limitations

Datatype	Minimum length	Maximum length
dbString	1	18
dbString2	1	254
defString	0	254
descString	0	120

Table 200: Data field limitations (continued)

Datatype	Minimum length	Maximum length
dvString	1	254
emailString	0	254
fileExtString	1	20
formString	1	20
indexParm	0	32,700
nameString	1	60
nameStringUG (Users and Groups)	1	128
passwordString	0	128
phoneString	0	32
postProcString	0	1000
queueString	1	60
routingString	0	24
valString	0	8
userMiscString	0	60

Troubleshooting

Troubleshooting scenario 1: Unable to combine two applications in one application group

Symptom

While attempting to have more than one application in an application group, I received this error:

```
arsload: Processing file /ars1/ATEST.TST.TST11.SYS001.1285.021030.075349.ARD<
The application >< does not exist within the application group >ATEST<
```

Possible causes

The -A parameter is not specified in the ARSLOAD program. If you are loading from the JES SPOOL, ARSLOAD has two parameters: -A and -G. You do not need to specify -A if you only have one application in the application group, because ARSLOAD uses that one application for the load. However, if you have more than one application in the application group, then the -A parameter must be specified. As an alternative to specifying -A, you might use the ARS.RSADUPDT exit.

For example, if an IEBGENER job that sends the report to the spool specifies

```
//OUTPUT1 OUTPUT WRITER=TESTAPP1,FORMS=TESTAPG
```

then the corresponding -G and -A parameters on ARSLOAD is

```
-G FORMS -A WRITER
```

In this case, ARSLOAD uses the TESTAPG application group and the TESTAPP1 application.

Action

If multiple applications share the same indexing and storage management attributes, there are advantages in including them in the same application group. Combining the applications saves you from having to create and maintain a separate, identical application group for each of those applications. For example, if you have two different reports and want to index both of them on the same values and store both of them for five years, consider creating the corresponding applications for those reports in the same application group. For more information, refer to [Chapter 24, “ARSLOAD,”](#) on page 185. There might be potential performance implications caused by combining more than one application in an application group.

Troubleshooting scenario 2: ARSLOAD error

Symptom

While attempting to load a report from spool with the following JCL:

```
//ARSLOAD EXEC PGM=ARSLOAD,REGION=0M,  
// PARM=('/' -A JOBNAME -uadmin -ppassword -v -C F  
// -g "ClaseU" -f -c /u/u1/ctemp ')  
//STEPLIB DD DISP=SHR,DSN=ARS.V710.SARSLOAD  
// DD DISP=SHR,DSN=SYS1.DSN610M.SDSNEXIT  
// DD DISP=SHR,DSN=DSN610M.SDSNLOAD  
// DD DISP=SHR,DSN=APK.ACIF.ONDE.SAPKMOD1  
//SYSPRINT DD SYSOUT=*  
//SYSOUT DD SYSOUT=*
```

, you received this error from ARSLOAD:

The application >< does not exist within the application group >ClaseU<

Possible causes

If you use -g (lowercase g) to specify an application group, ARSLOAD is unable to determine the application or application group from the spool file. In other words, -G and -A are ignored if you specify -g. As a result, ARSLOAD tries to do a load to application group ClaseU, with no application specified, and it fails.

Action

Important: This solution requires you to define each job name as an application.

Use -G WRITER -W CLASEU -A JOBNAME, and have the jobs specify a writer name of CLASEU.

Example:

```
PARM=('/'-G WRITER -W CLASEU -A JOBNAME -v -f')
```

Troubleshooting scenario 3: ARSLOAD not loading data from HFS directory specified with -d parameter

Symptom

You attempted to use ARSLOAD to load from an HFS directory rather than from spool. I specified the following for my ARSLOAD PARM statement:

```
PARM=(' -h ARCHIVE -C A -d /ars/upload')
```

However, ARSLOAD does not load the files from your /ars/upload directory.

Possible causes

The -d and -C parameters are mutually exclusive. If the -C parameter is specified, ARSLOAD will load from spool and will not monitor the directory specified by the -d parameter.

Action

Remove the -C A from the PARM statement. The PARM statement should instead look like this:

```
PARM=(' -h ARCHIVE -d /ars/upload')
```

Troubleshooting scenario 4: ARSLOAD failed while specifying the host with the -h parameter

Symptom

ARSLOAD failed when you attempted to specify the host with the -h parameter.

Possible causes

This scenario discusses some of the issues that arise in using the -h parameter with ARSLOAD, and provides some guidance about how to minimize problems that can occur with ARSLOAD in conjunction with the -h parameter. It is not intended to describe all the ways that `ars.ini` and `ars.cfg` can be configured, nor is it a reference document.

When the -h host option is specified, ARSLOAD uses the string host to look for a matching instance entry in `/opt/IBM/ondemand/V10.1/config/ars.ini`. The entry is delimited by `[@SRV@_instance]`. The host is NOT compared with the `HOST=` in the `ars.ini`. It is only compared with the instance in `[@SRV@_instance]`.

If ARSLOAD fails to find a matching instance name, it will attempt to connect to host using the default port (1445). This might work, or it might produce unexpected results. Because it found no matching instance, it will not use the `PORT=` statement or process the file specified by `SRVR_OD_CFG=`.

If ARSLOAD cannot read `/opt/IBM/ondemand/V10.1/config/ars.ini`, the same unfortunate effects can occur. That is, not finding the file has the same effect as not finding the instance entry. There is no parameter to tell ARSLOAD to look in any place besides `/opt/IBM/ondemand/V10.1/config/ars.ini`.

If ARSLOAD finds a matching instance, it uses the associated `HOST=` and `PORT=` values to communicate to the server. ARSLOAD also uses the `ars.cfg` file associated with the `SRVR_OD_CFG=` entry. Of interest later is the `ARS_TMP=` entry in `ars.cfg`. If no -h host parameter is specified, ARSLOAD looks for an instance named ARCHIVE. That is why the sample `ars.ini` is shipped with an ARCHIVE instance.

Action

If you are running with a single Content Manager OnDemand instance, follow these suggestions:

- In the `ars.ini` file for the default instance (ARCHIVE), set the `HOST=` to the host running the Content Manager OnDemand server. This is the host that is running ARSSOCKD.
- If you wish to not use the default port (1445), set the `PORT=` to the desired port. Otherwise leave it as 0.
- In the `ars.cfg` file pointed to by the `SRVR_OD_CFG=`, set the `ARS_TMP=` to a valid directory to use for temporary files. This directory should have permissions 1777. It could be `/tmp`.
- Do not use the -h option in ARSLOAD. ARSLOAD uses the default (ARCHIVE), determine the correct host and port to use, and create the temporary files in the directory pointed to by `ARS_TMP`.

If you are running multiple instances, make sure each instance has a unique `HOST=` and `PORT=` pair. That is, you cannot have two ARSSOCKDs running on the same host with the same port. Instances should not share `ars.cfg` files. Unpredictable results will occur. For ARSLOAD with multiple instances, use the -h option to specify the instance name and not the host name.

