Content Manager OnDemand for Multiplatforms
Version 8.5

Administration Guide

IBM ®

Content Manager OnDemand for Multiplatforms
Version 8.5

Administration Guide

IBM®

Note

Before using this information and the product it supports, read the information in "Notices" on page 463.

Contents

About this publication

The information in this book can help administrators maintain an IBM® Content Manager OnDemand for Multiplatforms Version 8 Release 4 Modification 2(Content Manager OnDemand) system. Content Manager OnDemand requires a database manager product, such as IBM DB2 Universal Database™ (DB2®). If you plan to maintain reports in archive storage, Content Manager OnDemand requires the IBM Tivoli® Storage Manager product. If you plan to use the Content Manager OnDemand server print function, then you must install the IBM Infoprint Manager (Infoprint) on a workstation that is connected to the Content Manager OnDemand network. If you plan to transmit files from a z/OS® system to an Content Manager OnDemand server, IBM recommends that you use Download for z/OS (Download), a licensed feature of Print Services Facility™ (PSF) for z/OS.

This book describes how to:
- Maintain the database
- Manage cache storage and archive storage
- Use the administrative client to define reports to the system and maintain other types of objects on the system
- Load reports and other types of data into the system
- Use administrative commands to complete other types of tasks

Before you begin, IBM recommends that you familiarize yourself with concepts and terminology used in this book. See the *IBM Content Manager OnDemand for Multiplatforms: Introduction and Planning Guide* for information about Content Manager OnDemand.

IBM hopes that you find using this book and administering Content Manager OnDemand straightforward. You can let IBM know if you find this book helpful and how well the information in this book was presented. Use the reader's comment form at the back of the book to send your comments to IBM.

In this publication, the term UNIX® refers to the AIX®, HP-UX, Solaris, Linux® and Linux for zSeries® operating systems on which you can run the Content Manager OnDemand server programs. Content Manager OnDemand supports 64-bit AIX, HP-UX, Solaris, Linux and Linux for zSeries operating systems only.

The term Windows® client refers to the Content Manager OnDemand client software that runs under Windows XP and Windows Vista. The term Windows server refers to the Content Manager OnDemand server software that runs under the following Windows systems:
- **For 32 bit Windows system:** Windows Server 2003 R2
- **For 64 bit Windows system:** Windows Server 2003 R2 64 bit version

Who should use this publication

This book is of primary interest to administrators that are responsible for working with and maintaining an Content Manager OnDemand system. Some administrators can use this book and the tools described in it to define reports to the system. Other administrators can use this book and the tools described in it to maintain users, groups, printers, storage sets, and so forth. Still other

administrators can use the administrative commands described in this book to maintain the database and cache storage, extract documents from the system, and so forth.

For backup and recovery information, see *Content Manager OnDemand Backup, Recovery, and High Availability* at:
`http://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/SG246444.html`

Support and assistance

Product support is available on the Web. Click Support from the product Web site at: http://www.ibm.com/software/data/ondemand/mp/support.html

Information Center

You can view the product documentation in an Eclipse-based information center that you can install when you install the product. By default, the information center runs in a Web server mode that other Web browsers can access. You can also run it locally on your workstation. See the information center at: www.ibm.com/software/data/ondemand/mp/support.html

PDF publications

You can view the PDF files online using the Adobe® Acrobat Reader for your operating system. If you do not have Acrobat Reader installed, you can download it from the Adobe Web site at: www.adobe.com.

You can find PDF publications for IBM Content Manager OnDemand for Multiplatforms at: http://www.ibm.com/support/docview.wss?rs=129 &uid=swg27012713

How to send your comments

Your feedback helps IBM to provide quality information. Please send any comments that you have about this publication or other documentation. Visit the IBM Data Management Online Reader's Comment Form (RCF) page at www.ibm.com/software/data/rcf.

Be sure to include the name of the product, the version number of the product, and the name of the book. If you are commenting on specific text, please include the location of the text (for example, a chapter and section title, a table number, a page number, or a help topic title).

What's new is version 8.5

Database concepts

Note: This section describes database objects in DB2. If you plan to use some other database management product with Content Manager OnDemand, then see your database management product information for detailed information about the database objects.

This section provides an overview of the key database objects:
- System
- Instance
- Database
- Table
- Index
- Table space
- Containers
- System catalog tables
- Buffer pool
- Configuration parameters
- Log files
- Recovery history file
- Backup files

System

A database *system* represents a physical machine that contains a copy of the database manager, sometimes called an *instance*. In Content Manager OnDemand, the *library server* is the physical machine that contains the instance.

Instance

A copy of the database manager and the data that it manages. An instance has its own data, which other instances cannot access, and also has separate security from other instances on the same machine. Part of initial installation and configuration of the database manager is to create an instance. For Content Manager OnDemand, you should name your primary production, or only, instance *archive* to take advantage of default configuration options and to help with problem determination and support.

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, 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*.

1

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 supports System Managed Space (SMS) table spaces. For an SMS table space, each *container* is a directory in the file space of the operating system. The operating system's file manager controls the storage space.

Content Manager OnDemand system and application group tables, by default, exist in a table space called USERSPACE1. If you define one or more table space file systems to Content Manager OnDemand, then the application group tables can be stored in them instead. (IBM **strongly** encourages you to define table space file systems.) By storing tables in table spaces other than the USERSPACE1 table space, you can improve performance, enable more efficient backup and recovery options, and provide a more flexible configuration.

Container

A container is a physical storage device. It can be identified by a directory name, a device name, or a file name.

A container is assigned to a table space. All database and table data is assigned to table spaces.

A single table space can span several containers, but each container can belong to only one table space. It is possible for multiple containers (from one or more table spaces) to be created on the same physical disk.

The database manager attempts to balance the load of data across the containers.

For SMS table spaces, Content Manager OnDemand decides on the number and locations of the containers, the database manager controls their names, and the file system is responsible for managing them.

Descriptions of concepts

Figure 1 on page 3 illustrates the concepts described so far in this section. The picture also shows the relationship of tables and table spaces, and SMS table spaces.

System

Instance
(archive)

Database

Table Space

Table

Table Spaces are
where tables are
stored:
    SMS
Each container is
a directory in the
file space of the
operating system.
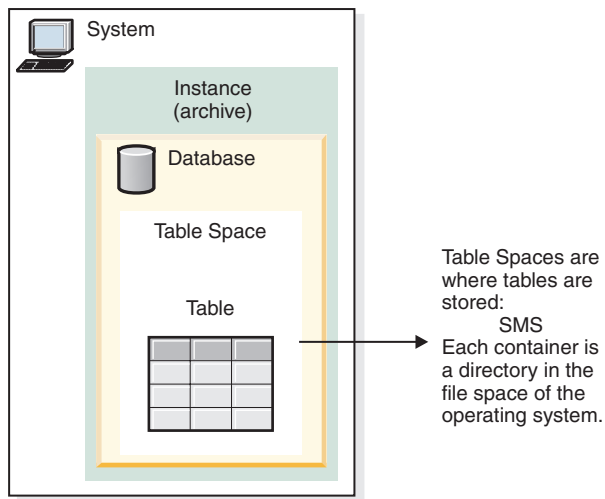
Figure 1. Database objects and relationships

Index

In Content Manager OnDemand, an index is a key that points to a document. An index allows more efficient access to documents by creating a direct path to a document through pointers.

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.

The SQL optimizer automatically chooses the most efficient way to access data in tables. The optimizer takes indexes into consideration when determining the fastest access path to data.

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.

Configuration parameters

When a database instance or a database is created, a corresponding configuration file is created with default parameter values. You can modify the parameter values to improve performance. Configuration files contain parameters that define values such as the resources allocated to the database manager and to individual databases, and the diagnostic level. There are two types of configuration files: the database manager configuration file for each instance and the database configuration file for each database.

The database manager configuration file is created when an instance is created. The parameters it contains affect system resources at the instance level, independent of any one database that is part of that instance. Many of these parameters can be changed from the system default values to improve performance or increase capacity, depending on your system's configuration.

A database configuration file is created when a database is created, and resides where the database physically resides. There is one configuration file per database. Its parameters specify, among other things, the amount of resources to be allocated to that database. Many of the parameters can be changed to improve performance or increase capacity. Different changes may be required depending on the type of activity in that specific database.

Log files

Each database includes recovery logs, which are used to recover from application or system errors. In combination with database backups, logs are used to recover the consistency of the database right up to a point in time when an error occurred.

All databases have logs associated with them. These logs keep records of database changes. Some logs, called *active logs*, prevent a failure (system power, application error) from leaving a database in an inconsistent state. They restore the state of a database to the point before the change. Other logs, called *archived logs*, enable forward recovery to any point in time before the failure.

Active and archive logging

Active logs contain transactions which have not yet been committed or rolled back, or whose changes have not yet been written to disk. Active logs are located in the database log path directory (see ARS_PRIMARY_LOG_PATH in the ARS.CFG file).

Archive logging enables forward recovery using active and archived logs to any point in time before a failure, rather than only to the point in time of a full backup.

With archive logging, active logs are retained and become online, archived logs. In addition, archived logs can be moved off line and still used for roll-forward recovery.

With archive logging, it becomes necessary to pay more attention to the handling of the logs and to ensure their safety. The ability to perform roll-forward recovery of your database is dependent on the integrity of the logs. Performance factors such as the location and size of the logs needs to be evaluated for the impact on overall system performance. In planning for disaster recovery, be sure to remember that log data must be stored off site, or at least safely away from the disaster, in order to recover your database beyond the point of the last full, off line backup.

When all changes in the active log are no longer needed for normal processing, the log is closed and becomes an archived log. An archived log is said to be *online* when it is stored in the database log path directory.

You also have the ability to store archived logs in a location other than the database log path directory, by using a user exit program. An archived log is said to be offline when it is not stored in the database log path directory.

In Content Manager OnDemand, archive logging is activated by enabling the USEREXIT parameter. You select a location for the archived log files by specifying DISK or TSM when you create the database. If you specify DISK, then the archived logs are stored in the directory named by the ARS_ARCHIVE_LOG_PATH parameter in the ARS.CFG file. If you specify TSM, then you must configure TSM to maintain the archived logs. See the *Installation and Configuration Guide* for help with configuring TSM to maintain the archived logs.

Figure 2 shows the database using active log files and offline archived log files that you maintain on disk.
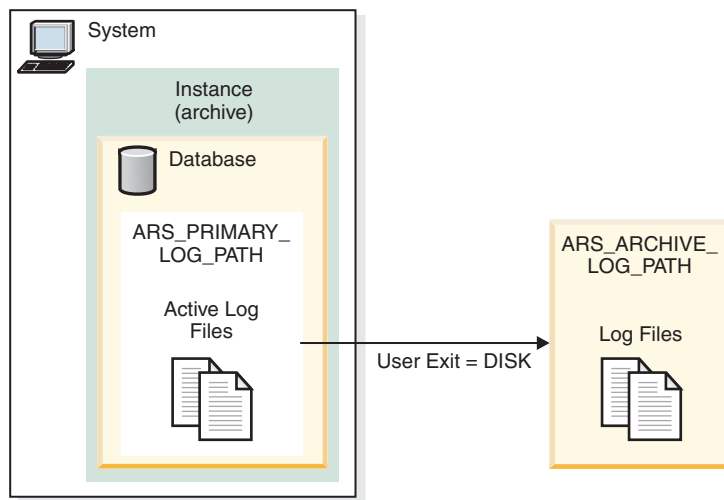
Figure 2. Database objects and relationships

Figure 3 on page 6 shows the database using active log files and offline archived log files that are maintained by the archive storage manager.
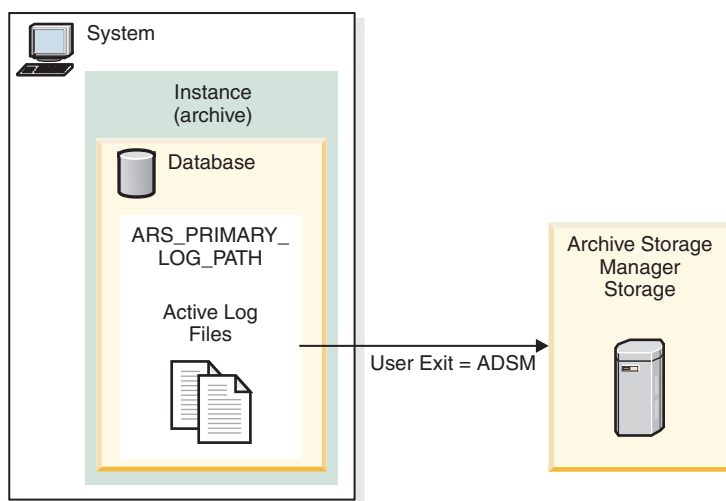
Figure 3. Database objects and relationships

Configuration parameters for database logging

Content Manager OnDemand uses two main configuration parameters for database
logging:

- Log file size. Determines the size of a log file, in 4 KB blocks. The default value
  (see ARS_LOGFILE_SIZE in the ARS.CFG file) is 1000, which means that each
  log file is about 4 MB in size.
- The number of active log files. The maximum number of active log files that can
  be allocated to the database at one time. The default value (see
  ARS_LOG_NUMBER in the ARS.CFG file) is 40.

These values provide approximately 160 MB of active log file space.

Important: The database manager will stop if there is not enough active log file
space to complete a transaction.

Using the standard Tivoli Storage Manager backup commands for file system backups

You can use the standard Tivoli Storage Manager backup commands to do file
system backups. See the *Tivoli Storage Manager Administrator's Reference* for detailed
information about the commands that you can use and to schedule backups.

If you use the standard Tivoli Storage Manager backup commands to backup file
systems on the server, it is recommended that you exclude the database file
systems and the cache file systems from the backup. The database file systems
should be backed up using the facilities provided by OnDemand, such as the
ARSDB program, although the database backup images created by the ARSDB
program can be maintained by Tivoli Storage Manager. You should use the data
loading and migration facilities of OnDemand to make sure that data in cache
storage is copied to Tivoli Storage Manager-managed storage as needed.

While you can store the file system backups in the same optical library as the
OnDemand data, you should define different client nodes to hold the file system
backups. The client nodes should be assigned to a different domain than the
OnDemand data. The retention period of the file system backups will likely be
different than the retention period for the OnDemand data.

File system backups generally require re-writable media.

You should plan a library with at least two drives:
- One for the file system backup.
- One for the OnDemand data.

If you plan to run the file system backups at the same time when you load data into OnDemand, plan a library with four drives, or you may store the file system backups in a different library than the OnDemand data.

Recovery history file

A recovery history file contains a summary of the backup information that can be used in case all or part of the database must be recovered to a given point in time. A recovery history file is created with each database and is automatically updated when certain actions are performed, such as backups and restores.

Backup files

Backup files contain a copy of the database or table spaces that can be used to recover the database.

The following are general guidelines for planning what types of backup files you might need:
- Databases that contain read-only data do not need to be protected through archive logging if full, offline backups are run following each new data load activity. For most customers, the table spaces that contain application group data is probably read only.
- With continuously updated data that is deemed important to your business, you must use archive logging. If you use the audit feature of Content Manager OnDemand to update documents throughout the day, then you probably need to use archive logging.
- If your database must be continuously available, you must take online backups. This requires the use of archive logs.
- If in the event of a failure your database must be recovered in a short time, you will need to run more frequent backups. In this case, you need to establish how long it would take to recover from a failure (the sum of the time to restore the database from a backup plus the time needed to roll the log forward).

   Note: Storing application group data in table spaces may reduce the time required to recover from a failure of a single device.

   Consideration should also be given to keeping the database on disk arrays or mirrored volumes.

Probably the most common type of failure is caused by media problems. This is not limited to disk problems, but can extend to other I/O devices, including disk controllers and tape devices. As a starting point, do not back up your database to the same disk on which the production version exists: use either a separate disk or external media. The handling of your logs should be similar: consider directing these to a separate physical disk from that of the database. In addition to protecting against a disk failure affecting both, this may also result in performance improvements.