Troubleshooting scenario 5: Error message EDC5084I

Symptom

You received

```
EDC5084I The program was not run because of redirection errors on the command line
```

with ARSLOAD.

Possible causes

ARSLOAD uses the HFS for temporary files. Problems occur if ARSLOAD cannot use a writable directory. ARSLOAD currently uses the following environment variables (in order) to use as the temporary directory: *ARS_TMP*, *TEMP*, and *TMP*. The first one found is used. If none of these variables is set, ARSLOAD uses the current working directory (as returned by `getcwd()`). If the directory is not writable, ARSLOAD will fail. One symptom is message

```
EDC5084I The program was not run because of redirection errors on the command line
```

This error occurs because ARSLOAD tries to run ARSADMIN, and uses redirection to a file in the temporary directory. The C runtime attempts to create the file and cannot because the directory is not writable.

Use the `-c idx_dir` option to specify the directory to use for some, but not all, of the temporary files. For the rest of the files, it uses the above described environment variable search.

One possible cause of the temporary directory not being set correctly is specifying a host on the `-h` option of ARSLOAD that is not an instance name. As a result, the `ars.cfg` file does not get processed, which causes *ARS_TMP* to not get set. If the *TEMP* and *TMP* environment variables are not set, the current working directory will be used. Determining which directories are being used for the temporary files can be problematic.

Action

In the `ars.cfg` file pointed to by the `SRVR_OD_CFG=` entry in `/opt/IBM/ondemand/V10.1/config/ars.ini`, set the `ARS_TMP=` to a valid directory to use for temporary files. This directory should have permissions `1777`. It could be `/tmp`.

Troubleshooting scenario 6: ARSMAINT fails with ARS0020E

Symptom

ARSMAINT receives this error message:

```
ARS0020E ARSMAINT SM ERROR: ARSMVSDW: 0000000B(0000000C-7404FCDB)
PROD.ONDEMAND.DATA(BAA.R11), RC=11, REASON=0, FILE =ARSSMSMS.C, LINE=304
SRVR->SERVER.FQDN.COM 10.10.1.2<-
```

Possible causes

When `RC=11` is present in the `ARS0020E` message, the `0000000C-7412FCDB` corresponds to the OAM return code-reason code. In this case, the OAM return code is 12 (hexadecimal C). Byte 0 of the OAM reason code is hexadecimal 74, which indicates a DB2 error with `SQLCODE` of bytes 2 and 3 of the OAM reason code (hexadecimal `FCDB`, which is decimal `-805`). `SQLCODE = -805` indicates `DBRM or PACKAGE NAME ... NOT FOUND IN PLAN`. As an additional diagnostic aid, issuing the following TSO `OSREQ`

command using the collection and object name from the ARS0020E message should yield the same OAM return code/reason code:

```
OSREQ QUERY PROD.ONDEMAND.DATA BAA.R11 OSREQ QUERY unsuccessful.  
Return code = 0000000C, reason code = 7412FCDB.  
OSREQ QUERY response time is 53 milliseconds.
```

Action

It is likely that a bind is needed for the CBRIDBS plan. If you recently applied OAM maintenance, ensure binds indicated by hold data have been performed.

Troubleshooting scenario 7: ARSADMIN fails while attempting to run from USS

Symptom

Attempting to run `/opt/IBM/ondemand/V10.1/bin/arsadmin` from Unix System Services (USS) yields "arsadmin: FSUM7351 not found".

Possible causes

You need a link to run ARSADMIN from USS. To check whether the link exists, look for an external link to ARSADMIN from `/opt/IBM/ondemand/V10.1/bin`; it looks like this:

```
OD39090:/opt/IBM/ondemand/V10.1/bin: >ls -l arsadmin  
erwxrwxrwx 1 root root 8 Oct 15 2001 arsadmin -> ARSADMIN
```

Your SARSLOAD PDS needs to be part of STEPLIB. For example:

```
OD39090:/opt/IBM/ondemand/V10.1/bin: >echo $STEPLIB  
ARSV710.ODMP710.SARSLOAD:DB2.V10R1M0.DSNA.SDSNEXIT:DB2.V10R1M0.SDSNLOAD
```

Action

If you do not have an external link to ARSADMIN from `/opt/IBM/ondemand/V10.1/bin`, the ARSISMKD job has not been run. This job creates the links in `/opt/IBM/ondemand/V10.1/bin` for ARSLOAD and other executables, as well as directories needed by SMP. The ARSISMKD sample job is in SARSINST. Refer to the section "Allocate SMP/E Target and Distribution Libraries and Paths" in *Program Directory for IBM Content Manager OnDemand for z/OS, Version 8.5*.

Important: PTFs that change ARSISMKD or the REXX EXEC it invokes (ARSMKDIR) also have a ++HOLD ACTION, which indicates that the ARSISMKD job must be re-run.

Troubleshooting scenario 8: ARS0025E messages in ARSLOAD

Symptom

You received an error with a report that the cache is full.

Possible causes

There is not sufficient space on the `ars.cache` file systems.

To determine whether this problem is caused by insufficient space, see the cache configuration file (for example, `ars.cache`) that the `SRVR_SM_CFG` parameter in `/opt/IBM/ondemand/V10.1/config/ars.ini` points to, and check the directory or directories that are specified in `ars.ini` for available space. Use the USS command "`df -kP /ars1`" (where `/ars1` is a directory specified in `ars.cache`) to see how much Content Manager OnDemand cache space remains.

Action

Make sure that you have sufficient space on the ars.cache file systems.

Troubleshooting scenario 9: Error occurred during the CD-ROM mastering process

Symptom

You received this error message during the CD-ROM mastering process: Unable to open original file for staging.

Possible causes

It is possible that the Content Manager OnDemand client was updated on your PC, but the new master INI file was not transferred.

Action

Copy the new master INI file over to your PC.

Troubleshooting scenario 10: The .ARD file cannot be removed after the loading by using load service

Symptom

You used Content Manager OnDemand standard demon or load service to load a fully composed .ARD file. You placed the .ind, .out and .res files under the right directory. However, after loading, all of the .ind, .out and .res files were removed except the .ARD file.

Possible causes

This happened because you have indexed the data on z/OS and created the .ind, .out, .res files. As a result, ACIF does not run and the information in your index tab of application is not used, and Content Manager OnDemand is not aware of the need to clean up the .ARD.

Action

You can use generic indexer instead of ACIF, or use a fully composed AFP document, and let ACIF Windows build the .ind and .res files via the ARSLOAD process. Any .out file that is built by ACIF using the convert=yes or some other user exit causes the .ARD file to be deleted, as we only load the .out and have no further use for .ARD at this time.

Troubleshooting scenario 11: Error reading resource file while loading AFP documents with ARSLOAD

Symptom

While attempting to load AFP documents with the Content Manager OnDemand for z/OS, you receive an error while reading the resource file. The error in ARSLOAD is:

```
mresource_file = /tmp/Dec022004153833352.238939392 Unable to read from  
offset 819406 for length 8197 from the file: /tmp/Dec022
```

Possible causes

There is not enough storage available in the /tmp TFS directory.