Database concepts **7**

Though unlikely, it is possible that your backup media could suffer a problem just when it is needed to let you recover from a disk failure. Consider the impact of a tape going bad. If your data is absolutely critical, you should consider having duplicate tape media. Another strategy is to minimize the potential for impact caused by a bad disk. This applies to the disks that both the database and logs reside upon. Using disk arrays for your database volumes or logs (or both) is perhaps the best defense against disk media failures. If you extend redundancy to disk controllers as well, it is highly unlikely that your database will ever be unavailable or that logs will be lost due to a media failure.

Table spaces and storage volumes

Overview

A database is organized into table spaces. A table space is a place to store tables. With DB2, Content Manager OnDemand supports SMS table spaces. For an SMS table space, each *container* is a directory in the file space of the operating system. The operating system's file manager controls the storage space.

All database and table data is assigned to table spaces. Content Manager OnDemand system and application group data tables, by default, exist in a table space called USERSPACE1. If you define one or more table space file systems to Content Manager OnDemand, then the application group data tables can be stored in them instead. (IBM **strongly** encourages you to define table space file systems.) By storing tables in table spaces other than the USERSPACE1 table space, you can improve performance, enable more efficient backup and recovery options, and provide a more flexible configuration.

A container is a physical storage device. It can be identified by a directory name, a device name, or a file name. A container is assigned to a table space. A single table space can span several containers, but each container can belong to only one table space. It is possible for multiple containers (from one or more table spaces) to be created on the same storage volume.

Figure 4 shows an example of the relationship of tables and table spaces within the database and the containers, disks, and file systems associated with the database. The LOAN.1RBA table is in the LOAN.1RBA table space which spans Containers 0, 1 and 2. This example shows each container existing on a separate storage volume. The database manager attempts to balance the load of the table data across the containers. As a result, all containers will be used to store table data.
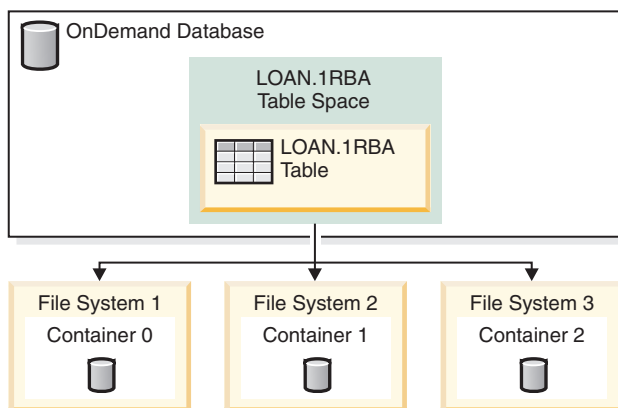
Figure 4. Tables and table spaces within a database

In Figure 4, there are three table space file systems defined to Content Manager OnDemand. When creating a table, Content Manager OnDemand first creates a table space and then creates the table in the table space. (In Content Manager OnDemand, for application groups that use SMS table spaces, there is a one-to-one relationship between tables and table spaces.) Content Manager OnDemand attempts to balance the placement of containers across the file systems and the

storage volumes within the file systems. Content Manager OnDemand creates containers for the table space in the file system (or file systems) with the most free space. If there are multiple table space file systems defined to Content Manager OnDemand, then containers may be placed in more than one file system. If there are multiple storage volumes in a table space file system, Content Manager OnDemand places the container on the storage volume with the most free space.

Content Manager OnDemand decides on the number and locations of the containers, the database manager controls their names, and the file system is responsible for managing them. By controlling the amount of data written to each container, the database manager evenly spreads the data over the table space file systems.

Database storage configuration

For SMS table spaces, Content Manager OnDemand determines the number of containers to create when creating the table space. By defining multiple table space file systems to Content Manager OnDemand and assigning multiple storage volumes to the file systems, you enable Content Manager OnDemand to create multiple containers and spread them across the storage system, which improves the performance of your system. When you need to increase the size of your table space file systems, add an equal amount of storage to each file system. The pictures that follow provide more information about configuring and using table space file systems.

Figure 5 shows the placement of containers and table data with one table space file system defined to Content Manager OnDemand. The file system contains one storage volume. All of the data is written to a single container and storage volume.
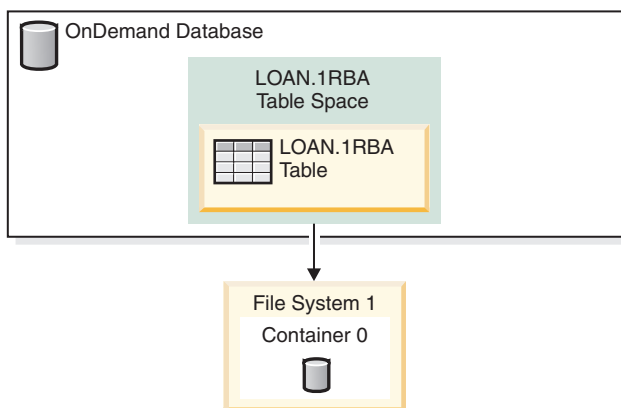
Figure 5. Defining table space file systems. Part 1 of 7

Figure 6 on page 11 shows the placement of containers and table data with one table space file system defined to Content Manager OnDemand. The file system contains two storage volumes. Content Manager OnDemand creates a table space with one container on each storage volume in the file system. The database manager balances the load of data across the containers.
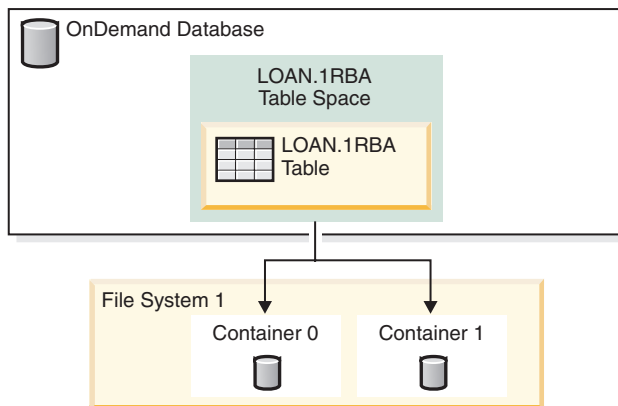
Figure 6. Defining table space file systems. Part 2 of 7

Figure 7 shows the placement of containers and table data with two table space file systems defined to Content Manager OnDemand. Each file system contains one storage volume. Content Manager OnDemand creates a table space with two containers – one on each storage volume in each file system. The database manager balances the load of data across the containers.
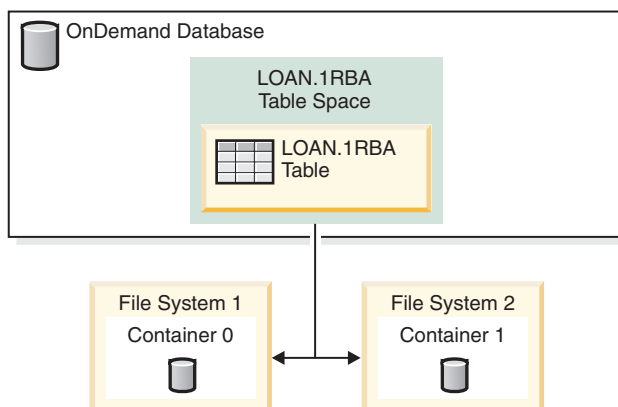
Figure 7. Defining table space file systems. Part 3 of 7

Figure 8 on page 12 shows the placement of containers and table data with two table space file systems defined to Content Manager OnDemand. Each file system contains two storage volumes. Content Manager OnDemand creates four containers – one on each of the storage volumes in each file system. The database manager balances the load of data across the containers.
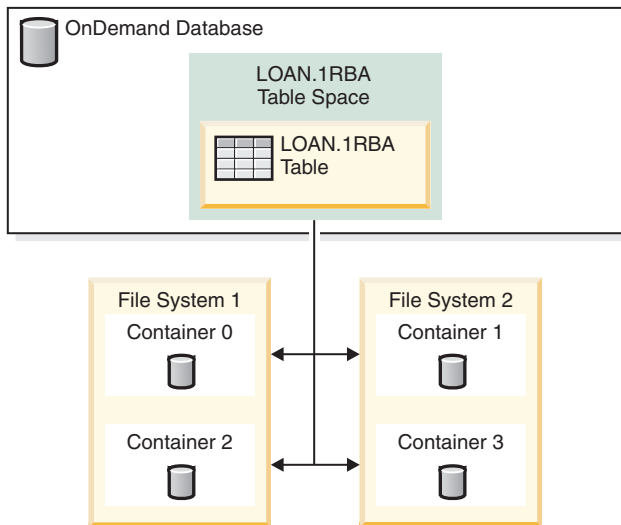
Figure 8. Defining table space file systems. Part 4 of 7

Figure 9 shows the recommended method for increasing the size of a table space file system. Add an equal amount of storage to each file system. In the example, one storage volume is added to each file system.
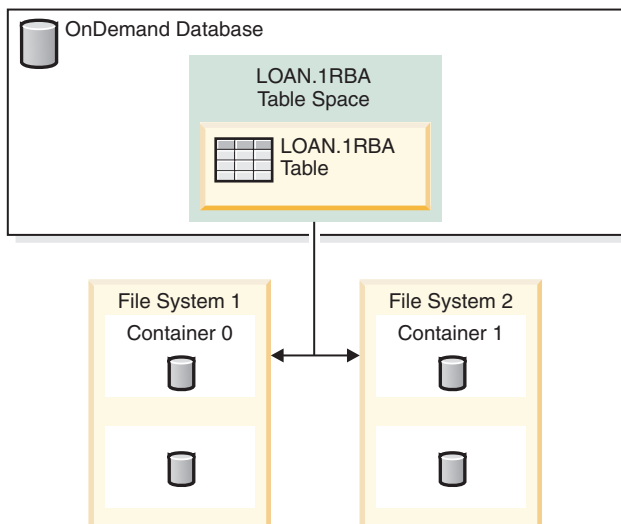
Figure 9. Defining table space file systems. Part 5 of 7

Because all of the containers for an SMS table space must be created when the table space is created, the only time that Content Manager OnDemand can use new storage in a table space file system is when it creates a new table space. For existing table spaces, the database manager continues to load data into the containers that were created when the table space was created. When Content Manager OnDemand creates a new table space, it can create containers on the new storage volumes, as Figure 10 on page 13 shows.
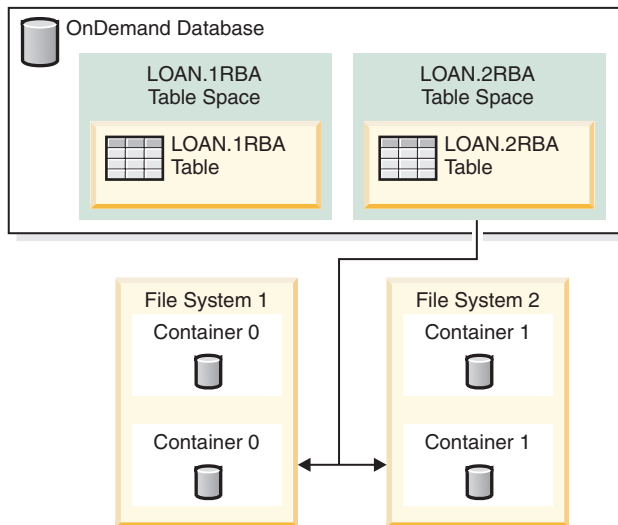
Figure 10. Defining table space file systems. Part 6 of 7

Figure 11 shows what would happen if, instead of adding storage volumes to existing table space file systems, you placed the storage volumes in a new file system and added the new file system to the list of Content Manager OnDemand's table space file systems. One of the parameters that Content Manager OnDemand uses to determine where to place the containers for new table spaces is the amount of free space available in a table space file system. Because the new file system contains the most free space (at least for the time being), that is where Content Manager OnDemand will place the containers for new table spaces. Adding a new table space file system to Content Manager OnDemand may unbalance the performance of your database storage configuration. (In the example, all new data loading, and searching for that data, takes place on the same file system.)
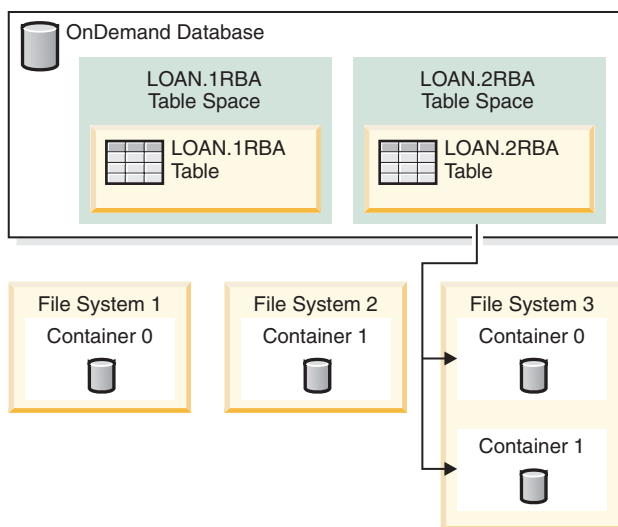
Figure 11. Defining table space file systems. Part 7 of 7

Recommendations for database storage configuration

For a database using SMS table spaces, do the following:

- Estimate the database storage space required to hold the index data for each report that you plan to load into Content Manager OnDemand. See the *IBM Content Manager OnDemand for Multiplatforms: Introduction and Planning Guide* for details and examples.

- Allocate enough database storage space to keep the index data for any given version of a report in the database at least until such time that 99 percent of the requests for the report have passed. As a rule of thumb, if there is any chance that someone in your organization will need to retrieve a version of a report, keep the index data in the database – don't allow it to be migrated.

- Define table space file systems to Content Manager OnDemand and enable table space support in your application groups.

- Configure your table space file systems deep and wide and, most importantly, even. Before you begin loading reports, define as many table space file systems as your hardware configuration permits. Add the same number of storage volumes to each file system. The storage volumes should be the same size and speed. (A typical configuration for a large database is four table space file systems, with four, 8.4 GB storage volumes in each file system. Given such a configuration, Content Manager OnDemand could create up to 16 containers for each table space, enabling the database manager to balance the load of data across the storage configuration. This would not only improve the performance when loading data, but also help when lots of users access data from the same table.)

- After your Content Manager OnDemand system is in production for a while, it may become necessary to increase the amount of storage space available for your database. You may find that you need to allocate more storage for new reports, growing application groups, and so forth. When you are ready to increase the size of your database storage configuration, add an equal amount of storage to each table space file system. That way, Content Manager OnDemand can continue to balance the load of the table data across the storage configuration. (Of course, if you've reached the operating system limits placed on file systems, you'll need to define new table space file systems to Content Manager OnDemand.)

- Expire data periodically, and migrate data only when absolutely necessary. (Never use migration as a convenient way to reclaim database storage space. If you are running low on database storage space, add more storage volumes.)

- Before you import migrated index data, add an equal amount of storage space to each table space file system.

Note: If you plan to place your database on a disk array or use mirrored volumes, see your database manager and operating system information for help with optimizing performance and the hardware configuration.

Managing database growth

As a database administrator, you need to estimate the size of tables and indexes, and to check the amount of space available in a table space, adding more space to an existing table space when it gets full. You should use the database manager tools to monitor table and table space growth to make sure that space allocation does not become a critical issue. You should also monitor the space used by the database and the space that is available on the system.

This section describes how to:
* Estimate the size of tables and indexes
* Check the amount of space available in a table space
* Add more space to an existing table space when it starts to get full

Estimating table and index size

You can estimate the amount of storage space required for new or existing tables or indexes by using the worksheets provided with the *IBM Content Manager OnDemand for Multiplatforms: Introduction and Planning Guide*. For DB2, you can also estimate the size of existing tables and indexes by invoking the Estimate Size dialog. You can invoke this dialog by selecting a table or index in the Control Center, clicking mouse button two, and selecting Estimate Size. The size is estimated on the definition of the particular table and its dependent indexes. The estimate is the projected amount of storage space that would be used when the table has a given number of rows. See the online help for the Control Center for more information.