Action

Increase the size of the /tmp directory so that enough storage is available to create a temporary file for ARSLOAD that contains the resources of the report. You can use command through OMVS or telnet: `df -kP` to display the file systems and available storage in HFS. If there is not enough space allocated to the /tmp mount point for the resource temporary file, increase the amount of space allocated to /tmp. If you use a TFS, only 1 MB is allocated to the file system by default. You might be able to use:

```
MOUNT FILESYSTEM('/TMP') TYPE(TFS) MOUNTPPOINT('/tmp') PARM('-s 10')
```

to mount /tmp with 10 MB of space, rather than accepting the default of 1 MB.

The larger /tmp allows Content Manager OnDemand to store the resource group to a temporary file and then successfully read that entire file without encountering the read error as noted in this problem.

If you prefer to not use /tmp for the resource file, apply PTF UQ94076. This PTF causes the ARSLOAD to use the `ARS_TMP` environment variable defined in `/opt/IBM/ondemand/V10.1/config/ars.cfg` to determine which HFS directory should be used to write the temporary resource file.

Troubleshooting scenario 12: While running ARSMAINT to expire cache objects, the job ends with return code 13

Symptom

You received a return code 13 while running ARSMAINT with the following parameters: `-I ARCHIVE -cdrsv -n 00 -x 00 -u userid -p password`

Possible cause

The application group permissions does not allow delete operation.

Action

ARSMAINT can encounter this error if the user does not have delete authority on the application group, or there is an error verifying the user ID/password combination.

By changing the ARSMAINT parameters to specify a particular application group (for example, adding the `-g` parameter), you can narrow down the problem to a specific application group or set of application groups. When you find a failing application group with the `-g` parameter, check the permissions for the user ID on the application groups permissions tab in the administration client. It is like that the user ID does not have delete authority.

After the user ID is given delete authority over the application group, ARSMAINT (with the `-g` specified) should then complete with return code 0. Another alternative is to check the permissions on each application group for the user ID if you cannot determine the failing application group with the `-g` parameter.

Troubleshooting scenario 13: Empty file created while using ARSLOAD to monitor an HFS directory

Symptom

While using ARSLOAD to capture data sent to an HFS directory, you created files similar to the following:

```
MVS1.MYAG.SYSUT2.MYAPP.04317.115803.ARD  
MVS1.MYAG.SYSUT2.MYAPP.04317.115803.out  
MVS1.MYAG.SYSUT2.MYAPP.04317.115803.ind
```

MYAG

Your application group name

MYAPP

Your application name

The output from ARSLOAD appears as follows:

```
arsload: Processing file >/tmp/  
MVS1.MYAG.SYSUT2.MYAPP.040317.115803.ARD<  
arsload: 03/17/04 13:28:54 -- Indexing started, --UNKNOWN-- bytes to  
process  
arsload: Generic Indexer requires the data to have already been  
indexed.  
Index the data or verify your input file and resubmit the  
job  
arsload: Output/Indexer file was not created  
arsload: 03/17/04 13:28:54 Indexing failed  
arsload: Processing failed for file >/tmp/  
MVS1.MYAG.SYSUT2.MYAPP.040317.115803.ARD<  
arsload: Unloading of data was NOT performed
```

This creates an empty file called `MVS1.MYAG.SYSUT2.MYAPP.040317.115803.ard.failed` and the load doesn't complete.

Possible cause

ARSLOAD processing of an HFS directory fails.

Action

While using the `-d` parameter with ARSLOAD, the input must already be indexed when supplied to Content Manager OnDemand, because Content Manager OnDemand does not invoke an indexer when the `-d` parameter is specified. This data should be in generic indexer format. That is, there must be an `.out` file containing the data and an `.ind` file containing the indexes in generic indexer format. There must also be an `.res` file when resources are used. Another file (a dummy file) with extension `.ARD` signifies that the files are ready to be processed by ARSLOAD and when this file is present, ARSLOAD starts the process and assumes when the `.ARD` file is present, the other files have been transferred completely to the directory.

If you named your files with extensions `.ind`, `.out` and `.ARD`, the load process should be able to continue and find the generic indexer files. The file names are the same, with the exception of the file name extensions. From the above example, the file names should be:

```
MVS1.MYAG.SYSUT2.MYAPP.04317.115803.ARD,  
MVS1.MYAG.SYSUT2.MYAPP.04317.115803.ARD.out ,and  
MVS1.MYAG.SYSUT2.MYAPP.04317.115803.ARD.IND.
```

If you used the example without the `ARD` qualifier for the generic indexer files, you also need to make an appropriate update to your `.ind` file to specify the proper file name for the `GROUP_FILENAME` parameter. You should use an absolute (from root) file name for this parameter.

The key point is having the `ARD` qualifier in all the file names. The file ending with extension `.ARD` provides the file name for the other files needed by the generic indexer by just adding the `.ind` and `.out` extensions.

Troubleshooting scenario 14: The ARSMAINT -f command produces only usage instructions

Symptom

When I enter the command `ARSMAINT -f`, it does not produce the expected alert message for the cache storage file system. It produces only the usage information for the command itself.

Possible cause

The `ARSMAINT -f` command flag requires the `-s` command.

Action

Run the command with the -s flag. The ARSMaint command sends an alert message when the cache storage file system is full. For example,

```
arsmaint -f -s
```

Troubleshooting scenario 15: DB2 reports that OnDemand database is almost full**Symptom**

You received a message indicating that the OnDemand database is approaching the maximum allowable number of objects.

Possible causes

DB2 has a maximum of 32767 and you are approaching that maximum.

Action

Create another OnDemand database by following the instructions in "Create the storage group and database" of the *Configuration Guide for z/OS*. Use the same values you used to create the original database, except specify a different name on the line `CREATE DATABASE databaseName`. Notify users that they can specify the name of this second database when they query documents.

Notices

This information was developed for products and services that are offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at www.ibm.com/privacy and IBM's Online Privacy Statement at www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM

Software Products and Software-as-a-Service Privacy Statement” at www.ibm.com/software/info/product-privacy.

Privacy policy considerations

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, See IBM’s Privacy Policy at <http://www.ibm.com/privacy> and IBM’s Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled “Cookies, Web Beacons and Other Technologies” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at <http://www.ibm.com/software/info/product-privacy>.

Index

Numerics

- 128 ApplGrp Segment Expire [6](#)
- 14 DB Info Exported [7](#)
- 164 ApplGrp Segment Maintenance [3](#)
- 166 ApplGrp Segment Export [7](#)
- 168 ApplGrp Segment Not Available [9](#)
- 87 ApplGroup Load [80](#), [99](#)
- 87 ApplGrp Load (System Migration) [7](#)

A

- accessibility [285](#)
- accessing reports [69](#)
- ACIF
 - fields
 - defining [69](#)
 - indexes
 - defining [70](#)
 - indexing parameters [69](#), [70](#)
 - resource information [70](#)
 - triggers [69](#)
- ACIF library
 - specifying in ARSLOAD job [196](#)
 - STEPLIB DD statement [196](#)
- adding
 - application groups [64](#)
 - applications [64](#)
 - folders [64](#)
 - groups [55](#)
 - primary storage nodes [60](#)
 - printers [58](#)
 - reports [64](#)
 - server printers [58](#)
 - servers [18](#)
 - storage nodes [60](#)
 - storage sets [60](#)
 - users [54](#)
 - users to a group [56](#)
- administrative client
 - About window [24](#)
 - adding servers [18](#)
 - assistants and wizards [26](#)
 - changing a password [19](#)
 - customizing [24](#)
 - displaying [24](#)
 - hardware requirements [13](#)
 - installing [13](#)
 - logging on a server [17](#)
 - passwords [19](#)
 - program properties [22](#)
 - report wizard [26](#)
 - software requirements [13](#)
 - splash screen [24](#)
 - start up parameters [22](#)
 - starting [17](#)
 - updating [15](#)
- administrative client (*continued*)
 - using [13](#), [17](#)
 - wizards and assistants [26](#)
- administrative objects
 - exporting [103](#)
 - importing [103](#)
- administrators
 - application group/folder/cabinet administrator [29](#)
 - system administrator [29](#)
 - user administrator [29](#)
- advanced system administration
 - authority [42](#)
 - native Lightweight Directory Access Protocol (LDAP)
 - support
 - Enabling LDAP authentication [51](#)
 - How Content Manager OnDemand works with LDAP [51](#)
 - LDAP server requirements [51](#)
 - overview [41](#)
 - permissions [44](#)
 - system administration [47](#)
 - user types [41](#)
- AFP data
 - converting to line data [223](#)
 - indexing information [69](#)
 - viewing information [68](#)
- annotation group permissions table (ARSANN) [292](#)
- annotation permission [36](#)
- annotations [21](#)
- Application group composite index table (ARSAGINDEX) [291](#)
- application group data table [320](#), [321](#)
- application group field alias table (ARSAGFLDALIAS) [291](#)
- application group field table (ARSAGFLD) [290](#)
- application group object [334](#)
- application group permissions table (ARSAGPERMS) [292](#)
- application group table (ARSAG) [288](#)
- application group to folder field mapping table (ARSAG2FOL) [289](#)
- application group/folder administrator [30](#)
- application group/folder/cabinet administrator [29](#)
- application groups
 - access permission [36](#)
 - adding [64](#), [66](#)
 - administrator [29](#)
 - administrator permission [36](#)
 - annotation permission [36](#)
 - closing tables [219](#)
 - collecting statistics [3](#)
 - concepts [32](#)
 - defining [64](#), [66](#), [76](#), [83](#), [91](#)
 - document permission [36](#)
 - expiration processing [6](#), [10](#), [199](#)
 - fields
 - defining [67](#)
 - FORM parameter [187](#)
 - importing migrated index data [10](#)
 - importing migrated indexes [9](#)

- application groups (*continued*)
 - index data
 - maintaining [3, 6, 7, 9](#)
 - logical views permission [36](#)
 - maintenance [3, 6, 7, 9](#)
 - messages [3, 6, 7, 9](#)
 - migrating to table spaces [219](#)
 - migration processing [7, 9, 10, 199](#)
 - overview [32](#)
 - permissions [36, 37, 67](#)
 - query restriction [36](#)
 - storage management [66](#)
 - table spaces, migrating to [219](#)
 - tables, closing [219](#)
- application object [343](#)
- application table (ARSAPP) [293](#)
- applications
 - adding [64, 67](#)
 - AFP data [68, 69](#)
 - concepts [32](#)
 - DATASET parameter [187](#)
 - defining [64, 67, 76, 84, 92](#)
 - fields
 - defining [69](#)
 - indexes
 - defining [70](#)
 - indexing parameters [69, 70](#)
 - load information [70](#)
 - miscellaneous options [71](#)
 - overview [32](#)
 - resource information [70](#)
 - triggers [69](#)
- ARS_TMP parameter [11](#)
- ARS.CFG file
 - ARS_TMP parameter [11](#)
 - database configuration [137](#)
 - importing migrated index data [11](#)
- ARS.PTGN user exit [117, 187](#)
- ARSADMIN
 - deleting a report [101](#)
 - importing migrated index data [11](#)
 - load Id [101](#)
 - load identifier [101](#)
 - migration processing [11](#)
 - reference [119](#)
 - unloading a report [101](#)
- ARSAFPD
 - AFP files [127](#)
 - commands [127](#)
 - description [127](#)
 - examples [128](#)
 - parameters [127](#)
 - purpose [127](#)
 - syntax [127](#)
- ARSAG (application group table) [288](#)
- ARSAG2FOL (application group to folder field mapping table) [289](#)
- ARSAGFLD (application group field table) [290](#)
- ARSAGFLDALIAS (application group field alias table) [291](#)
- ARSAGINDEX (application group composite index table) [291](#)
- ARSAGPERMS (application group permissions table) [292](#)
- ARSANN (annotation table) [292](#)
- ARSAPP (application table) [293](#)
- ARSAPPUSR (user logical views table) [294](#)
- ARSCAB (cabinet table) [295](#)
- ARSCAB2FOL (cabinet to folder mapping table) [295](#)
- ARSCABPERMS (cabinet permissions table) [296](#)
- ARSDATE
 - reference [131](#)
- ARSDDB
 - reference [137](#)
- ARSDFBDDT (report bundle definition table) [310](#)
- ARSDFCRT (report ID table) [311](#)
- ARSDFDCT (distribution definition table) [304](#)
- ARSDFDRT (distribution request table) [305](#)
- ARSDFDST (Document Status Table) [306](#)
- ARSDFLIS (recipient list table) [309](#)
- ARSDFPPT (print processor table) [307](#)
- ARSDFUOT (recipient/list table) [309](#)
- ARSDOC
 - reference [141](#)
- arsdoc command
 - parameters [157](#)
- ARSEXPIR
 - reference [181](#)
- ARSFOL (folder table) [297](#)
- ARSFOLFLD (folder field table) [297](#)
- ARSFOLFLDUSR (folder user fields table) [298](#)
- ARSFOLPERMS (folder permissions table) [299](#)
- ARSGROUP (group table) [300](#)
- ARSLOAD
 - reference [185](#)
 - running [88, 95](#)
- ARSLOAD (load table) [303](#)
- ARSLOAD EXEC statement
 - PARM keyword [196](#)
 - PGM keyword [196](#)
- ARSLOAD job
 - ACIF library [196](#)
 - CEEDUMP dataset [196](#)
 - DD statement for STEPLIB [196](#)
 - DSNAOINI dataset [196](#)
 - dump dataset [196](#)
 - EXEC statement for ARSLOAD [196](#)
 - input dataset [196](#)
 - JCL statements defined [196](#)
 - load library [196](#)
 - ODF datasets [196](#)
 - parameters [196](#)
 - PARM keyword [196](#)
 - SDSNEXIT dataset [196](#)
 - SDSNLOAD dataset [196](#)
 - specifying parameters [196](#)
 - STEPLIB DD statement [196](#)
- ARSMOINT
 - collecting statistics [3](#)
 - deleting imported migrated indexes [11](#)
 - expiration processing [5, 11](#)
 - messages [3, 5, 6](#)
 - migration processing [6](#)
 - reference [199](#)
- ARSNAMEQ (named query table) [303](#)
- ARSNODE (node table) [304](#)
- ARSPRT (printer table) [312](#)
- ARSPRTOPTS (printer options table) [312](#)
- ARSPRTUSR (printer user table) [313](#)
- ARSRES (resources table) [313](#)
- ARSRPT