Estimating the size of a table or index is helpful if you want to:
* Create a new table and you want to know how large to make the table space
* Create a new table based on the size estimate of an existing table
* Know how much space is used by different table and index objects in a table space because the system is running out of storage space
* Estimate the projected size of a table prior to loading a report into the system

Checking space available in a table space

In DB2, to check the amount of space available in an SMS table space, use the facilities provided by your operating system to monitor space usage and to ensure that available room in the directory for the table space is not exhausted.

Adding more space to a table space

In general, you cannot extend the size of an SMS table space very easily because SMS capacity depends on the space available in the file system and the maximum size of the file supported by the operating system. However, depending on your operating system, you may be able to increase the size of a file system using the operating system facilities. For an SMS table space on a UNIX-based file system, you can increase the size of the table spaces by using the appropriate UNIX-based system commands. See the documentation for the UNIX system that you are running. If the file system containing the SMS table space also contains non-DB2 files, then you may be able to move these files to another file system, thus making

more room available in the file system for DB2's use. You can also perform a redirected restore, which involves restoring a table space into a larger number of containers than it was backed up from. See the DB2 information for help with a redirected restore process.

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.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides help with scheduling tasks.

You can also manually collect statistics by running the ARSMAINT program from the prompt. See 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.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides help with scheduling tasks.

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 ApplGrp Segment Maintenance (ApplGrp) (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 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. If the Database Organization is Single Load per Database Table, then each table holds the indexes for one and only one load.

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 12 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.
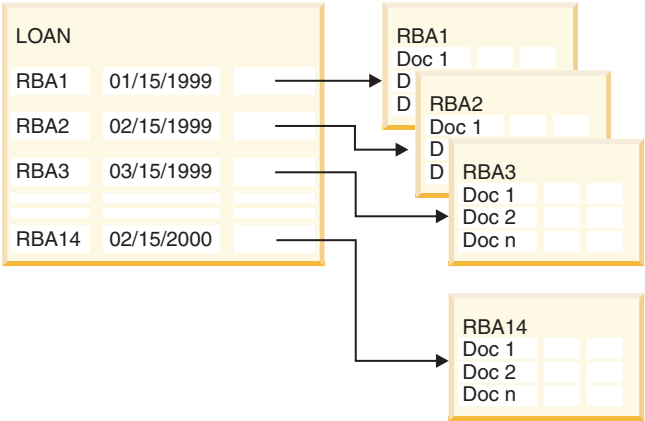
Figure 12. Removing index data. Part 1 of 2

Figure 13 on page 20 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 was 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
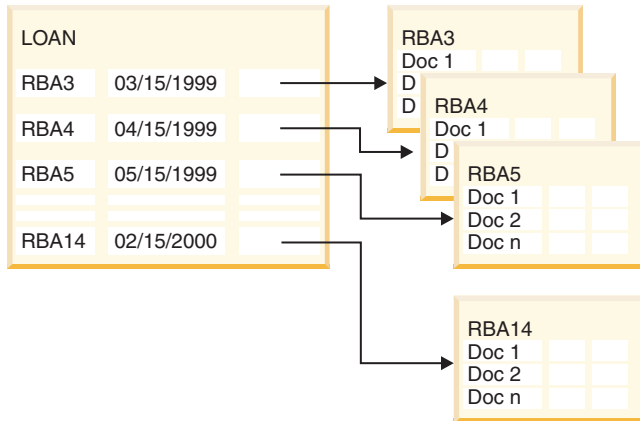
Figure 13. Removing index data. Part 2 of 2

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 "ARSMAINT" on page 291 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.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides help with scheduling tasks.

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 "ARSMAINT" on page 291 for more information about the ARSMAINT program.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides help with scheduling tasks.

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 "ARSMAINT" on page 291 for more information about the ARSMAINT program.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides help with scheduling tasks.

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 ARSMAINT 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.

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 into the database. 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.

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 prints the message at the */dev/console* (UNIX servers) or saves the message in the Event log (Windows servers).

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, such as the Tivoli system management console. 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 ApplGrp 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 Multiplatforms: Installation and Configuration Guide* for more information about the system log user exit.

Tivoli Storage Manager

Before Content Manager OnDemand can migrate index data to archive storage, you must configure Tivoli Storage Manager to maintain Content Manager OnDemand data. You must define the library in which Tivoli Storage Manager will maintain the index data, define a device class and storage pool for the library, and configure a policy domain with the management information that Tivoli Storage Manager uses to maintain the data. The policy domain should maintain the data indefinitely. If you need the system to maintain a backup copy of the index data, you should configure a copy storage pool in Tivoli Storage Manager. See your Tivoli Storage Manager information for assistance with defining libraries, policy information, and configuring a copy storage pool.

Storage sets

Content Manager OnDemand uses the System Migration application group to manage all index data that is migrated to archive storage. You must assign the System Migration application group to a storage set that identifies a client node in Tivoli Storage Manager. You must also configure the System Migration application group to maintain the data indefinitely. See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring the System Migration application group.

The storage set must identify a client node in Tivoli Storage Manager. The client node must be registered in a policy domain that stores the System Migration data on archive media and maintains the data indefinitely. See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with defining Tivoli Storage Manager client nodes and policy domains.

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 index data and report data 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 that Content Manager OnDemand maintains the imported index data in the database before it is scheduled for deletion.

To migrate data from one application group to another application group, you must reload the data. You cannot associate data from one `ApplGroup` to another `ApplGroup`.

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 thirty 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 */dev/console* (UNIX servers) or the Event log (Windows servers) and the system log user exit. If you provide a user-written program to be invoked by the system log user exit, 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 (such as the Tivoli system management console), you should provide a user-written program to process the message. The function of the program can vary, however at a minimum, the program should 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 file system that Content Manager OnDemand uses for temporary storage. The ARS_TMP parameter in the ARS.CFG file (UNIX servers) determines the location that Content Manager OnDemand uses for temporary storage. For Windows servers, you can use the configurator program to define temporary storage locations to Content Manager OnDemand.

Verify database storage space
About this task

Importing migrated index data from archive storage back into the database requires additional database storage. Before you import the index data, you should verify that sufficient free space is available in the database file systems. For example, for DB2, the ARS_DB2_DATABASE_DRIVE parameter in the ARS.CFG file (UNIX servers) determines the location of the base database file system. If you store application group index data in table spaces, the ARS.DBFS file identifies the table space file systems. For Windows servers, you can use the configurator program to determine the drives and directories used by the database.

Verify database log file space
About this task

Importing migrated index data from archive storage back into the database requires database log file storage. Before you import the index data, you should verify that sufficient free space is available in the log file directories. For example, for DB2, the ARS_PRIMARY_LOGPATH and ARS_ARCHIVE_LOGPATH parameters in the ARS.CFG file (UNIX servers) determine the locations for the log files. For Windows servers, you can use the configurator program to determine the drives and directories used to hold the log files.

Run the ARSADMIN program
About this task

Content Manager OnDemand provides the ARSADMIN program to import tables of index data from archive storage back into the database. The following examples shows how to run the ARSADMIN program on a UNIX server 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 Content Manager OnDemand saved in the system log.

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

To run the ARSADMIN program on a Windows server:

Procedure

1. Start → Programs → IBM Content Manager OnDemand for Windows → Command Window.
2. Enter the command and runtime parameters at the prompt. For example:

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

Results

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 "ARSADMIN" on page 215 for a description of the ARSADMIN program and information about the parameters and options.

After you import index data

About this task

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 schedules an imported index table for deletion after it resides in the database for the number of days specified in the Length of Time to Keep Imported Indexes property in application groups. After an imported index data 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, using the CRON facility (UNIX servers) or the configurator program (Windows servers). See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for information about scheduling the ARSMAINT program. You can also run the ARSMAINT program command from the prompt. See "ARSMAINT" on page 291 for a description of the ARSMAINT program and information about its parameters and options.

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.

Monitoring database performance

One of the major tasks involved in administering the Content Manager OnDemand database is to monitor the database performance and tune the system to keep optimization information up-to-date.

This section briefly describes the tools that are available to do these tasks in DB2. For more detailed information, see the *DB2 Administration Guide*. If you are using some other database management product, please see your product information for help with doing these tasks.

Administration tools

DB2 provides GUI tools to help you administer the database from one central location called the "Control Center". From the Control Center, you get a clear overview of all the objects that make up the Content Manager OnDemand database. The Command Center enables you to issue DB2 database commands, SQL statements, and operating system commands. The Alert Center notifies you when thresholds that you have set have been exceeded. The Journal allows you to view the status of jobs and to view the messages log. The Information Center gives you quick access to the information in the DB2 product manuals and sample programs and provides access to other sources of DB2 information on the Web.

For some functions that you can perform with the GUI tools, you are given the option of using a SmartGuide. SmartGuides are invoked from the pop-up menus in the Control Center. They provide a greater level of help by prompting you step-by-step on how to fill in the information necessary for the task that you are doing and even making calculations and recommendations based on information that you supply. SmartGuides are very useful if you are a new database administrator or someone who only administers a database occasionally. See the online help for Control Center for a list of available SmartGuides and how to use them.

From Control Center, you can administer database objects for the DB2 family of products for UNIX, and Windows. You can also administer DB2 for z/OS subsystems from the Control Center if an Administration Server (DAS) is running on the DB2 for z/OS system and if a DB2 Connect™ product is available to the client on which the Control Center is running. See your DB2 information for details.

You can administer the following database objects from the Control Center:
- Systems
- Instances
- Databases
- Tables
- Indexes
- Users and Groups

You can do the following tasks from the Control Center:
- Manage database objects. For example, you can create, alter, and drop table spaces; you can also manage users.

- Manage data. For example, you can import, export, and reorganize data and gather statistics.
- Schedule jobs.
- Perform preventative maintenance by backing up the database.
- Monitor performance and perform troubleshooting.
- Configure and tune instances and databases.

Monitoring database performance

The Performance Monitor provides information about the state of DB2 and the data that it controls. It is a graphical utility that you can customize for your database environment. You can define thresholds or zones that trigger warnings or alarms when the values being collected by the Performance Monitor are not within acceptable ranges. When a threshold is reached, you can specify that you want any or all of the following actions to occur:

- You are notified through the Alert Center
- You receive an audible alarm
- A program is run
- A message is displayed
- No notification is given

You can monitor DB2 objects such as instances, databases, tables, table spaces, and connections by selecting the object in the Control Center and clicking mouse button two. From there, you can choose to start monitoring activity.

When an object is being monitored, its color represents the severity of a problem as defined by the thresholds that you have defined. Green signifies that the monitor is running and everything is fine. Yellow is a warning and signifies that the monitor is reaching the thresholds that you have set. Red indicates an alarm and that the monitor has reached the threshold. You can use the predefined monitors that are included with DB2 or you can create your own monitors. See the *DB2 Administration Guide* for a list of the predefined monitors. See the online help for examples of how to use the predefined monitors and how to create your own monitors.

You can use the Performance Monitor when you need to monitor an existing problem or when you want to observe the performance of your system. It lets you take a snapshot of database activity and performance data at a point in time. These snapshots are used for comparison over time. The information can help you to identify and analyze potential problems, or identify exception conditions which are based on thresholds that you set. You should use the Performance Monitor if you need to know the performance of the database manager and the Content Manager OnDemand database at a single point in time and look at trends over time. You can also use it to get a visual overview of what elements are in a state of alarm. This helps you to identify which parameters may need tuning. You can then look closely at the parameters that have been set for that element and change them to improve performance.

You can use the Performance Monitor to see if bottlenecks are hardware related. For example, you may want to monitor database connection activity or table space, buffer pool, and I/O activity. To see if bottlenecks are hardware related, you would analyze the information collected by the Performance Monitor if:

- Too many database tasks are scheduled during peak time

- There is a high number of user connections
- Database partitioning (hardware load balancing) is not well optimized
- The server is being used for more than just a database server

Tuning an Oracle database

This section briefly describes the tools that are available to tune the Content Manager OnDemand database when using Oracle. For more detailed information, see your Oracle technical product information.

In general, most customers use Content Manager OnDemand in one of two ways:

- Long-term archive for larger reports. These customers create table spaces for each application group that they add to the system, load many rows into the database at one time, maintain data on the system for many months or years, and delete a report at a time from the system. These parameters usually result in very static data and a low maintenance operation.
  - A table space contains data from one application group.
  - Inserts are done by a high-volume batch process.
  - After a table reaches its Maximum Rows value, OnDemand closes the table and no additional inserts are made to the table. Closed tables remain available for queries until the data is removed from the system.
  - Data is removed from the system by dropping a table or deleting a large number of consecutive rows at a time.

  Because of the low rate of change within the tables, these customers should seldom or never need to tune their database.

- Short-term archive for smaller reports. These customers store all application group data in the SYSTEM table space (or in one or more DATA table spaces), load very few rows into the database at one time, maintain data on the system for a short period of time, or use the Delete Document method to remove data from the system. These parameters usually result in very dynamic data and a high maintenance operation.
  - A table space contains data from more than one application group.
  - Inserts are done by a low-volume batch process.
  - Inserts and deletes happen frequently.
  - Data is removed from the system by deleting one row at a time.

  Because of the high rate of change within the tables, these customers should plan to tune their database on a time-based schedule, such as a weekly or nightly process. **Note:** Some customers tune their database every day. However, most customers do not tune their database until the optimizer ignores the current set of statistics or generates an inefficient plan. Those customers should tune the database just before they plan to take an offline backup of the database. This schedule keeps the optimization information up to date and minimizes the impact to system availability, because a system outage is already planned.

Tuning the database is done by collecting statistics on the tables, which can provide faster access to the data, thereby improving performance. Statistics on tables are gathered by using the ANALYZE command. When you analyze a table, its associated indexes are automatically analyzed as well. The frequency with which you analyze the tables depends on the rate of change within the tables. **Note:** If you collect statistics and do not notice a visible performance improvement, then dropping and recreating the indexes to your tables may help. Customers in a high maintenance operation may need to periodically rebuild the indexes on their most active tables. See your Oracle information for details about rebuilding indexes.

Content Manager OnDemand provides two programs to collect statistics on database tables: the ARSDB program and the ARSMAINT program.

Using the ARSDB program

You can use the ARSDB program to collect statistics on the Content Manager OnDemand system tables. The Content Manager OnDemand system tables include the user table, the group table, and the application group table. See "System control tables" on page 329 for a complete list of the Content Manager OnDemand system tables. For most customers, the Content Manager OnDemand system tables require very little maintenance. You can probably schedule the ARSDB program to collect statistics once a month (or less often).

The syntax is:

```
/opt/ondemand/bin/arsdb <options>
```

The options are:

–e   Drop configuration indexes

–r   Create configuration indexes

–s   Collect statistics

Using the ARSMAINT program

You can use the ARSMAINT program to maintain the tables that contain user-defined application group data. User-defined application groups are the application groups that you define to the system. Customers in a high maintenance operation should run the ARSMAINT program on a regular schedule.

The syntax is:

```
/opt/ondemand/bin/arsmaint <options>
```

The options are:

–d, –i   Expire index data from the database. The **–i** parameter expires index data that has been imported from archive storage. If you do not migrate index data to archive storage, then you do not need to specify the **–i** parameter.

–e   Migrate index data from the database to archive storage. If you do not migrate index data to archive storage, then you do not need to specify the **–e** parameter.

–r   Collect statistics. **Note:** The ARSMAINT program collects statistics only on the tables that have changed since the last time that statistics were collected. OnDemand keeps information about all of its tables, including the last time that it modified a table and the last time that it collected statistics on a table.

–g applGroup
Process the tables for the specified application group. If you do not specify this parameter and name an application group, then the ARSMAINT program processes all of the user-defined application groups.