ARSRPT (*continued*)
 parameters [207](#)
 ARSSEG (segment table) [314](#)
 ARSSET (storage set table) [315](#)
 ARSSLU
 reference [211](#)
 ARSSOCKD
 description [213](#)
 examples [214](#)
 notes [216](#)
 parameters [213](#)
 purpose [213](#)
 stopping [213](#)
 syntax [213](#)
 ARSSTASH
 reference [217](#)
 ARSSTASH commands
 examples [217](#)
 ARSSUPPORT utility [286](#)
 ARSSYS (system parameters table) [316](#)
 ARSTBLSP
 reference [219](#)
 ARSUSER (user table) [317](#)
 ARSUSRGRP (users in groups table) [319](#)
 ARSUSRGRPID (user group ID table) [320](#)
 ARSUUPD
 exit [187](#)
 report specification archive definition exit [187](#)
 user exit [187](#)
 ARSXAFP
 reference [223](#)
 ARSXLINE
 reference [227](#)
 ARSXML
 reference [231](#)
 ARSYSPIN, JES Spool Data Capture facility [241](#)
 assistants and wizards [26](#)
 authority
 create application groups [30](#)
 create cabinets [30](#)
 create folders [30](#)
 create groups [30](#)
 create holds [30](#)
 create users [30](#)

B

backup and recovery
 database [81](#), [137](#)
 buffer pools [2](#)

C

cabinet administrator [29](#)
 cabinet object [367](#)
 cabinet permissions table (ARSCABPERMS) [296](#)
 cabinet table (ARSCAB) [295](#)
 cabinet to folder mapping table (ARSCAB2FOL) [295](#)
 cabinets
 administrator [29](#)
 concepts [33](#)
 overview [33](#)
 cache storage
 cache storage (*continued*)
 expiration processing [199](#)
 migration processing [199](#)
 CEEDUMP dataset
 specifying in ARSLOAD job [196](#)
 client
 default server printer [71](#)
 code page [119](#)
 collecting statistics [3](#)
 column delimiter [121](#)
 commands
 adding documents [141](#)
 AFP data, viewing [223](#)
 ARSADMIN [119](#)
 ARSDATE [131](#)
 ARSDB [137](#)
 ARSDOC [141](#)
 ARSMaint [199](#)
 ARSRPT [207](#)
 ARSSLU [211](#)
 ARSSTASH [217](#)
 ARSTBLSP [219](#)
 ARSXAFP [223](#)
 compress [119](#)
 creating reports [207](#)
 data, loading [185](#)
 decompress [119](#)
 deleting data [199](#)
 deleting documents [141](#)
 documents, adding, deleting, getting, printing, updating [141](#)
 documents, loading [185](#)
 expiring data [199](#)
 exporting an XML file [231](#)
 getting documents [141](#)
 import [119](#)
 importing an XML file [231](#)
 loading data, documents, reports [185](#)
 printing documents [141](#)
 reports, loading [185](#)
 retrieve [119](#)
 store [119](#)
 unload [119](#)
 updating documents [141](#)
 viewing AFP data [223](#)
 XML file, exporting [231](#)
 XML file, importing [231](#)
 compression
 ARSADMIN program [119](#)
 Content Manager OnDemand
 date formats [131](#)
 internal date format [131](#)
 Content Manager OnDemand object
 end of object [108](#)
 object attributes [107](#)
 Start of Object [107](#)
 Content Manager OnDemand objects [106](#)
 Content Manager OnDemand XML files
 creating
 for delete and export [110](#)
 conventions
 group name [31](#)
 user name [29](#)
 copying items [25](#)

- create application groups authority [30](#)
- create cabinets authority [30](#)
- create folders authority [30](#)
- create groups authority [30](#)
- create users authority [30](#)
- creating an XML file [105](#)

D

data

- adding [141](#)
- capture facility, ARSYSPIN [241](#)
- deleting [4](#), [141](#), [181](#)
- deleting a report [101](#), [119](#)
- estimating storage space for [75](#)
- expiration processing [4](#), [181](#)
- image data [83](#)
- importing migrated indexes [9](#)
- JES Spool Data Capture facility, ARSYSPIN [241](#)
- load Id [101](#)
- load identifier [101](#)
- loading [75](#), [83](#), [91](#), [185](#)
- migration processing [6](#), [9](#)
- printing [141](#)
- restarting a load process [99](#)
- retrieving [119](#), [141](#)
- Spool Data Capture facility, ARSYSPIN [241](#)
- storing [119](#)
- TIFF image data [83](#)
- unloading [101](#), [119](#)
- updating [141](#)

database

- activating [137](#)
- adding index data [79](#)
- backup [81](#), [137](#)
- collecting statistics [3](#)
- column delimiter [121](#)
- concepts [1](#)
- creating [137](#)
- deleting index data [4](#), [9](#), [181](#)
- expiration processing [4](#), [9](#), [181](#), [199](#)
- exporting tables [137](#)
- fields
 - defining [67](#)
- importing migrated index data [9](#), [119](#)
- importing tables [137](#)
- maintenance [3](#)
- migration processing [6](#), [9](#), [119](#), [199](#)
- reorganizing [137](#)
- segment table [79](#)
- statistics [3](#), [137](#)
- updating [141](#)

database manager

- concepts [1](#)

database tables [287](#)

DATASET parameter

- applications [187](#)
- ARSLOAD program [187](#)

date

- Content Manager OnDemand internal format [131](#)
- formats of [131](#)
- internal format [131](#)
- obtaining Content Manager OnDemand internal format [131](#)

DB2

- ARS_TMP parameter [11](#)
- backup [137](#)
- buffer pools [2](#)
- collecting statistics [3](#), [137](#)
- concepts [1](#)
- database [1](#)
- exporting tables [137](#)
- importing migrated index data [11](#)
- importing tables [137](#)
- indexes [1](#)
- maintenance [137](#)
- starting [137](#)
- statistics [3](#)
- stopping [137](#)
- system catalog tables [2](#)
- table space backup [137](#)
- table spaces [1](#)
- tables [1](#)

DB2 tables

- ARSDFBTD (report bundle definition table) [310](#)
- ARSDFCRT (report ID table) [311](#)
- ARSDFDCT (distribution definition table) [304](#)
- ARSDFDRT (distribution request table) [305](#)
- ARSDFDST (distribution status table) [306](#)
- ARSDFLIS (recipient list table) [309](#)
- ARSDFPPT (print processor table) [307](#)
- ARSDFUOT (recipient/list table) [309](#)
- distribution definition table (ARSDFDCT) [304](#)
- distribution request table (ARSDFDRT) [305](#)
- Document Status Table (ARSDFDST) [306](#)
- print processor table (ARSDFPPT) [307](#)
- recipient list table (ARSDFLIS) [309](#)
- recipient/list table (ARSDFUOT) [309](#)
- report bundle definition table (ARSDFBTD) [310](#)
- report ID table (ARSDFCRT) [311](#)

decompress

- ARSADMIN program [119](#)

default directory

- administrative client [24](#)

defining

- application groups [64](#)
- applications [64](#)
- folders [64](#)
- groups [55](#)
- primary storage nodes [60](#)
- printers [58](#)
- reports [64](#)
- server printers [58](#)
- storage nodes [60](#)
- storage sets [60](#)
- users [54](#)

deleting a report [101](#), [119](#)

deleting index data [4](#), [9](#)

diagrams

- syntax [114](#)

disability [285](#)

distribution definition table (ARSDFDCT) [304](#)

distribution object [373](#)

distribution request table (ARSDFDRT) [305](#)

distribution status table (ARSDFDST) [306](#)

document permission [36](#)

drag and drop operations [26](#)

DSNAOINI dataset

DSNAOINI dataset (*continued*)
specifying in ARSLOAD job [196](#)
dump dataset
specifying in ARSLOAD job [196](#)

E

Enabling LDAP authentication [51](#)
estimating storage space [75](#)
examples
ARS RTP [209](#)
instance configuration [214](#)
process usage [214](#)
start the server [214](#)
stop the server [214](#)
exits
ARS.PTGN [117](#), [187](#)
ARSUUPD [187](#)
report specification archive definition [187](#)
expiration of passwords [19](#)
expiration processing [4](#), [9](#), [181](#), [199](#)
expired passwords [19](#)
expiring reports and TSM [4](#)
exporting administrative objects [103](#)
exporting items [25](#)
exporting tables [137](#)

F

fields
application group
defining [67](#)
database
defining [67](#)
defining [67](#), [72](#)
folder
defining [72](#)
indexing
defining [69](#)
fields permission [36](#)
file input
ARSLOAD program [185](#)
file size [77](#)
folder administrator [29](#)
folder field table (ARSFOLFLD) [297](#)
folder object [358](#)
folder permissions table (ARSFOLPERMS) [299](#)
folder table (ARSFOL) [297](#)
folder user fields table (ARSFOLFLDUSR) [298](#)
folders
access permission [36](#)
adding [64](#), [71](#)
administrator [29](#)
administrator permission [36](#)
concepts [33](#)
defining [64](#), [71](#), [85](#), [93](#)
fields
defining [72](#)
fields permission [36](#)
named queries permission [36](#)
overview [33](#)
permissions [36](#), [37](#), [72](#)
FORM parameter

FORM parameter (*continued*)
application groups [187](#)
ARSLOAD program [187](#)
full-text indexing [1](#)

G

generic indexer
parameter file [86](#), [94](#)
group administration [55](#)
group object [331](#)
group owner [31](#)
group table (ARSGROUP) [300](#)
groups
adding [55](#)
adding users [56](#)
assigning users [56](#)
concepts [31](#)
defining [55](#)
naming [31](#)
overview [31](#)
owner [31](#)
permissions [37](#)

H

hold object [369](#)
holds
concepts [34](#)
overview [34](#)

I

ICU
ARSADMIN program [119](#)
image data
loading [83](#)
image files
index data [86](#)
importing administrative objects [103](#)
Importing and exporting administrative objects through an
XML interface
Files provided [104](#)
overview [103](#)
importing migrated index data [9](#), [119](#)
importing tables [137](#)
inactivity time out [20](#)
index data
about [1](#)
adding [141](#)
column delimiter [121](#)
creating [86](#), [94](#)
deleting [4](#), [9](#), [101](#), [141](#), [181](#)
expiration processing [4](#), [9](#), [181](#)
fields
defining [69](#)
generic indexer [86](#), [94](#)
importing [9](#), [119](#)
indexes
defining [70](#)
load information [70](#)
loading [75](#), [83](#), [91](#), [185](#)
migration processing [6](#), [9](#)

index data (*continued*)
 resource information [70](#)
 restarting a load process [99](#)
 retrieving [141](#)
 system catalog tables [2](#)
 triggers [69](#)
 updating [141](#)
indexes
 defining [70](#)
indexing [1](#)
input dataset
 specifying in ARSLOAD job [196](#)
installing the administrative client [13](#)
instances
 -h parameter [117](#)
 -I parameter [117](#)
 programs, specifying [117](#)
internal date format [131](#)

J

JES Spool Data Capture facility, ARSYSPIN [241](#)

K

keyboard [285](#)

L

LDAP [51](#)
limitations
 file size [77](#)
 size of input files [77](#)
line data
 converting to AFP [223](#)
load Id [101](#)
load identifier [101](#)
load library
 specifying in ARSLOAD job [196](#)
 STEPLIB DD statement [196](#)
load table (ARSLOAD) [303](#)
loading data [75](#), [83](#), [91](#)
loading image data [83](#)
loading index data [75](#), [83](#), [91](#)
loading reports [75](#)
loading TIFF image data [83](#)
loading user-defined data [91](#)
logging on a server [19](#)
logical views permission [36](#)
logical views table (ARSAPPUSR) [294](#)
login processing [21](#), [22](#)

M

maximum password age [19](#)
messages
 128 ApplGrp Segment Expire [6](#)
 14 DB Info Exported [7](#)
 164 ApplGrp Segment Maintenance [3](#)
 166 ApplGrp Segment Export [7](#)
 168 ApplGrp Segment Not Available [9](#)
 87 ApplGroup Load [80](#), [99](#)
 87 ApplGrp Load (System Migration) [7](#)

messages (*continued*)
 application groups [3](#), [6](#), [7](#), [9](#), [80](#), [99](#)
 collecting statistics [3](#)
 expiration processing [6](#)
 loading data [80](#), [99](#)
 migration processing [7](#), [9](#)
migrating data to table spaces [219](#)
migration processing [6](#), [9](#), [119](#), [199](#)
minimum password length [20](#)
MODIFY command [281](#)

N

named queries permission [36](#)
named query table (ARSNAMEQ) [303](#)
naming groups [31](#)
naming users [29](#)
NLS [119](#)
node table (ARSNODE) [304](#)

O

OAM objects
 administration [76](#)
 ARSEXPIR program [181](#)
 collection name [181](#)
 expiration processing [181](#)
 installing [76](#)
 planning [76](#)
objects
 Content Manager OnDemand [106](#)
obtaining Content Manager OnDemand internal date format [131](#)
ODF datasets
 DSNAOINI [196](#)
 SDSNEXIT [196](#)
 SDSNLOAD [196](#)
 specifying in ARSLOAD job [196](#)
 STEPLIB DD statement [196](#)
OnDemand object
 child objects [107](#)
OnDemand XML file
 data model [322](#)
 objects [322](#)
operator commands
 MODIFY command [281](#)
 overview [281](#)
 START command [281](#)
 STOP command [284](#)

P

parameters
 ARSLOAD EXEC statement [196](#)
 specifying in ARSLOAD job [196](#)
PARM parameter
 ARSLOAD EXEC statement [196](#)
 specifying in ARSLOAD job [196](#)
passwords
 ADMIN user password [29](#)
 age [19](#)
 ARS.PTGN user exit [117](#)
 built-in administrator password [29](#)