Recommended practice for managing table space for Oracle

There are both advantage and disadvantage if you use one table per table space if you use Oracle for the OnDemand database.

- **Advantage:** Having one table per table space enables you to move tables among volumes easily. To move tables among volumes, you should take the table space offline, move the table space's datafile to a new location, rename the datafile in the database, and then bring the table space back online. You might also notice a small performance improvement when each table has its own table space.
- **Disadvantage:** Having one table per table space requires more frequent updates to the `ars.dbfs` file. Also, if each OnDemand application group has its own data table, after a new application group is created, a new application group data table and an associated table space are created.

Therefore, in most cases, it is recommended that you bundle the table space into one single logical unit and spread that logical unit over as many devices as possible by using striping or a logical volume manager. This method provides equal or faster performance than manually separating table spaces. However, with one table per tablespace, other DBA tasks, especially monitoring the growth of tablespaces, can become more difficult, because there are more tablespaces.

Note: Oracle Version 8 allows a maximum of 1023 data files per tablespace. If your operating system allows a maximum size of 2 GB per file, the maximum size for a tablespace is 2 TB (terabytes) minus 2 GB. For example, if you want to add 2 GB to tablespace "ROOT_DBA12", use this command:
```
ALTER TABLESPACE "ROOT_DBA12" ADD DATAFILE '/a099' SIZE 2048M;
```

Also, you can resize a datafile if you use Oracle, for example:
```
ALTER DATABASE foo DATAFILE '/mydatafile' resize 2048M;
```

Database backup and recovery

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

DB2 table spaces

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

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

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides information to help you configure the system for table spaces.

Database backup

Content Manager OnDemand provides the ARSDB program to create backup images of the Content Manager OnDemand database. The ARSDB program can take incremental table space backups and full database backups. The ARSDB program can run while other users are connected to the system or you can run it after you stop the Content Manager OnDemand server programs and other related processes:

- An online backup can be taken when other applications or processes are connected to the database. That is, other applications and processes can continue to read or modify data while the backup is in process. An online backup can be scheduled with the CRON facility (UNIX servers) or Content Manager OnDemand configurator program (Windows servers).

- During an offline backup, only the backup task is connected to the database. Before starting an offline backup, stop the Content Manager OnDemand server programs and other related processes to make sure that no users can connect to the database.

When you back up the database with the ARSDB program, Content Manager OnDemand removes the log files from the archived log file directory, releasing the space taken by files that are no longer needed. However, if you use Tivoli Storage Manager to maintain DB2 log files, then the Tivoli Storage Manager policy domain determines when the archived log files are removed.

If your schedule allows, create offline backups on a regular schedule, perhaps once a week. Taking weekly offline backups can reduce the time required to rebuild table spaces or the database if you need to do so. Create offline backup images to removable media or storage that is managed by Tivoli Storage Manager. You should keep the backup images in a safe place until the next time that you create an offline backup image of the table space or database. If your schedule does not allow enough time to take offline backups (that is, your system must always be available to users), then you should take online backups on a regular schedule.

"ARSDB" on page 235 shows examples of scheduling the database backup command and provides details about the parameters and options that you can specify to backup the database.

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

Using Tivoli Storage Manager to maintain backup images

You can use Tivoli Storage Manager to maintain DB2 backup image files in archive storage. This eliminates the need for you to maintain the DB2 backup image files on disk. Tivoli Storage Manager can maintain the incremental table space backups and full database backups that you create with the ARSDB program.

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides information to help you configure Tivoli Storage Manager to maintain the DB2 backup image files. See "ARSDB" on page 235 for details about how to use the ARSDB program to backup the database to storage that is managed by Tivoli Storage Manager.

Backup to a tape device

If you plan to backup the database to a tape device, then you may need to configure the Buffer Size Limit in DB2. If you plan to backup the database to a tape device and specify a variable block size, then you must configure the DB2 buffer size to a value that is less than or equal to the maximum block size limit for the backup device. For maximum performance, you should set the buffer size to the maximum block size for the backup device. See the DB2 documentation for

details. Contact the IBM support center if you have questions.

Creating a full offline backup of the database

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

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

The ARSDB program sends processing messages to the system log. After the backup completes, you can open the System Log folder to search for and view the messages generated by the backup process.

On UNIX servers, you can run the ARSDB program from a terminal window to create a full offline backup of the database on tape. For example:

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

To run the ARSDB program on a Windows server, do the following:
1. Start → Programs → IBM Content Manager OnDemand for Windows → Command Window.
2. Enter the command and runtime parameters at the prompt. For example:
   ```
   arsdb -v -y \\.\Tape0
   ```

See "ARSDB" on page 235 for details about the ARSDB program, parameters, and options. The *DB2 Administration Guide* provides details about backing up a database.

Database backup in Windows

The Content Manager OnDemand configurator program that is provided for Windows servers allows for the scheduling of database backups. You can perform a backup while the database is either *online* or *offline*.
- If the backup is to be performed online, other applications or processes can continue to connect to the database, as well as read and modify data while the backup operation is running.
- If the backup is to be performed offline, only the backup operation can be connected to the database; other Content Manager OnDemand services and the rest of your organization cannot connect to the database while the backup task is running.

To schedule an offline backup with the configurator program, you must do the following:
1. Manually disconnect all other processes from the database before the backup task is scheduled to begin. This includes stopping the Content Manager OnDemand LibSrvr, MVSD Server, and Load Data services on the library server. In addition, if you load data to the library server from another object

server, then you should manually stop the Content Manager OnDemand ObjSrvr and Load Data services on the object server.

2. Run the offline backup.
3. Verify that the offline backup completed successfully.
4. Manually restart the Content Manager OnDemand LibSrvr service and the Content Manager OnDemand MVSD Server and Content Manager OnDemand Load Data services on the library server. If you stopped Content Manager OnDemand services on an object server, manually restart the services.

Database logging

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

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

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

Using Tivoli Storage Manager to maintain archived log files

You can use Tivoli Storage Manager to maintain the DB2 archived log files. This eliminates the need for you to maintain the DB2 archived log files on disk.

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides information to help you configure the system to use Tivoli Storage Manager to maintain the DB2 archived log files.

Database recovery

This section provides an overview of the different recovery methods that you can use in the event that there is a problem involving the database.

Note: Before you begin using the system, ask your IBM representative about the strategies that are available to you when there are problems with the database. You should also speak with a DB2 specialist to help implement a backup and recovery plan that is best suited to your business and operating environment. The *DB2 Administration Guide* provides details about database backup and recovery.

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

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

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

  This type of recovery is done with the DB2 RESTART DATABASE command. If you want this type of recovery to occur in every case of a failure, you can use the AUTOMATIC RESTART ENABLE database configuration parameter. The default for this configuration parameter is that the RESTART DATABASE routine will be started every time it is needed. Once enabled, you do not need to do anything to have this command done at the time of a failure.

- The second type of recovery deals with corruption of the Content Manager OnDemand database and is usually caused by media failure. For example, one of the disk storage volumes that belongs to the database volume group becomes damaged and unusable. To recover from this type of failure, an administrator must intervene to recover the database.

  The combination of the DB2 log files and a full backup copy of the database can be used to recreate the Content Manager OnDemand database to any particular point in time. First, the latest full backup image of the database rebuilds the database to a point in time. Then, a roll-forward recovery restores all of the units of work that occurred since the backup image was created. This allows you to restore the database to a state identical to the time of the failure.

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

Factors affecting recovery

To decide which database recovery method to use, you should consider the following:

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

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

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

- How much time is spent associated with recovery?

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

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

- How much storage space can you allocate for backup copies and archived log files?

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

- Table space level or full database level backup.

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

  IBM strongly encourages you to contact the IBM support center to help you with a backup and recovery plan that includes table space backup and recovery.

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

Document storage

This section is organized as follows:

- Defining document storage management
- Migrating documents
- Removing documents

Note: This section describes how to do some of the storage management tasks, but you will need other Content Manager OnDemand information and your archive storage manager product information to do others.

Defining document storage management

The document storage management definitions that you create on the library server determine where and when Content Manager OnDemand stores documents and how it maintains them.

Figure 14 shows how the components of document storage management work together to load documents and migrate them from one storage location to another.
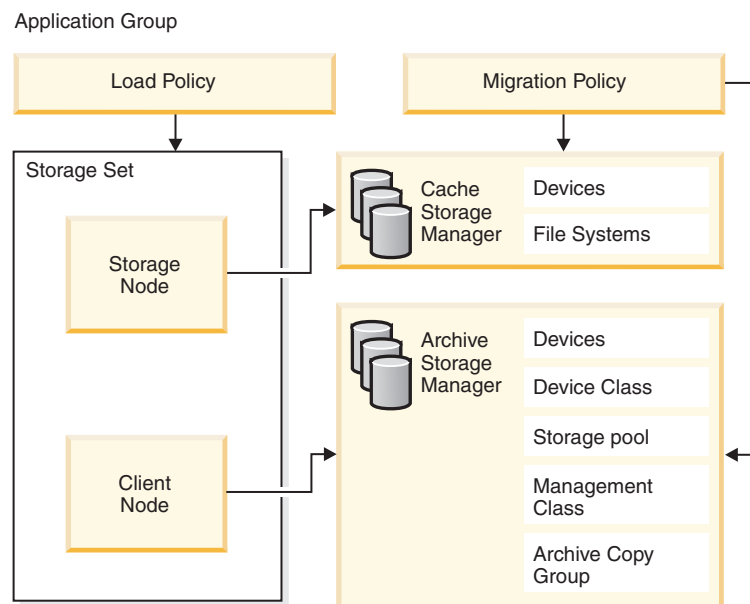
Figure 14. Document storage management

When you load a document into Content Manager OnDemand, you assign it to an application group. The application group is the last document storage management component that you define, because it requires a storage set and storage manager definitions, which you must create first. The load policy identifies the storage set and determines where documents should be loaded. You assign each application group to a storage set. The nodes in the storage set identify the object server on which documents are loaded. You can load documents into cache storage, into archive media, or into both cache storage and archive storage. The cache storage manager maintains documents temporarily on disk. The cache storage manager uses a list of file systems to determine the devices available to store and maintain

documents. The archive storage manager maintains documents on optical and tape storage. The archive storage manager uses devices, a device class, a storage pool, a management class, and an archive copy group to determine where to store documents and how long to maintain them. Depending on the load policy, documents may remain where the loading program put them for the number of days that are specified by the migration policy. After a document ages for the specified number of days, the migration process can move it from cache storage to archive storage.

Application groups

The application group is the last component that you must define because it requires a storage set and storage manager definitions. The application group provides a way to group related documents. All documents in the application group are loaded in the storage nodes that are part of the storage set to which the application group is assigned. All documents in the application group migrate according to the rules that are defined for the application group's migration policy.

Use the administrative client to create the application groups that determine the document storage for your documents. You typically define one application group for each set of your documents that have similar storage requirements. For example, documents that must be retained for a specific length of time, in specific storage locations and stored on specific types of media.

Load policy

A load policy contains the rules for loading documents into an application group. It requires one or more storage sets, which you must create first. The load policy determines if documents are loaded into cache storage, archive storage, or both. If the load policy causes documents to be stored only in cache storage, then the migration policy specifies when (or if) documents are copied to archive storage.

You define the load policy when you create the application group. The following properties on the Storage Management page comprise the load policy:
- Storage Set. Determines where documents will be loaded.

  Note: If you specify Cache Only, then documents can be loaded into cache storage only.
- Cache Data. Determines if documents will be loaded into cache storage.

  Note: If the storage set is a cache-only storage set, then documents must be loaded into cache storage.
- Migrate Data from Cache. If you specify When Data is Loaded, then documents will be loaded into archive storage. (Migration is disabled for the application group.)

Migration policy

Migration is the process of copying documents from cache storage to archive storage as controlled by the rules of the application group's migration policy. However, because a document is eligible to be migrated does not mean that it will be migrated. Other factors affect migration, such as the frequency with which you run migration processing (migration cannot take place until you run migration processing).

A migration policy contains the rules for migrating the documents in an application group. Migration requires an archive storage manager and its associated devices, storage pools, and so forth, which you must install and configure before you begin migrating documents.

The migration policy determines how long a document stays in cache storage and, through the storage set, where the document will be copied to next. The client node in the storage set identifies the next location.

You define the migration policy when you create the application group. The following settings on the Storage Management page comprise the migration policy:

- Storage Set. Determines the next location for documents.

  Note: If you specify Cache Only, then migration is disabled for the application group.
- Migrate Data From Cache. Determines when documents are eligible to be migrated.

  Note: If you specify No or When Data is Loaded, then migration is disabled for the application group.

Cache storage manager

The cache storage manager is the interface between the object server and the disk storage system. The cache storage manager maintains documents temporarily on disk. Before loading documents, you must identify the file systems that the cache storage manager can use to store and maintain documents. You must define at least one storage set for each object server. Documents migrate from cache storage to archive storage based on the migration policy that is defined for the application group. The cache storage manager can delete documents after they exceed the **Cache Document Data for n Days** or **Life of Data**, whichever occurs first. See "Removing documents" on page 47 for more information.

Archive storage manager

The archive storage manager is the interface between the object server and an optical or tape storage system. The archive storage manager maintains a backup or long-term copy of documents. Before loading documents, you must configure your archive storage devices and define storage pools, client nodes, and management classes to the archive storage manager. The management class determines how long documents remain in archive storage. The archive storage manager can delete documents after they exceed the **Retention Value** specified for the management class. See "Removing documents" on page 47 for more information.

Migrating documents

Content Manager OnDemand provides automatic migration to copy documents from cache storage to archive storage (for documents that were not loaded to archive storage) and to make documents eligible for deletion to maintain free space in cache storage file systems. Migration helps to ensure that there is sufficient free space in the cache storage file systems, where faster devices can provide the most benefit to your users.

Important: If you use migration to copy documents to archive storage (that is, you do not load documents to archive storage), then you should run migration

processing on a regular schedule to make sure that a backup copy of your documents gets created as soon as practically possible. If you defer the migration of documents to archive storage and cache storage were to become corrupted, then you could be left without a backup copy of your documents.

You control automatic migration processing by scheduling the ARSMAINT program to run with the appropriate options. See "ARSMAINT" on page 291 for details about the options. See your operating system information for details about how to schedule tasks. You can also manually start migration processing by running the ARSMAINT program from the prompt.

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

- If you use **Next Cache Migration** to control when migration for an application group occurs, then the cache storage manager runs migration processing each time that you start the ARSMAINT program with the appropriate options.
- If you use **After n Days in Cache** to control when migration for an application group occurs, then a document must be stored in cache storage for at least the specified number of days before it is eligible to be migrated.

The ARSMAINT program migrates documents from each cache storage file system listed in the cache storage file system file.

The cache storage space that migrated documents occupy can be reclaimed by the cache storage manager after expiration processing completes. After you run migration processing, you should run expiration processing so that the cache storage manager can reclaim the cache storage space occupied by migrated documents.

Figure 15 shows an example of migration processing.
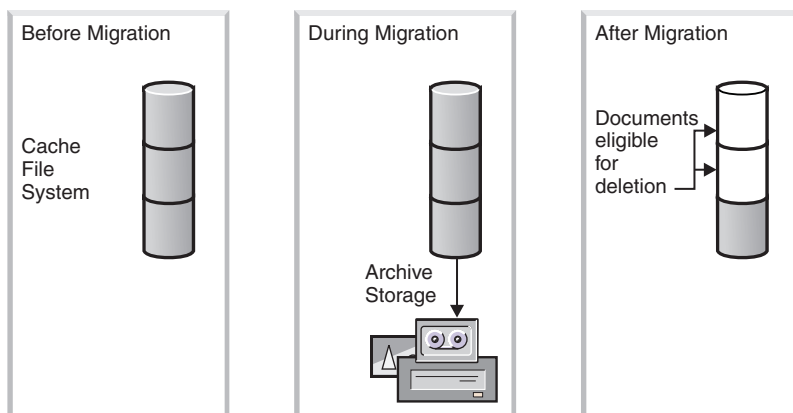
Figure 15. Migration Processing

In the example, assume that you have never run migration processing on this particular cache storage file system. The box on the left shows the cache storage file system before migration processing begins – it is quite full. (You would have ignored all of the "full cache file system" messages in the Content Manager OnDemand system log.) The box in the middle shows what happens during migration – the process of copying documents that are eligible to be migrated to archive storage. The box on the right shows the cache storage file system when migration processing completes. The cache storage file system is still full, however,

some two-thirds of the documents are eligible to be removed. You need to run expiration processing to remove documents from the cache storage file system, reclaiming at least some of the space occupied by migrated documents. After expiration processing completes, you will have free storage available in the cache storage file system to load additional documents.

Migration processing in the system log

When you run the ARSMAINT program, it saves messages about its activities in the system log. The types of messages saved in the system log depend on the options that you specify when you run the ARSMAINT program. The number of messages saved in the system log during a migration process depend on the options that you specify for the ARSMAINT program, the number of application groups and segments of data processed, and the number of cache storage file systems defined on the server. You will see one set of messages for each object server on which you run the ARSMAINT program. Table 1 lists the messages you could see in the system log following migration processing.

Table 1. Messages from the ARSMAINT program in the Content Manager OnDemand System Log

Message Number	Message Information	Explanation
110 Cache Migration	(Date) (Server)	About to begin cache migration on the specified server. Migration processing uses the specified date (the default is "today" in internal format).
197 Cache Migration	(ApplGrp) (ObjName) (Server)	One of these messages for each storage object migrated to archive storage. Migration copies a storage object if its "After n Days in Cache" period has passed or the application group uses the "Next Cache Migration" migration method.
124 Filesystem Statistics	(filesystem) (% full) (server)	One of these messages for each cache file system on the server. Information only to report the percentage of space used in the file system.

Important: In addition to the messages listed in Table 1, you should monitor the system log every day for messages that indicate your cache storage file systems are becoming full. The ARSMAINT program automatically saves a message in the system log when the amount of space used in a cache storage file system exceeds a threshold. The default threshold is 95%. You can specify a different threshold by using the **-f** parameter when you run the ARSMAINT program.

Removing documents

Documents *expire* (are eligible for removal) because their cache expiration date or archive retention period has passed. Expired documents can then be removed by the storage managers. The cache storage manager identifies documents for removal by using the application group's expiration policy and high and low expiration thresholds. The archive storage manager marks documents for removal based on the criteria defined in the archive copy group.

Documents expire from cache storage when they reach their cache expiration date. If a document's cache expiration date is less than its Life of Data period, then the document is simply removed from cache storage. Subsequent requests for the document are satisfied by the archive storage manager. When the document

reaches its Life of Data period, information about it is removed from the Content Manager OnDemand database (the document can no longer be retrieved). When the document's archive retention period has passed, information about it is removed from the archive storage manager database.

Because a document is eligible to be removed does not mean that it will be deleted from storage. The cache storage manager does not delete expired documents from storage until expiration processing runs. During expiration processing, the archive storage manager deletes information about expired documents from its database. However, the actual documents remain on archive media until such time that the space that they occupy is reclaimed.

Important: Content Manager OnDemand and the archive storage manager delete documents independently of each other. Each uses their own criteria to determine when documents expire and should be removed from the system. Each uses their own utilities to remove documents. However, for final removal of documents from the system, you should specify the same criteria to Content Manager OnDemand and the archive storage manager. The Life of Data, which is used by Content Manager OnDemand, and the Retention Period, which is used by the archive storage manager, should be the same value.

Removing documents from cache storage

The expiration policy determines when documents are eligible for deletion from cache storage. You define the expiration policy when you create the application group. The following properties on the Storage Management page comprise the expiration policy:

- Cache Document Data For n Days. The length of time in days to keep documents in cache storage. The documents include documents that are already in the cache and any documents that are subsequently loaded. After a document reaches this value, it is eligible to be deleted from cache storage.
- Life of Data. The length of time in days to maintain documents on the system.

  Note: If you specify Never Expire, then expiration processing is disabled for the application group.
- Expiration Type. Determines whether one or more documents are eligible to be deleted at a time. For example, the **Segment** expiration type means that a *segment* of data (unless you specify otherwise, 10 million documents) can be deleted at a time.

  Note: This is the first time that segment has been mentioned. Up to now, documents and reports have been discussed, which are the data objects that most people associate with the Content Manager OnDemand system. However, administrators who maintain the system work primarily with segments, which represent many documents, and *storage objects*, which are containers of compressed documents that are maintained by the storage managers.

The cache storage manager does not delete expired documents from cache storage 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 runs periodically so that the cache storage manager can reclaim the space that is occupied by expired documents.

You control automatic expiration processing by scheduling the ARSMAINT program to run with the appropriate options. See"ARSMAINT" on page 291 for details about the options. See your operating system information for details about how to schedule tasks. You can also manually start expiration processing by running the ARSMAINT program from the prompt.

The ARSMAINT program uses expiration thresholds to control when expiration processing begins and ends. The thresholds are set as levels of the space that is used in a cache storage file system, expressed as a percent of total space available in the file system. For each cache storage file system, the ARSMAINT program compares the high threshold with a calculation of the amount of data stored in the file system as a percent of the actual data capacity of the storage volumes that belong to the file system. When the amount of data stored in a cache storage file system exceeds the high threshold, expiration begins. The ARSMAINT program deletes documents from the file system until the amount of space used in the cache storage file system falls below the low expiration threshold. The ARSMAINT program expires documents from each cache storage file system listed in the cache storage file system file. You can use the defaults for the expiration thresholds, or you can change the threshold values to identify the minimum and maximum amount of space for your cache storage file systems.

Figure 16 shows an example of expiration processing.
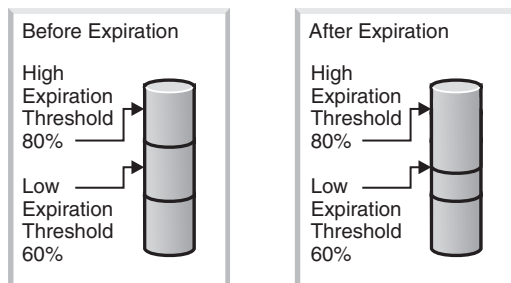
Figure 16. Expiration Processing

This example uses the cache storage file system from the migration example in Figure 15 on page 46. Some two-thirds of the file system contains documents that are eligible to be removed. When you run the ARSMAINT program, it first determines that the cache storage file system's capacity is equal to or exceeds the high threshold. The ARSMAINT program can then begin deleting documents from the file system, beginning with the oldest documents. After the ARSMAINT program deletes the documents that have the oldest date, it checks the low migration threshold. If the amount of space that is used in the file system is now below the low expiration threshold, then expiration ends. If not, then the ARSMAINT program deletes the next oldest documents, and the process continues. In the example, expiration processing ends before all of the eligible documents have been removed. That's typically OK for two reasons:

- The expiration process has probably reclaimed enough space to load new documents. (In our example, that is certainly true; some 40 percent of the cache storage file system is now free space.) If not, you need to check your high and low thresholds or add more storage volumes to the cache storage file system.
- Because a document is eligible to be removed from cache storage does not always mean that it is a good thing to do so. For example, suppose you copy a document to cache storage for 60 days and to archive media for two years. After 60 days, the document is eligible to be removed from cache storage. However,

your users continue to access the document on a regular basis for 90, or even 120, days. With the correct set of high and low thresholds, you can probably guarantee that the document will remain in cache storage for another 30 or more days beyond its expiration date, where faster devices can provide the most benefit to your users. (Of course, you could just change the load policy, but that's another story.)

Expiration processing in the system log

When you run the ARSMAINT program, it saves messages about its activities in the system log. The types of messages saved in the system log depend on the options that you specify when you run the ARSMAINT program. The number of messages saved in the system log each time that expiration processing runs depends on the options that you specify for the ARSMAINT program, the number of application groups and segments of data processed, and the number of cache storage file systems defined on the server. You will see one set of messages for each object server on which you run the ARSMAINT program. Table 2 lists the messages you could see in the system log following expiration processing.

Table 2. Messages from the ARSMAINT program in the Content Manager OnDemand System Log

Message Number	Message Information	Explanation
109 Cache Expiration	(Date) (Min%) (Max%) (Server)	About to begin cache expiration processing on the specified server. Migration processing uses the specified date (the default is "today" in internal format). Expiration processing begins on each cache file system that exceeds the Max% (default 80%) and ends when the free space available in the file system falls below the Min% (default 80%).
196 Cache Migration	(ApplGrp) (ObjName) (Server)	One of these messages for each storage object deleted from cache storage. A storage object is eligible to be deleted when its "Cache Document Data for n Days" or "Life of Data" period has passed, whichever occurs first.
124 Filesystem Statistics	(filesystem) (% full) (server)	One of these messages for each cache storage file system on the server. Information only to report the percentage of space used in the file system.

Important: In addition to the messages listed in Table 2, you should monitor the system log every day for messages that indicate that your cache storage file systems are becoming full. The ARSMAINT program automatically saves a message in the system log when the amount of space used in a cache storage file system exceeds a threshold. The default threshold is 95%. You can specify a different threshold by using the **-f** parameter when you run the ARSMAINT program.

Removing documents from archive storage

Important: Removing a document from archive storage means that the backup or long-term copy of the document will be deleted from the system. You typically remove documents from archive storage when you no longer have a business or legal requirement to keep them.

A management class contains an archive copy group that specifies the criteria that makes a document eligible for deletion. Documents become eligible for deletion under the following conditions:

- Administrators delete documents from client nodes
- An archived document exceeds the time criteria in the archive copy group (how long archived copies are kept)

The archive storage manager does not delete information about expired documents from its database until expiration processing runs. You can run expiration processing either automatically or manually by command. You should make sure that expiration processing runs periodically to allow the archive storage manager to reuse storage pool space that is occupied by expired documents. When expiration processing runs, the archive storage manager deletes documents from its database. The storage space that these documents occupy then becomes reclaimable. See "Reclaiming space in storage pools" for more information.

You control automatic expiration processing by using the expiration processing interval (EXPINTERVAL) in the server options file (dsmserv.opt). You can set the option by editing the dsmserv.opt file (see the *Installation and Configuration Guide* for details).

If you use the server option to control when expiration processing occurs, the archive storage manager runs expiration processing each time that you start the server. After that, it runs expiration processing at the interval that you specified with the option, measured from the start time of the server.

You can manually start expiration processing by issuing the EXPIRE INVENTORY command. Expiration processing then deletes information about expired files from the database. You can schedule this command by using the DEFINE SCHEDULE command. If you schedule the EXPIRE INVENTORY command, set the expiration interval to 0 (zero) in the server options so that the archive storage manager does not run expiration processing when you start the server. You can control how long the expiration process runs by using the DURATION parameter with the EXPIRE INVENTORY command.

Reclaiming space in storage pools

Space on a storage pool volume becomes reclaimable as documents expire or are deleted from the volume. For example, documents become obsolete because of aging.

The archive storage manager reclaims the space in storage pools based on a reclamation threshold that you can set for each storage pool. When the percentage of space that can be reclaimed on a volume rises above the reclamation threshold, the archive storage manager reclaims the volume. The archive storage manager rewrites documents on the volume to other volumes in the storage pool, making the original volume available for new documents.

The archive storage manager checks whether reclamation is needed at least once per hour and begins space reclamation for eligible volumes. You can set a reclamation threshold for each storage pool when you define or update the storage pool.

During reclamation, the archive storage manager copies the files to volumes in the same storage pool unless you have specified a reclamation storage pool. Use a

reclamation storage pool to allow automatic reclamation for a storage pool with only one drive. See your archive storage manager documentation for details.

After the archive storage manager moves all documents to other volumes, one of the following occurs for the reclaimed volume:

- If you have explicitly defined the volume to the storage pool, the volume becomes available for reuse by that storage pool
- If the volume was acquired as a scratch volume, the archive storage manager deletes the volume from its database

Important: See your archive storage manager documentation for more information about reclamation processing, including choosing a reclamation threshold, reclaiming volumes in a storage pool with one drive, reclamation for WORM optical media, reclamation for copy storage pools, and reclamation of off-site volumes.

Managing Tivoli Storage Manager storage

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

Overview

This part of the book provides a brief overview of tasks that Tivoli Storage Manager administrators need to do to manage Tivoli Storage Manager storage. The *Tivoli Storage Manager Administrator's Guide* presents the details of the tasks and the concepts that you need to understand to complete them. The *Installation and Configuration Guide* is another important source of information. When you installed and configured an Content Manager OnDemand server with Tivoli Storage Manager, you completed many of the tasks described here: configuring devices and defining them to Tivoli Storage Manager, defining policy management information, preparing storage volumes, registering client nodes, and increasing the size of the database and recovery log.

The following topics provide more information:
- Using magnetic disk devices with Tivoli Storage Manager
- Using removable media devices with Tivoli Storage Manager
- Managing removable media operations
- Defining drives and libraries
- Defining device classes
- Managing storage pools
- Managing storage pool volumes
- Managing policies
- Managing client nodes
- Automating server operations
- Managing server operations
- Managing the database and recovery log
- Monitoring the server
- Protecting and recovering your data

Using magnetic disk devices

In an Content Manager OnDemand system, the primary use of magnetic disk devices with Tivoli Storage Manager is the storage of the database and recovery log. The Tivoli Storage Manager database contains information needed for server operations and information about data that has been backed up, archived, and space-managed. The database contains pointers to the locations of all client files in the Tivoli Storage Manager storage pools. Changes to the database are recorded in the recovery log in order to maintain a consistent database image. The recovery log contains information about updates that have not yet been committed to the database. If the database is unusable, the entire Tivoli Storage Manager server is unavailable. If a database is lost and cannot be recovered, the backup, archive, and space-managed data for that server is lost. Refer to "Managing the database and recovery log" on page 57 and "Protecting and recovering data" on page 58 for

steps that you can take to protect your database.

Using removable media devices

In an Content Manager OnDemand system, removable media devices can be used with Tivoli Storage Manager for the following purposes:

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

Tivoli Storage Manager allows you to use and reuse removable media to store data. You must prepare removable media for initial use by Tivoli Storage Manager. You also control how and when media are reused. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* shows examples of labeling removable media for initial use and checking storage volumes into a library. For detailed guidance and scenarios on configuring your removable media devices, see the *Tivoli Storage Manager Administrator's Guide*.

Managing removable media operations

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

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

For automated libraries, Tivoli Storage Manager works with the operating system and the library to accomplish volume mounts. Mount messages are not sent to an operator. However, information about problems with the library are still sent to the mount message queue. You can see these messages on administrative clients that have been started with either the mount mode or console mode parameter. However, you cannot use the Tivoli Storage Manager REPLY command to respond to these messages. You can get information about pending operator requests either by using the QUERY REQUEST command or by checking the mount message queue on an administrative client started in mount mode.

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

For most types of requests, such as volume mounts, the server detects when the operator performs the action. The operator does not usually need to respond to the Tivoli Storage Manager server carrying out the requested activity. However, sometimes the server cannot detect the completion of the requested action. When

the server requires a reply, the message that is displayed by the server requests that the operator reply when the activity has been completed. For example, a request to mount a scratch volume requires that the operator reply when a scratch volume has been placed in the drive. Tivoli Storage Manager waits for a reply to prevent the use of the wrong volume.

For information about managing removable media operations, see the *Tivoli Storage Manager Administrator's Guide*.

Defining drives and libraries

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides examples of defining drives and libraries. For detailed information about defining drives and libraries, see the *Tivoli Storage Manager Administrator's Guide*.

Defining device classes

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides examples of defining device classes. See the *Tivoli Storage Manager Administrator's Guide* for detailed information about device classes.

Managing storage pools

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides examples of defining primary storage pools. For more information about copy storage pools, see the *Tivoli Storage Manager Administrator's Guide*.