- passwords (*continued*)
 - case sensitivity [21](#), [22](#)
 - changing [19](#)
 - exits [117](#)
 - expiration [19](#)
 - length [19](#), [20](#)
 - login processing [21](#), [22](#)
 - maximum password age [19](#)
 - minimum password length [20](#)
 - omitting [117](#)
 - security user exit [20](#)
 - specifying in ARSLOAD [187](#)
 - user exits [117](#)
- permissions
 - about [35](#)
 - application groups [36](#), [67](#)
 - concepts [35](#)
 - examples [37](#)
 - folders [36](#), [72](#)
 - overview [35](#)
 - strategies [39](#)
- primary storage nodes
 - adding [60](#)
 - concepts [32](#)
 - defining [60](#)
 - overview [32](#)
- print processor table (ARSDFPPT) [307](#)
- printer object [332](#)
- printer options table (ARSPTOPTS) [312](#)
- printer table (ARSPRT) [312](#)
- printer user table (ARSPRTUSR) [313](#)
- printers
 - adding [58](#)
 - concepts [31](#)
 - defining [58](#)
 - options [71](#)
 - overview [31](#)
- printing documents [141](#)
- processing rows
 - in the ARSOAM_DELETE table [177](#)
- program properties [22](#)
- programs
 - ARSEXOAM [177](#)
 - ARSEXPIR [181](#)
 - ARSYSPIN [241](#)
 - capture facility, ARSYSPIN [241](#)
 - date formats [131](#)
 - deleting data [181](#)
 - expiring data [181](#)
 - file input [185](#)
 - ICU [119](#)
 - instance name [117](#)
 - internal date format [131](#)
 - JES Spool Data Capture facility, ARSYSPIN [241](#)
 - obtaining Content Manager OnDemand internal format [131](#)
 - omitting password [117](#)
 - omitting userid [117](#)
 - processing rows [177](#)
 - quote characters [116](#)
 - reading a syntax diagram [113](#)
 - Spool Data Capture facility, ARSYSPIN [241](#)

Q

- query restriction [36](#)

R

- reading a syntax diagram [113](#)
- recipient list object [372](#)
- recipient list table (ARSDFLIS) [309](#)
- recipient object [371](#)
- recipient/list table (ARSDFUOT) [309](#)
- removing a report [119](#)
- reorganizing the database [137](#)
- report bundle definition table (ARSDFBDT) [310](#)
- report bundle object [373](#)
- report ID object [373](#)
- report ID table (ARSDFCRT) [311](#)
- report specifications archive definition exit [187](#)
- report to monitor usage [207](#)
- report wizard [26](#)
- report, creating from system log [207](#)
- reporting utility [207](#)
- reports
 - accessing [69](#)
 - adding [64](#), [141](#)
 - defining [64](#)
 - deleting [101](#), [119](#), [141](#)
 - estimating storage space for [75](#)
 - example of [69](#)
 - expiring [4](#)
 - interaction with TSM [4](#)
 - load Id [101](#)
 - load identifier [101](#)
 - loading [75](#), [185](#)
 - printing [141](#)
 - restarting a load process [99](#)
 - retrieving [119](#), [141](#)
 - storing [119](#)
 - unloading [101](#), [119](#)
 - updating [141](#)
- requirements
 - administrative client [13](#)
 - hardware [13](#)
 - software [13](#)
- resources [70](#), [80](#)
- resources table (ARSRES) [313](#)
- restarting a load process [99](#)
- retrieve
 - ARSADMIN program [119](#)
- rows
 - processing [177](#)

S

- SDSNEXIT dataset
 - specifying in ARSLOAD job [196](#)
- SDSNLOAD dataset
 - specifying in ARSLOAD job [196](#)
- security and user administration [54](#), [55](#)
- segment table [79](#)
- segment table (ARSSEG) [314](#)
- server printers
 - adding [58](#)

server printers (*continued*)

- concepts [31](#)
- defining [58](#)
- options [71](#)
- overview [31](#)

servers

- adding [18](#)
- adding items [25](#)
- copying items [25](#)
- drag and drop operations [26](#)
- exporting items [25](#)
- logging on [19](#)
- system parameters [23](#)
- updating [18](#)

size of input files [77](#)

Spool Data Capture facility, ARSYSPIN [241](#)

START command [281](#)

start up parameters [22](#)

starting the administrative client [17](#)

statistics

- collecting [3](#), [137](#)

STEPLIB DD statement

- specifying in ARSLOAD job [196](#)

STOP command [284](#)

stopping

- ARSSOCKD [213](#)
- server [213](#)

storage

- administration [60](#)
- for reports [75](#)
- management [66](#)

storage nodes

- adding [60](#)
- concepts [32](#)
- defining [60](#)
- loading data [80](#)
- overview [32](#)

storage objects

- loading [80](#)
- retrieving [119](#)
- storing [119](#)

storage set object [333](#)

storage set table (ARSSSET) [315](#)

storage set table (ARSSYS) [316](#)

storage sets

- adding [60](#)
- application group
 - assigning to [66](#)
- assigning to an application group [66](#)
- concepts [32](#)
- defining [60](#)
- loading data [80](#)
- overview [32](#)

store

- ARSADMIN program [119](#)

syntax diagram

- reading [113](#)

syntax diagrams

- reading [114](#)

system administrator [29](#)

system catalog tables [2](#)

system control tables

- annotation table (ARSANN) [292](#)
- application group data table [320](#)

system control tables (*continued*)

- application group field alias table (ARSAGFLDALIAS) [291](#)
- application group field table (ARSAGFLD) [290](#)
- application group permissions table (ARSAGPERMS) [292](#)
- application group table (ARSAG) [288](#)
- application group to folder field mapping table (ARSAG2FOL) [289](#)
- application table (ARSAPP) [293](#)
- ARSAG (application group table) [288](#)
- ARSAG2FOL (application group to folder field mapping table) [289](#)
- ARSAGFLD (application group field table) [290](#)
- ARSAGFLDALIAS (application group field alias table) [291](#)
- ARSAGINDEX (application group composite index table) [291](#)
- ARSAGPERMS (application group permissions table) [292](#)
- ARSANN (annotation table) [292](#)
- ARSAPP (application table) [293](#)
- ARSAPPUSR (user logical views table) [294](#)
- ARSCAB (cabinet table) [295](#)
- ARSCAB2FOL (cabinet to folder mapping table) [295](#)
- ARSCABPERMS (cabinet permissions table) [296](#)
- ARSCFSODWORK [296](#)
- ARSFOL (folder table) [297](#)
- ARSFOLFLD (folder field table) [297](#)
- ARSFOLFLDUSR (folder user fields table) [298](#)
- ARSFOLPERMS (folder permissions table) [299](#)
- ARSFTIWORK [299](#)
- ARSGROUP (group table) [300](#)
- ARSHOLD [301](#)
- ARSHOLDMAP [301](#)
- ARSHOLDPERMS [302](#)
- ARSHOLDWORK [302](#)
- ARSLOAD (load table) [303](#)
- ARSNAMEQ (named query table) [303](#)
- ARSNODE (node table) [304](#)
- ARSPRT (printer table) [312](#)
- ARSPRTOPTS (printer options table) [312](#)
- ARSPRTUSR (printer user table) [313](#)
- ARSRES (resources table) [313](#)
- ARSSEG (segment table) [314](#)
- ARSSSET (storage set table) [315](#)
- ARSSYS (system parameters table) [316](#)
- ARSUSER (user table) [317](#)
- ARSUSRGRP (users in groups table) [319](#)
- ARSUSRGRPID (user group ID table) [320](#)
- cabinet permissions table (ARSCABPERMS) [296](#)
- cabinet table (ARSCAB) [295](#)
- cabinet to folder mapping table (ARSCAB2FOL) [295](#)
- composite index table (application group composite index table) [291](#)
- folder field table (ARSFOLFLD) [297](#)
- folder permissions table (ARSFOLPERMS) [299](#)
- folder table (ARSFOL) [297](#)
- folder user fields table (ARSFOLFLDUSR) [298](#)
- group table (ARSGROUP) [300](#)
- important information [287](#)
- load table (ARSLOAD) [303](#)
- logical views table (ARSAPPUSR) [294](#)
- named query table (ARSNAMEQ) [303](#)