Managing storage pool volumes

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

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

- A scratch volume is a labeled volume that is empty or contains no valid data, and can be used to satisfy any request to mount a scratch volume. To support Content Manager OnDemand, you typically define scratch volumes to Tivoli

Storage Manager. Tivoli Storage Manager uses scratch volumes as needed, and returns the volumes to scratch when they become empty (for example, when all data on the volume expires).

- A private volume is a volume that is in use or owned by an application, and may contain valid data. Volumes that you define to Tivoli Storage Manager are private volumes. A private volume is used to satisfy only a request to mount that volume by name. When Tivoli Storage Manager uses a scratch volume, it changes the volume's status to private by defining it. Tivoli Storage Manager tracks whether defined volumes were originally scratch volumes. Volumes that were originally scratch volumes return to scratch status when they become empty.

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

Managing policies

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

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides examples of defining policies to support archiving documents, backing up DB2 files, and migrating index data. See the *Tivoli Storage Manager Administrator's Guide* for more information about establishing and managing policies.

Managing client nodes

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

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

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

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides examples of registering client nodes, defining client options for Content Manager OnDemand primary storage nodes, and registering administrators. See the *Tivoli Storage Manager Administrator's Guide* for more information about managing clients.

Automating server operations

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

See the *Tivoli Storage Manager Administrator's Guide* for information about scheduling Tivoli Storage Manager commands and operations.

Managing server operations

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

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

See the *Tivoli Storage Manager Administrator's Guide* for details about the day-to-day tasks involved in administering the server and about reports and information available to help you manage the server.

Managing the database and recovery log

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

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

Monitoring the server

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

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

Protecting and recovering data

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

- Mirroring, by which the server maintains one or more copies of the database or recovery log, allowing the system to continue when one of the mirrored disks fails. IBM recommends that you mirror at least one copy of the database and the recovery log to different physical storage volumes.
- Periodic backup of the database. IBM recommends that you schedule frequent backups of the database, after every load or system configuration change, or once a day. If you do not load reports every day or your system configuration does not change very often, you may be able to schedule backups less frequently, perhaps once a week.
- Periodic backup of the storage pools. To protect Content Manager OnDemand data stored in Tivoli Storage Manager, you may want to backup data in a primary storage pool to a copy storage pool. See your Tivoli Storage Manager information for assistance with configuring a copy storage pool.
- Recovery of damaged files.

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

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

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

Mapping Tivoli Storage Manager objects to OnDemand application group objects

You might need to reload data with files that are obtained from the Tivoli Storage Manager volume. For example, here is some typical Tivoli Storage Manager volume content:

```
OD6YNODE          Arch      /JCA          12      /RES/ 27120
OD6YNODE          Arch      /JCA          12      /DOC/ 39072FAAA
OD6YNODE          Arch      /LCA          15      /RES/ 7268
OD6YNODE          Arch      /LCA          15      /DOC/ 31844FAA1
```

In some instances, you might not be able to determine the application group name, or know how to search in the application group tables. You might also miss system log records to find out the load ID to extract the object file name. The challenge here is to determine and verify the object file name from the existing data.

The Tivoli Storage Manager file space names (for example, /JCA) correspond to the OnDemand application group three-letter identifier. With the three letter application group identifier, you can obtain the application group ID (AGID) by querying the ARSAG table.

You can use the application group ID to query the ARSSEG table for the corresponding application group table name or names. The table name is TABLE_NAME. The application group table name is something like AGIDn, where n is an integer. For example, JCA1.

The object file names are recorded in the DOC_NAME field of the application group table.

Important: If you delete an application group, the Tivoli Storage Manager filespace is also deleted, and all of the objects for that application group are removed regardless of their expiration parameters.

Exception: If you use data retention protection, the Tivoli Storage Manager filespace is not deleted, instead, Content Manager OnDemand deletes all application group objects.

Starting, halting, and restarting the server

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

Starting the server

To start the server, complete the following steps:
1. Change to the Tivoli Storage Manager server program directory.
2. Start the server:

       dsmserv

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

Important: You can capture Tivoli Storage Manager server console messages to a user log file with the Tivoli Storage Manager dsmulog utility. See the Tivoli Storage Manager documentation for more information.

Starting the Tivoli Storage Manager server command line interface

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

       dsmadmc

Tivoli Storage Manager provides you with ways to monitor processes and messages:

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

```
dsmadmc -consolemode
```

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

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

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

```
query actlog
```

Refer to the Tivoli Storage Manager documentation for more information about managing client sessions.

Halting the server

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

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

Restarting the server

To start the server after it has been halted, follow the instructions in "Starting the server" on page 59.

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

Using scratch and private volumes

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

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

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

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

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

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

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

Labeling storage volumes

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

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

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

Overwriting volume labels

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

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

Identifying volume labels

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

Labeling volumes one at a time

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

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

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

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

Where `od5000` is the volume label.

Searching the library

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

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

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

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

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

Adding storage volumes

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

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

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

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

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

Checking in volumes one at a time

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

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

To check in a private volume:

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

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

To check in a scratch volume:

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

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

Searching the library

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

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

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

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

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

Allowing swapping of volumes

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

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

Removing storage volumes

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

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

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

Removing volumes from a library

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

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

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

Returning volumes to a library

When you check out a volume that is defined to a storage pool, to make the volume available again, you must do the following:
1. Check in the volume for the library, with private status. Use the CHECKIN LIBVOL command with the STATUS=PRIVATE parameter.

2. Update the volume's ACCESS value. You must change the access from unavailable to read/write or read-only. Use the UPDATE VOL command with the ACCESS parameter.

Managing a full library

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

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

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

   ```
   update stgpool odaix1 ovflocation=room2948
   ```
2. When the library becomes full, use the MOVE MEDIA command.

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

   ```
   move media * stgpool=odaix1
   ```
3. Check in new scratch volumes, if needed.
4. As requested through Tivoli Storage Manager mount messages, check in volumes that Tivoli Storage Manager needs for operations. The mount messages include the overflow location of the volumes.

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

Offline storage of storage volumes

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

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

Protecting data with the data retention protection (DRP) protocol

To avoid the accidental erasure or overwriting of critical data, Content Manager OnDemand now supports the Tivoli Storage Manager Version 5.2.2 or later APIs related to data retention. Specifically:

data retention protection (DRP)
> Prohibits the explicit deletion of documents until their specified retention criterion is met. Although documents can no longer be explicitly deleted, they can still expire. **Attention:** DRP is permanent. Once it is turned on, it cannot be turned off.

Event-based retention policy

Retention based on an external event other than the storage of data. For Content Manager OnDemand, the retention event is the call to delete the data. A load, unload, application group delete, or expiration of data triggers the retention event.

Restriction: Content Manager OnDemand does not support *deletion hold*, a feature that prevents stored data from being deleted until the hold is released.

If you decide to use these policies in Tivoli Storage Manager, then the following scenarios result:

Table 3. Scenarios of using creation-based object expiration policy and event-based retention object expiration policy

	Creation-based object expiration policy	Event-based retention object expiration policy
Data retention protection off	Content Manager OnDemand issues a delete object command through the TSM API. Objects are deleted during the next inventory expiration. If an Content Manager OnDemand application group is being deleted, a `delete filespace` command is issued instead, and the objects are immediately deleted along with the filespace.	Content Manager OnDemand issues an event trigger command through the TSM API. The status of the objects affected are changed from PENDING to STARTED, and will be expired by TSM based on their retention parameters. If the retention parameters are set to NOLIMIT, then the objects never expire. If an OnDemand application group is being deleted, a `delete filespace` command is issued instead, and the objects are immediately deleted along with the filespace.

Table 3. Scenarios of using creation-based object expiration policy and event-based retention object expiration policy (continued)

	Creation-based object expiration policy	Event-based retention object expiration policy
Data retention protection on	Content Manager OnDemand issues no commands to TSM. The objects are effectively orphaned by Content Manager OnDemand and will be expired by TSM based on their retention parameters. If the retention parameters are set to NOLIMIT, then the objects never expire.	Content Manager OnDemand issues an event trigger command through the TSM API. The event status of the objects affected are changed from PENDING to STARTED and will be expired by TSM based on their retention parameters. If the retention parameters are set to NOLIMIT, then the objects never expire. If an Content Manager OnDemand application group is being deleted, then a `delete filespace` cannot be used with DRP on so the operation is treated the same as if a delete were indicated. The status of all the affected objects is changed from PENDING to STARTED, and they will be expired by TSM based on their retention parameters. Because this leaves the filespace entries in TSM, you must manually delete these entries when the filespace is empty (even with DRP on).

Recommendations:

- Set up the application groups to expire by load.
- Define the Tivoli Storage Manager archive copy groups to be event-based, and retain data for 0 days.
- Run the Tivoli Storage Manager inventory expiration regularly to ensure that expired data is cleaned up.

Additionally, Content Manager OnDemand supports the following devices:

IBM DR450 and DR550
> Disk-based system that contains a Tivoli Storage Manager running DRP.

EMC Centera
> Disk-based system, treated as a device by Tivoli Storage Manager. Tivoli Storage Manager must be running DRP.

Storage backup and recovery

Reports

Content Manager OnDemand can store copies of reports and resources in the cache storage and archive storage:

- The primary purpose of cache storage is short-term, high-speed storage and retrieval of reports. Cache storage consists of disk storage volumes maintained by Content Manager OnDemand on one or more object servers.
- The primary purpose of archive storage is long-term storage and retrieval of reports. The reports in archive storage can also be used as backup copies in the event that cache storage becomes corrupted or unavailable. Archive storage consists of optical or tape storage volumes managed by Tivoli Storage Manager.

Most customers configure the system to copy reports to cache storage and archive media at the same time, when they load a report into the system.

Content Manager OnDemand can retrieve a copy of a report from archive storage after the report has been deleted from cache storage or if the copy on cache storage is unavailable. However, you must configure the system to maintain multiple copies of reports. You must install and configure Tivoli Storage Manager, define storage devices to Tivoli Storage Manager, define policy management information to Tivoli Storage Manager, and configure Content Manager OnDemand to use archive storage. You configure Content Manager OnDemand to use archive storage by registering client nodes in Tivoli Storage Manager, defining storage sets that identify client nodes in Tivoli Storage Manager, assigning application groups to the storage sets, and configuring other storage management parameters in application groups.

Note: If you do not plan to copy reports to archive storage, then IBM recommends that you take regular backups of the file systems that comprise cache storage. However, if a media failure occurs or cache storage becomes corrupted, users cannot retrieve reports until the file systems are restored.

Cache storage

Cache storage is the primary, short-term storage location for reports.

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

Cache storage can be protected by maintaining it on high-availability storage devices. If no high-availability storage is available, IBM recommends that backups of reports in cache storage (the file systems) be taken on a regular basis.

Archive storage

The Content Manager OnDemand storage node identifies the object server and the client node in Tivoli Storage Manager where the primary copy of a report is maintained. Content Manager OnDemand retrieves the primary copy of the report from archive storage after the report has been deleted from cache storage. Customers with special business, legal, or performance reasons may want to

configure the system to maintain a backup (second) copy of their reports in archive storage. The backup copy can be used if the primary copy becomes corrupted or unavailable.

You must configure Tivoli Storage Manager to maintain multiple copies of reports. For example, you can define a *copy storage pool* to Tivoli Storage Manager. With a copy storage pool, Tivoli Storage Manager manages a backup copy of files that are stored in a primary storage pool independently and transparently to Content Manager OnDemand. The backup copy is stored in a copy storage pool that can be used to restore the original files if they become damaged, lost, or unusable. The copy storage pool can be assigned to the same library as the primary storage pool. However, most customers assign the copy storage pool to a different library. You can copy data from one or more primary storage pools to the same copy storage pool. Copy storage pools require additional space in the Tivoli Storage Manager database. A copy storage pool must reside on the object server where the primary storage pool resides.

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

The *Tivoli Storage Manager Administrator's Guide* provides details about defining and managing a copy storage pool, storage pool backup and recovery, and scheduling operations.

Installing the administrative client

System requirements

For administrative client system requirements, see: http://www.ibm.com/support/docview.wss?rs=129&uid=swg27016455, or search for 7016455 at http://www.ibm.com.

Installing the administrative client

About this task

The Setup program transfers the Content Manager OnDemand client files from the CD-ROM to each user's PC.

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

Running Setup

About this task

When you run the Setup program, the Setup screens show the names of the Content Manager OnDemand 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 user's PC

About this task

To install from the CD-ROM or to install from a network file server, follow these instructions:

Procedure

1. Insert the CD-ROM in the appropriate drive or obtain the drive letter of the network drive on which the Content Manager OnDemand software is located on the network file server.
2. From the Windows taskbar, click Start, and then choose Run.
3. Type x:\client\win32\setup (where x is the letter of your CD-ROM drive or the network drive).
4. 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 install on a network file server

About this task

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

To use automated install

About this task

Automated install allows administrators to standardize the Content Manager OnDemand installation for all users in an organization. It also allows administrators to install Content Manager OnDemand clients without the presence of users at their PCs. To read more about automated install, see the *IBM Content Manager OnDemand: Windows Client Customization Guide*.

Excluding the administrative client from the client installation package

About this task

Prerequisite: You need a file extraction and archiving utility, such as PKZIP or INFOZIP.

To exclude the administrative client from the client installation package, follow these steps:

Procedure

1. Download the `odwin32.zip` file from the ftp://service.software.ibm.com/ software/ondemand/fixes directory to a temporary directory, such as `c:\temp`.
2. Extract the files from the `odwin32.zip` file to a new working directory, such as `d:\ondemand`.
3. Delete the `arsadm32.exe` file from the `ars32` subdirectory of the working directory (for example, `d:\ondemand\ars32`).
4. Create a new ZIP file from the files in the working directory. Make sure that you use a file archiving utility that preserves the directory and file structure of the working directory.
5. Distribute the new ZIP file to your users. When your users run the `setup.exe` file, the installation program recognizes that the main administrative client file is not present. The option to install the administrative client will not be available to the user.

To uninstall

About this task

You can use Add/Remove Programs in Control Panel to remove the Content Manager OnDemand client from your PC. Add/Remove Programs removes the directories for Content Manager OnDemand and any references to Content Manager OnDemand in system files.

To run the uninstall:

Procedure

1. From the Windows taskbar, click Start. Choose Settings, and then choose Control Panel.
2. Double-click Add/Remove Programs.
3. From the list, select Content Manager OnDemand32.
4. 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.

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, 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.
- Place documents on hold so that the documents can be kept beyond their expiration dates.

Getting started
About this task

Starting the administrative client
About this task

To start the administrative client:

Procedure
1. Click **Start**.
2. Select **Programs**, then choose IBM Content Manager OnDemand32.
3. Click Content Manager OnDemand32 Administrator.
4. 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 OnDemand user ID and the name and version number of the 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 dialog box.
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 Windows server, you should select Attempt Unified Logon; otherwise, you should clear the Attempt Unified Logon check box. See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about unified logon.

8. If you are adding a Local server, specify a Directory and select an Operating System and a Database. See the online help for assistance.

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 Logon dialog box.

2. Type your Content Manager OnDemand userid and password in the spaces provided. Then click OK. **Note:** For a Local server, the built-in userid is `admin`; no password is assigned to the admin userid.

Results

Note: If the Content Manager OnDemand system is configured for unified logon and you are running the administrative client on a Windows system, the client attempts to automatically log on to the server by using the Windows logon account username. If the Windows logon account username exists in the Content Manager OnDemand user table, then the log on is successful. Otherwise, the client opens the Logon dialog box and you must enter a valid Content Manager OnDemand userid and password to continue. If you need to log on to a Windows server using an Content Manager OnDemand userid that is different than the Windows logon account username, you must use the Logon As command. To access the Logon As command, point to the server and click the right mouse button. If you are currently logged on to the server, you must log off the server before you can select the Logon As command. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about unified logon.

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 78 for details.

Changing a password
About this task

To change a user's password:

Procedure
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.