- system control tables (*continued*)
 - node table (ARSNODE) [304](#)
 - overview [287](#)
 - printer options table (ARSPRTOPTS) [312](#)
 - printer table (ARSPRT) [312](#)
 - printer user table (ARSPRTUSR) [313](#)
 - resources table (ARSRES) [313](#)
 - segment table (ARSSEG) [314](#)
 - storage set table (ARSSET) [315](#)
 - system parameters table (ARSSYS) [316](#)
 - user group ID table (ARSUSRGRPID) [320](#)
 - user logical views table (ARSAPPUSR) [294](#)
 - user table (ARSUSER) [317](#)
 - users in groups table (ARSUSRGRP) [319](#)

- system log messages
 - 128 ApplGrp Segment Expire [6](#)
 - 14 DB Info Exported [7](#)
 - 164 ApplGrp Segment Maintenance [3](#)
 - 166 ApplGrp Segment Export [7](#)
 - 168 ApplGrp Segment Not Available [9](#)
 - 87 ApplGrp Load (System Migration) [7](#)
 - application groups [3](#), [6](#), [7](#), [9](#)
 - collecting statistics [3](#)
 - enabling [20](#)
 - expiration processing [6](#)
 - loading data [80](#), [89](#), [97](#), [99](#)
 - migration processing [7](#), [9](#)
 - user exit programs [20](#)

- system log user exit
 - cache storage file system filled [201](#)
 - migration processing [9](#)

- system parameters
 - about [19](#)
 - annotations [21](#)
 - inactivity time out [20](#)
 - login processing [21](#), [22](#)
 - maximum password age [19](#)
 - minimum password length [20](#)
 - password [21](#), [22](#)
 - password age [19](#)
 - password expiration [19](#)
 - password length [20](#)
 - passwords [20](#)
 - setting [23](#)
 - system log [20](#)
 - time out [20](#)
 - user exit logging [20](#)
 - userid [21](#), [22](#)

- system tables
 - expiration processing [199](#)
 - migration processing [199](#)
 - Tablespaces [321](#)
- systemParameters object [324](#)

T

- table spaces
 - about [1](#)
 - backup [137](#)
 - exporting [137](#)
 - importing [137](#)
 - migrating data to [219](#)
- tables
 - closing [219](#)

- TIFF image data
 - loading [83](#)
- time out [20](#)
- triggers
 - defining [69](#)
- troubleshooting [384](#)
- troubleshooting scenarios
 - "EDC5084I" error message [380](#)
 - ARS0025E messages in ARSLOAD [381](#)
 - ARSLOAD error [378](#)
 - ARSLOAD failed while specifying the host with the -h parameter [379](#)
 - ARSLOAD not loading data from HFS directory specified with -d parameter [378](#)
 - Empty file created while using ARSLOAD to monitor an HFS directory [383](#)
 - error
 - Unable to open original file for staging [382](#)
 - Error reading resource file while loading AFP documents with ARSLOAD [382](#)
 - OD database almost full [385](#)
 - The .ARD file cannot be removed after the loading by using load service [382](#)
 - While running ARSMaint to expire cache objects, the job ends with return code 13 [383](#)

U

- unloading a report [101](#), [119](#)
- update servers dialog box [18](#)
- updating administrative client software [15](#)
- user [29](#), [30](#)
- user administration [54](#)
- user administrator [29](#), [30](#)
- user exit programs
 - logging messages [20](#)
- user exits
 - ARS.PTGN [117](#), [187](#)
 - ARSUUPD [187](#)
 - migration processing [9](#)
 - report specification archive definition [187](#)
 - system log [9](#)
 - system log user exit [201](#)
- user group ID table (ARSUSRGRPID) [320](#)
- user logical views table (ARSAPPUSR) [294](#)
- user object [327](#)
- user table (ARSUSER) [317](#)
- user-defined data
 - index data [94](#)
 - loading [91](#)
- userid
 - ARS.PTGN user exit [117](#)
 - case sensitivity [21](#), [22](#)
 - exits [117](#)
 - login processing [21](#), [22](#)
 - omitting [117](#)
 - specifying in ARSLOAD [187](#)
 - user exits [117](#)
- users
 - adding [54](#)
 - adding to a group [56](#)
 - ADMIN user [29](#)
 - administrator [29](#)
 - application group/folder/cabinet administrator [29](#)

users (*continued*)

- ARS.PTGN user exit [117](#)
 - assigning to a group [56](#)
 - authority [30](#)
 - built-in administrator [29](#)
 - changing a password [19](#)
 - concepts [29](#)
 - defining [54](#)
 - exits [117](#)
 - naming [29](#)
 - omitting password [117](#)
 - overview [29](#)
 - passwords [19](#), [21](#), [22](#)
 - permissions [35](#), [37](#)
 - system administrator [29](#)
 - types [29](#)
 - user [29](#)
 - user administrator [29](#)
 - user exits [117](#)
- users in groups table (ARSUSRGRP) [319](#)
- using the administrative client [13](#), [17](#)

V

VSAM data sets

- ARSEXPIR program [181](#)
- expiration processing [181](#)
- high level qualifier [181](#)
- managing [76](#)
- requirements [76](#)
- using for archive storage of documents [76](#)

W

- wizards and assistants [26](#)

X

XML

- exporting [103](#)
- importing [103](#)

XML file

- creating [105](#)
- exporting [231](#)
- importing [231](#)

XML objects

- application [343](#)
- application group [334](#)
- cabinet [367](#)
- distribution [373](#)
- folder [358](#)
- group [331](#)
- hold [369](#)
- printer [332](#)
- recipient [371](#)
- recipient list [372](#)
- report bundle [373](#)
- report ID [373](#)
- storage set [333](#)
- system parameters [324](#)
- user [327](#)

- XML schema file [105](#)



SC19-3364-02