   Note: If your organization has not implemented the security user exit, then Content Manager OnDemand security verifies only the first eight characters that you type. If you type more than eight characters, then the client will display an error message. If your organization has implemented the security user exit, then you can type a maximum of 128 characters.
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" 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 following 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.

If you select `Permit Blank Password`, meaning that passwords are not required, then the valid password length is 0 (zero) to 128 characters, if your organization uses the security user exit; otherwise, the maximum valid value that you can specify is 8 (eight) characters.

To set a specific minimum password length, select `At Least __ Characters` and type a number in the space provided. If your organization has implemented the security user exit, then the value can be from 1 (one) to 128. Otherwise, the value can be from 1 (one) to 8 (eight).

Note: The password authentication that is built into Content Manager OnDemand verifies only the first eight characters that are entered by the user. The additional characters are provided for customers that choose to implement their own password security by using the security user exit.

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.

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 time out 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 time out 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 point to process the system log messages. This exit allows a user-written program to filter the messages that Content Manager OnDemand saves in the system log and take action when a particular event occurs. For example, you could provide a user-written program to send a message to a security administrator when someone attempts and fails to log on to the system.

If you plan to migrate index data to archive storage, you should configure the system to save application group messages in the system log and process them with a user-written, system log user exit program. You should design the 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 manually import a copy of the table or tables that are required from archive storage to the database.

See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about 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

This section specifies which types of annotations (referred to as "notes" in the OnDemand client) can be added by a user. This selection applies 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 annotation that is selected as the default.

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.

LDAP Authentication

Specify whether you want to use LDAP (Lightweight Directory Access Protocol) authentication in your OnDemand server.

Select the **Enable** check box to use LDAP authentication. After LDAP authentication is enabled, OnDemand server makes an authentication request to the LDAP server every time it receives a login request from the client, and processes the client request only after the user information is verified by the LDAP server.

Clear the **Enable** check box to disable LDAP authentication.

Setting system parameters

To set the system parameters for an 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. To change the Minimum Password Age, select the appropriate option. If you select Expires In, enter the number of days in the space provided.
5. To change the Minimum Password Length, select the appropriate option. If you select At Least, enter the number of characters in the space provided.
6. To change the Inactivity Time Out, select the appropriate option. If you select Time Out In, enter the number of minutes in the space provided.
7. To choose a System Logging, User Exit Logging, or Login Processing option, select the check box next to the item. Content Manager OnDemand places a check mark in the check box. (To clear a System Logging, User Exit Logging, or Login Processing option, select the check box next to the item. Content Manager OnDemand removes the check mark from the check box.)
8. Specify which types of annotations can be used by a user. This selection applies to all users with authority to add annotations in the system. There are three annotation types available:

   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.

   In the Default Annotation Type section, specify the annotation that is selected as the default.
9. Specify whether the administrative client displays the System Log Comments window when you perform an add, update, or delete operation.
10. Specify whether you want to use LDAP (Lightweight Directory Access Protocol) authentication in your Content Manager OnDemand server.
11. To generate a summary of the system parameters and display the information in a window where it can be viewed and printed, click Summary.
12. 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.)

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 4 lists the start up parameters for the administrative client.

Table 4. Administrative client start up parameters

Parameter	Purpose	Example
/1 *location*	Identifies the drive and full path name of the national language environment program file directory.	/1 D:\Program Files\IBM\Content Manager OnDemand32\Locale\Enu
/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
/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
/P *password*	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
/S *server*	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

Table 4. Administrative client start up parameters (continued)

Parameter	Purpose	Example
/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
/U *userid*	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

Changing start up parameters
About this task

To change start up parameters:

Procedure

1. Click Start ⟶ Programs.
2. Point to IBM Content Manager OnDemand32.
3. Point to Content Manager OnDemand32 Administrator and click the right mouse button.
4. From the pop-up menu, select Properties.
5. Click the Shortcut tab.
6. In the Target field, enter the startup parameters. Make sure that you enter any startup parameters after the string:

   ```
   "D:\Program Files\IBM\Content Manager OnDemand32\ARSADM32.EXE"
       /1 D:\Program Files\IBM\Content Manager OnDemand32\LOCALE\ENU
   ```

7. 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 OnDemand splash screen or About window
About this task

When the administrative client is first started, an OnDemand splash screen is displayed.

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

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

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

Examples

The following program properties could be used to set up the administrative client so that when a person starts the program, the name `Customer Service` is displayed in the title bar of the administrator window.

```
"D:\Program Files\IBM\OnDemand32\ARSADM32.EXE"
  /1 D:\Program Files\IBM\OnDemand32\LOCALE\ENU /T "Customer Service"
```

The following program properties could be used to set up the administrative client so that when a person starts the program, Content Manager OnDemand displays only users and folders in the navigator pane. When the user selects folders, Content Manager OnDemand displays only the Permissions and Field Information pages.

```
"D:\Program Files\IBM\OnDemand32\ARSADM32.EXE"
  /1 D:\Program Files\IBM\OnDemand32\LOCALE\ENU /I /D 2
```

The following program properties could be used to set up the administrative client so that when a person starts the program, Content Manager OnDemand attempts to log on to the specified server, without displaying the Logon dialog box. If the logon attempt is not successful, then Content Manager OnDemand opens the Logon dialog box to allow the user to log on to the system.

```
"D:\Program Files\IBM\OnDemand32\ARSADM32.EXE"
  /1 D:\Program Files\IBM\OnDemand32\LOCALE\ENU /S peak /U admin /P pw
```

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 default directory for the temporary files that are created by the administrative client is `C:\Documents and Settings\user\Application Data\IBM\OnDemand32\REPORT`, where *user* is the user ID that uses the *REPORT* directory.

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. 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.

Important: The Report Wizard processes input files that contain line data or PDF data. You must select a sample input file to proceed.

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. See the Field Information page in application groups for detailed information about application identifiers.

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 the Report Wizard icon 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 the Report Wizard icon on the toolbar

Follow the on-screen instructions to add the report.

Using the Report Wizard

The screens in the Report Wizard are described below. On most screens, standard options will already be selected for you. Unless you have a clear reason not to, accept the defaults.

Note: Depending on how you use the Report Wizard, you may not see all of the screens described below.

Introduction screen

Provides a brief explanation of the Report Wizard. If you are processing line data, first decide whether you want to convert the input line data to AFP. Then select a file that contains a sample of the actual report data.

Figure 17. Report Wizard

After you select a file, the Report Wizard opens the Indexer Properties dialog box to the Data Format page. The Data Format page is where you can change options that the Report Wizard uses to display the report. For example, the Maximum pages to display field lets you limit the number of pages that the Report Wizard

reads into the Report window. For example, assume that the input report contains thousands of pages. However, you need only the first 20 pages of the report to define the indexing information. You can select the _____ Pages field and enter 20 in the space provided.

When you click OK, the Report Wizard reads the data into the Report window.

Report window

Displays the sample file 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 have finished defining the indexing, database, and folder information, be sure to save your changes.

Managing data screen

When you load a report into the system, you can specify that you want report data to be stored in Large Objects. 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.

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.

Administration concepts

Introduction

You can use the administrative client to maintain Content Manager OnDemand users, groups, printers, storage sets and storage nodes, application groups, applications, folders, and cabinets.

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 an 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 an 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.

DB2 Content Manager OnDemand can store a maximum of `1,080,000` separate user IDs.

An Content Manager OnDemand userid does not necessarily have to identify an individual by name. However, for accounting purposes, most customers assign an 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, 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 quote), 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 78 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, 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 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 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 cabinets.

Create Holds
An optional authority for users and user administrators. Lets the user create 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 quote), or blank character
- Can be mixed case. However, the case does not create a unique name (`LaGuarde` is the same as `laguarde`)
- 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

An 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 submitting a print job to an IBM Infoprint Server print queue. Infoprint is an intelligent printer driver that provides AFP capabilities for Content Manager OnDemand servers. These capabilities mean that you can use electronic forms, images, graphics, and typographical fonts in the documents that you print. Infoprint accepts input data streams, such as AFP and S/390 line data, prepares the data for the destination print device, and manages print jobs that are queued for printing, in process, or on the output device.

When a user selects an item and chooses the server print command, Content Manager OnDemand retrieves a copy of the item and places it on the queue (or printer) identified by the server printer. The data stream can include application print options. Infoprint processes the item and the print options, sending the data to the appropriate physical device. While Infoprint makes no distinction between print and FAX devices, Content Manager OnDemand manages these devices differently. When the user chooses a server printer that identifies a *Printer with Information*, the user can specify information that Content Manager OnDemand uses to generate a header page that is sent to the printer along with the document. When the user chooses a server printer that identifies a FAX device, Content Manager OnDemand generates a FAX cover page and places the cover page and the item on the appropriate queue (or printer).

If your application group contains multiple applications, and your document list contains a document from each of the applications, when you use the server print function to print all of the documents, one print request is generated for each document.

Content Manager OnDemand supports two types of server print devices: a FAX machine and a physical printer. A server print device can be physically connected to the library server or attached to another workstation on the network. Server print devices are managed by Infoprint.

A server printer names a queue in the AIX operating system. For AIX servers, the queue must be defined on the library server. The queue must be connected to an Infoprint print queue. For HP-UX, Sun Solaris, and Windows servers, in addition to identifying the queue name when you define the printer with the administrative client, you must also configure the ARSPRT file (on HP-UX and Sun Solaris servers) or the ARSPRT.BAT file (on Windows servers) with the TCP/IP host name or TCP/IP address of the workstation on which Infoprint is running. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for information about configuring Content Manager OnDemand server printing.

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.

You typically define one or two storage sets for an Content Manager OnDemand system, based on the storage requirements of the reports that you plan to store in Content Manager OnDemand. 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 add an application group to the system, 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.

If the data will be stored in archive storage, then the primary storage node must identify a client node in storage that is managed by Tivoli Storage Manager. The client node must be registered in a Tivoli Storage Manager policy domain. The policy domain determines the archive storage devices that will hold the data and how long Tivoli Storage Manager will maintain the data. In addition, the retention period of the files in Tivoli Storage Manager must be the same as the life of the data and indexes in Content Manager OnDemand.

If you use Tivoli Storage Manager to maintain data stored in the system and you need the system to maintain two copies of reports that are stored in archive storage, IBM recommends that you configure a copy storage pool in Tivoli Storage Manager. See your Tivoli Storage Manager information for assistance with configuring a copy storage pool.

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. Application group data can be managed by Content Manager OnDemand or it can be managed independently of Content Manager OnDemand.

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.
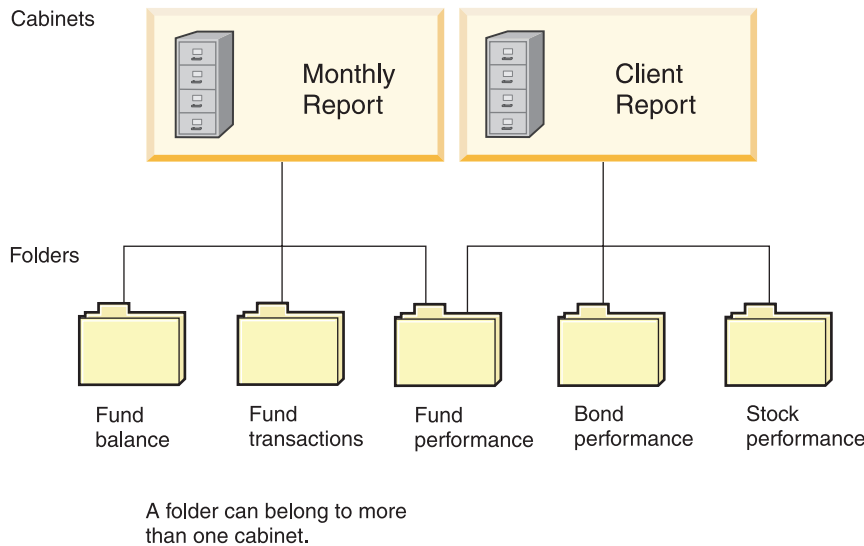
Figure 18. Cabinets and folders

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 an 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 102 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 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 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:

Application Group
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 5, a user that belongs to both groups can open the Student Bills and Student Transcripts folders.

Table 5. 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 6.

Table 6. 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 6, 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.

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 an 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.

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 an Content Manager OnDemand system. The tasks are add, update, delete, copy, export, create a summary, and view properties.

User Administrator
A user administrator has the authority to perform all tasks on all of the users that are defined to an 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.

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

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 110 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 110 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 110 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" 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" 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 107, 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.")

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.

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 an 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.

Holds

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 7 shows the administrative users and the tasks assigned to the users.

Table 7. Administrator Roles in the Object Type Model

User Type	Tasks
System Administrator	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	Create and maintain users
	Create and maintain groups
Application Group/Folder/Cabinet Administrator	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 8 shows the administrative users and the tasks assigned to the users.

Table 8. Administrator Roles in the Object Owner Model

User Type	Tasks
System Administrator	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	Create and maintain users
	Create and maintain groups

Table 8. Administrator Roles in the Object Owner Model (continued)

User Type	Tasks
User with Create Application Groups, Create Folders, and Create Cabinets authority	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 8 on page 116 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 an 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.

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:
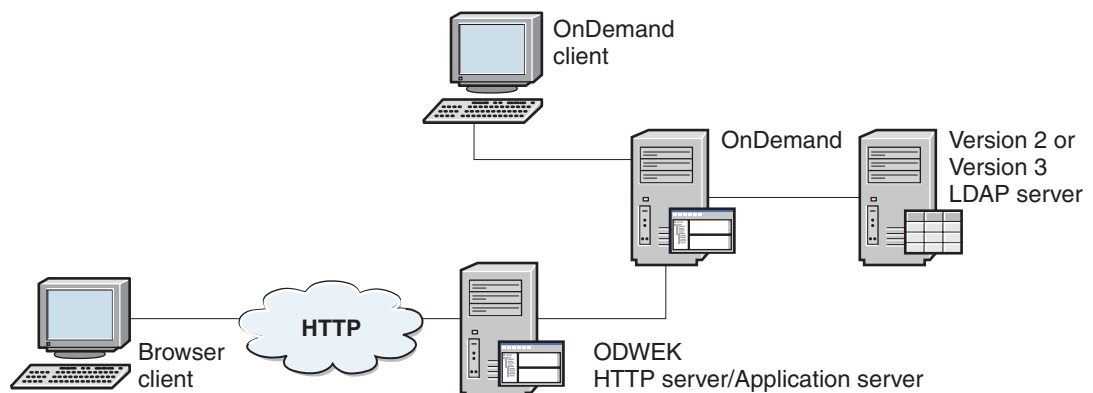
Figure 19. How OnDemand works with LDAP

When an 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.

Examples

Overview

About the examples

The examples that follow 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.
- Define a storage set that contains a primary storage node. OnDemand caches data on the object server identified by the primary storage node. The primary storage node also identifies a client node in archive storage that is managed by Tivoli Storage Manager.
- 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 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 OnDemand server. The report is indexed on the server.
- 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, 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, 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 four 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 122 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). 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 20 on page 126 shows the Add a Group dialog box with the basic information completed.
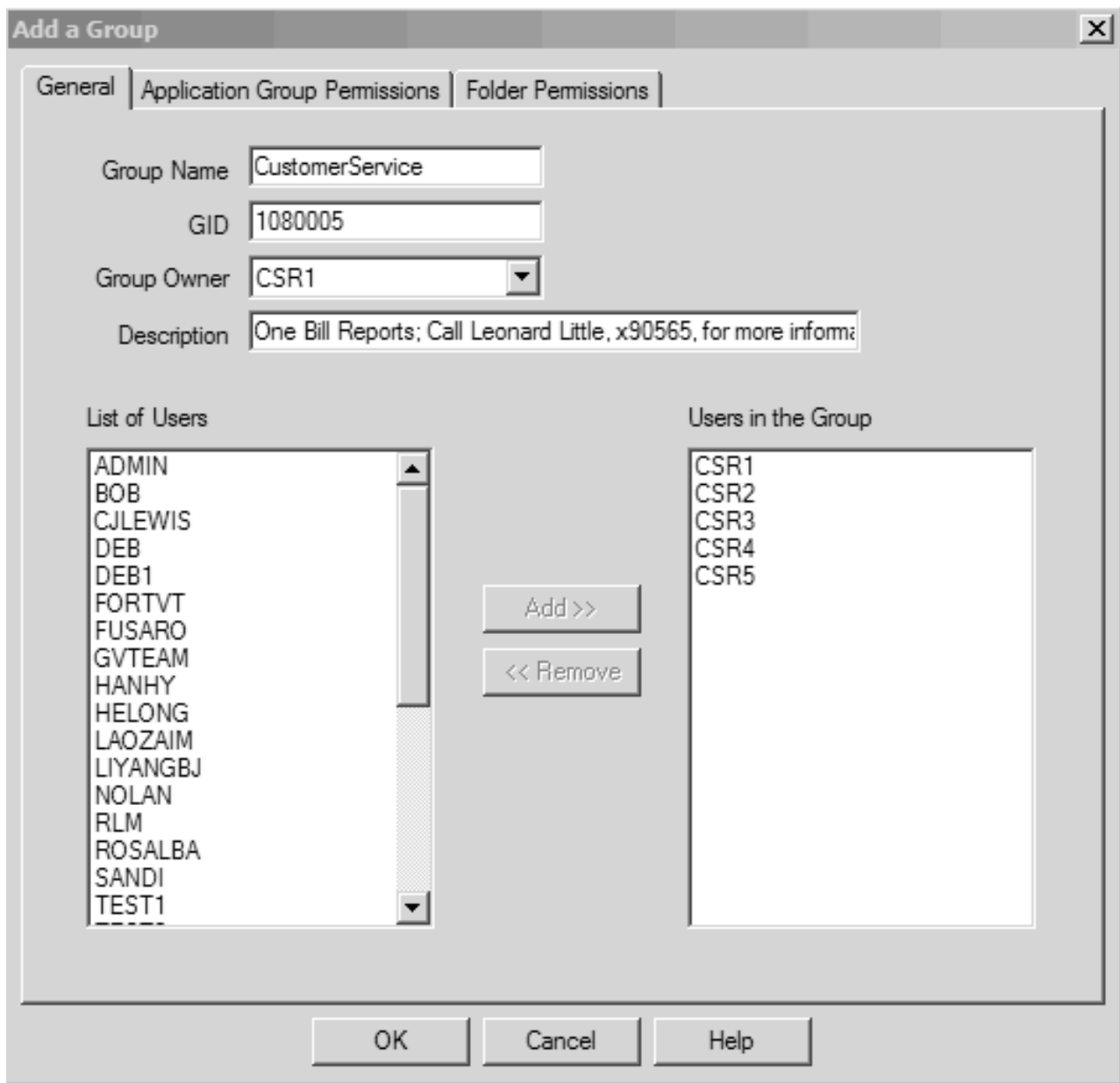
Figure 20. Add a Group dialog box

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 21 shows the Add a Group dialog box with the basic information completed.
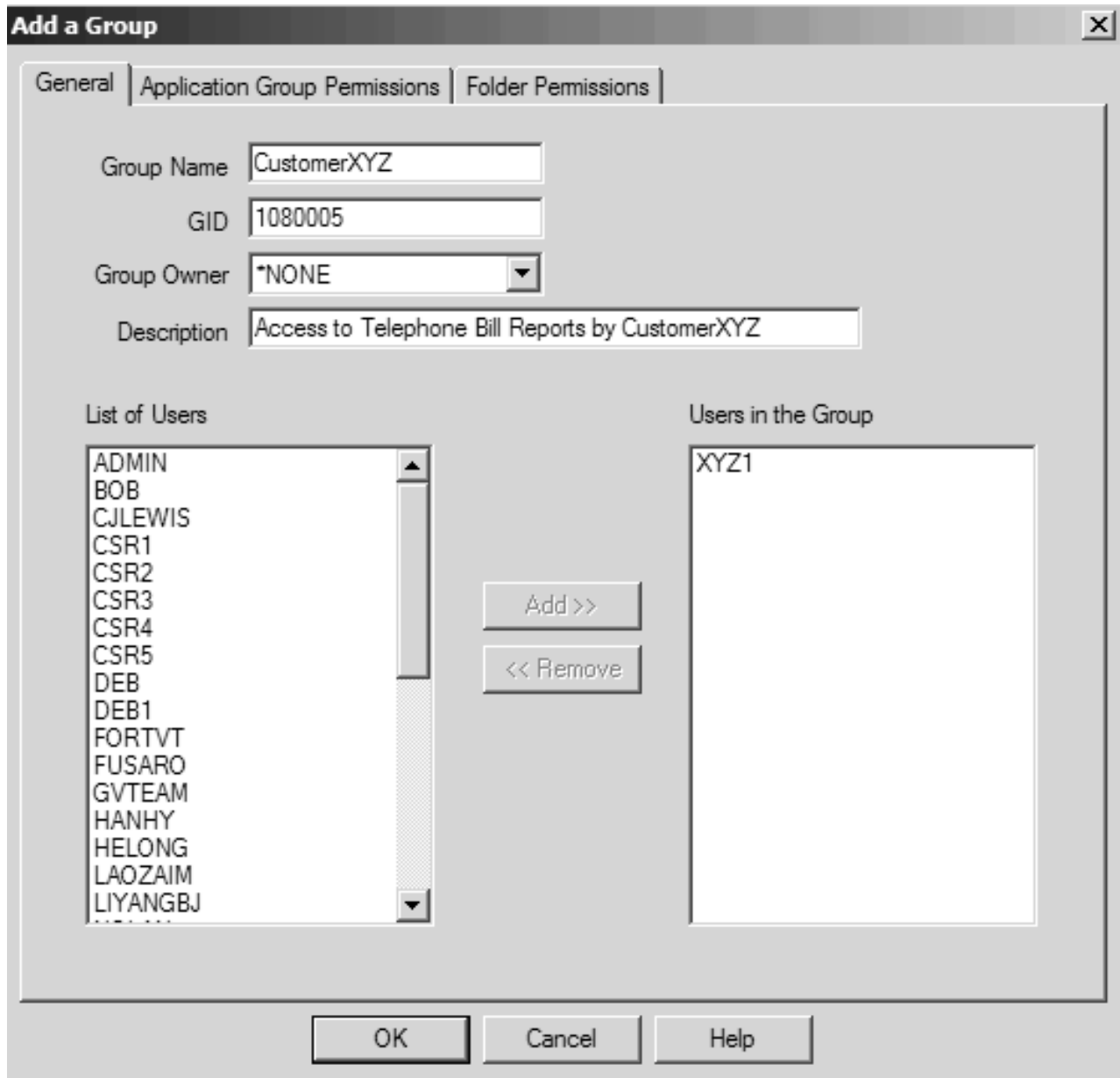
Figure 21. Add a Group dialog box

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 132. 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 22 on page 129 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 22 shows the completed Add a Printer dialog box.
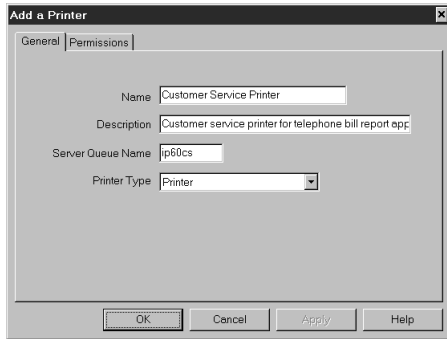
Figure 22. Add a Printer dialog box

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 132. 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 store node. However, you need to do two things to make sure that this happens. First, define cache storage file systems to Content Manager OnDemand. (The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* describes how to define cache storage file systems.) 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 set 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 workstation as the library server. Tivoli Storage Manager is used to maintain data in archive storage. The primary storage node identifies a client node in Tivoli Storage Manager.

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 pop-up 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 OnDemand).

Adding the storage set

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

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 pop-up menu, select New Storage Set to open the Add a Storage Set Dialog box.
2. In the Name field, enter the name of the storage set: *Five Year Storage Set*
3. In the Description field, enter: *Contact Leonard Little, x90059*
4. Accept the default Load Type of Fixed. This means that OnDemand will store data in the primary storage node that has the Load Data check box selected.
   Figure 23 on page 131 shows the Add a Storage Set dialog box with the basic information completed.
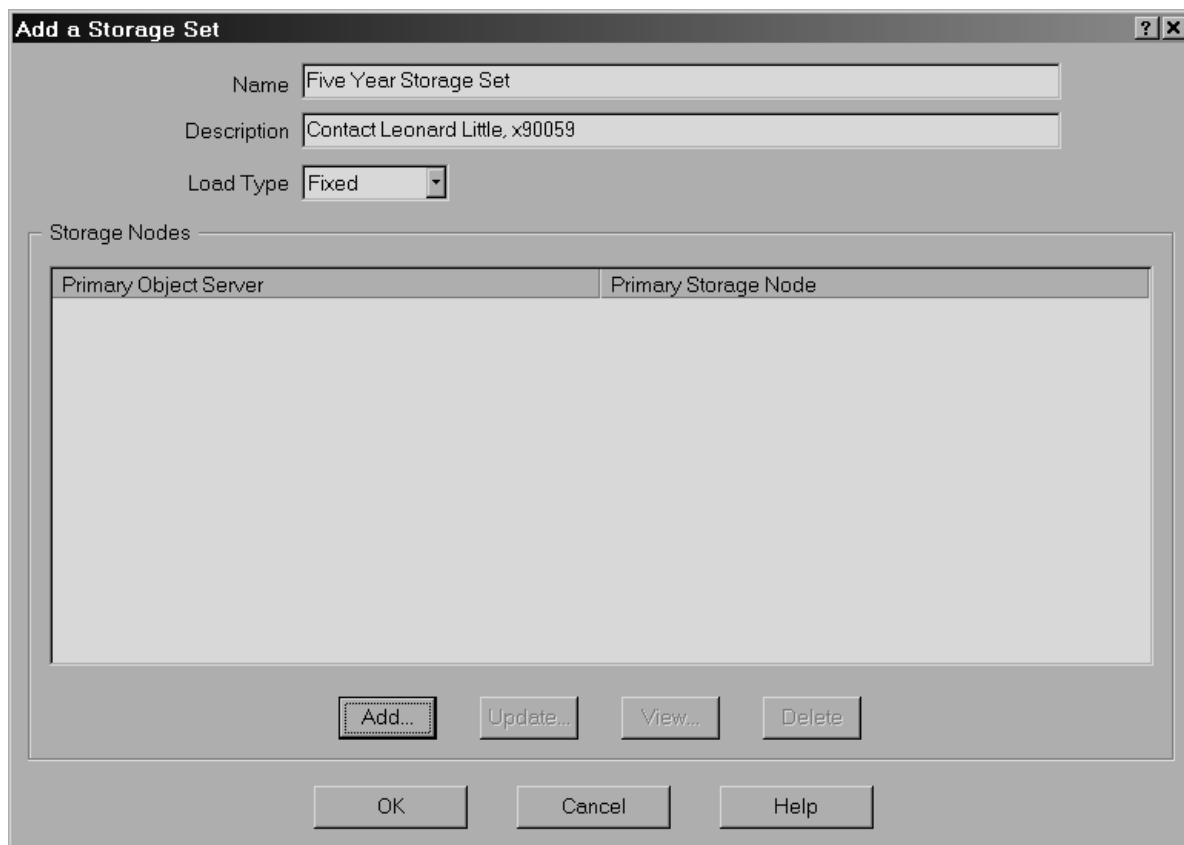
Figure 23. Add a Storage Set dialog box - basic information

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 workstation as the library server.

7. In the Storage Node field, enter: *FiveYearPSN1*. This is the name of the primary storage node. By convention, use the same name for the primary storage node and its associated client node in Tivoli Storage Manager.

8. In the Description field, enter: *Five year storage node*.

9. In the Logon field, enter: *FiveYearPSN1*. This is the name of the client node in Tivoli Storage Manager. The logon name must be the same as the client node in Tivoli Storage Manager.

10. In the Password field, enter: *FiveYearPSN1*. This is the password for the client node in Tivoli Storage Manager.

11. In the Verify Password, enter: *FiveYearPSN1*

12. Select the Load Data check box to identify this node as the primary storage node in which OnDemand will load data for the application groups that are assigned to the storage set.

13. 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.

14. Figure 24 on page 132 shows the completed Add a Storage Set dialog box.
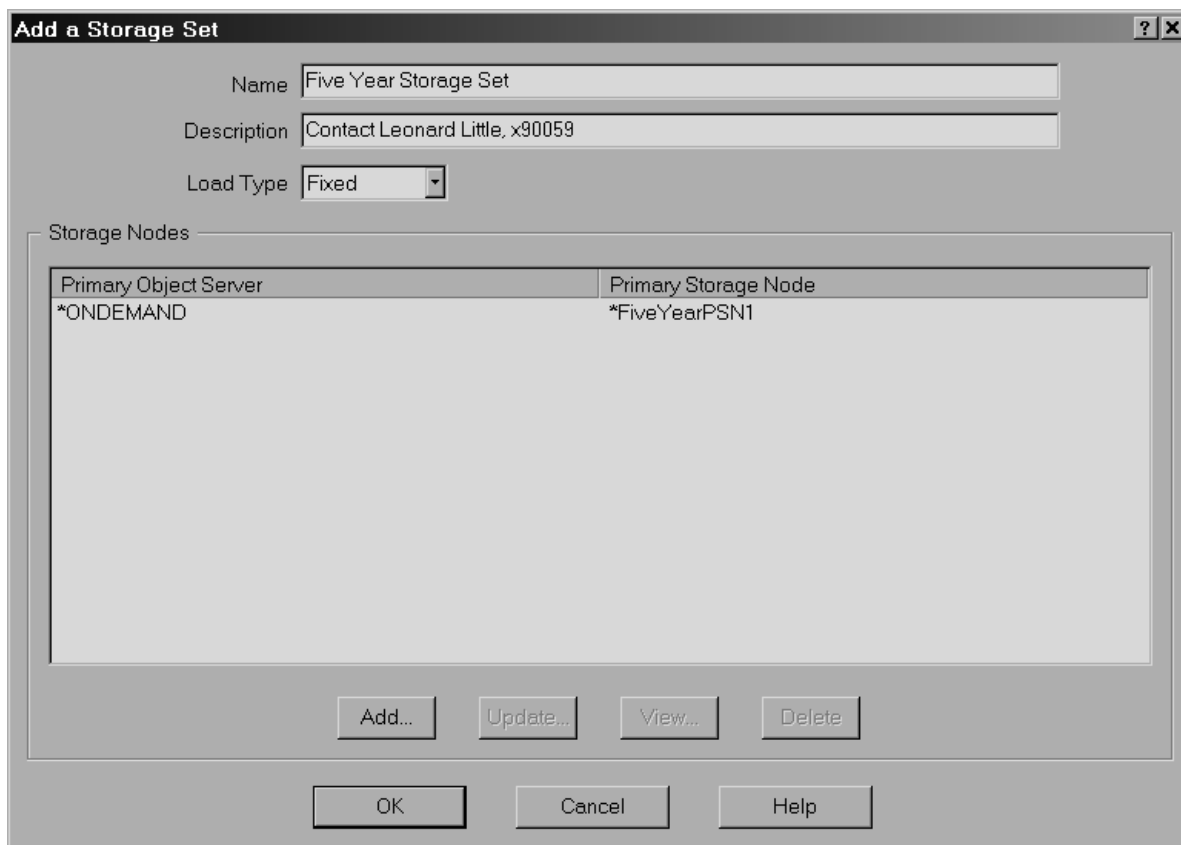
Figure 24. Add a Storage Set dialog box - completed

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. Figure 25 on page 133 shows an example of the report.
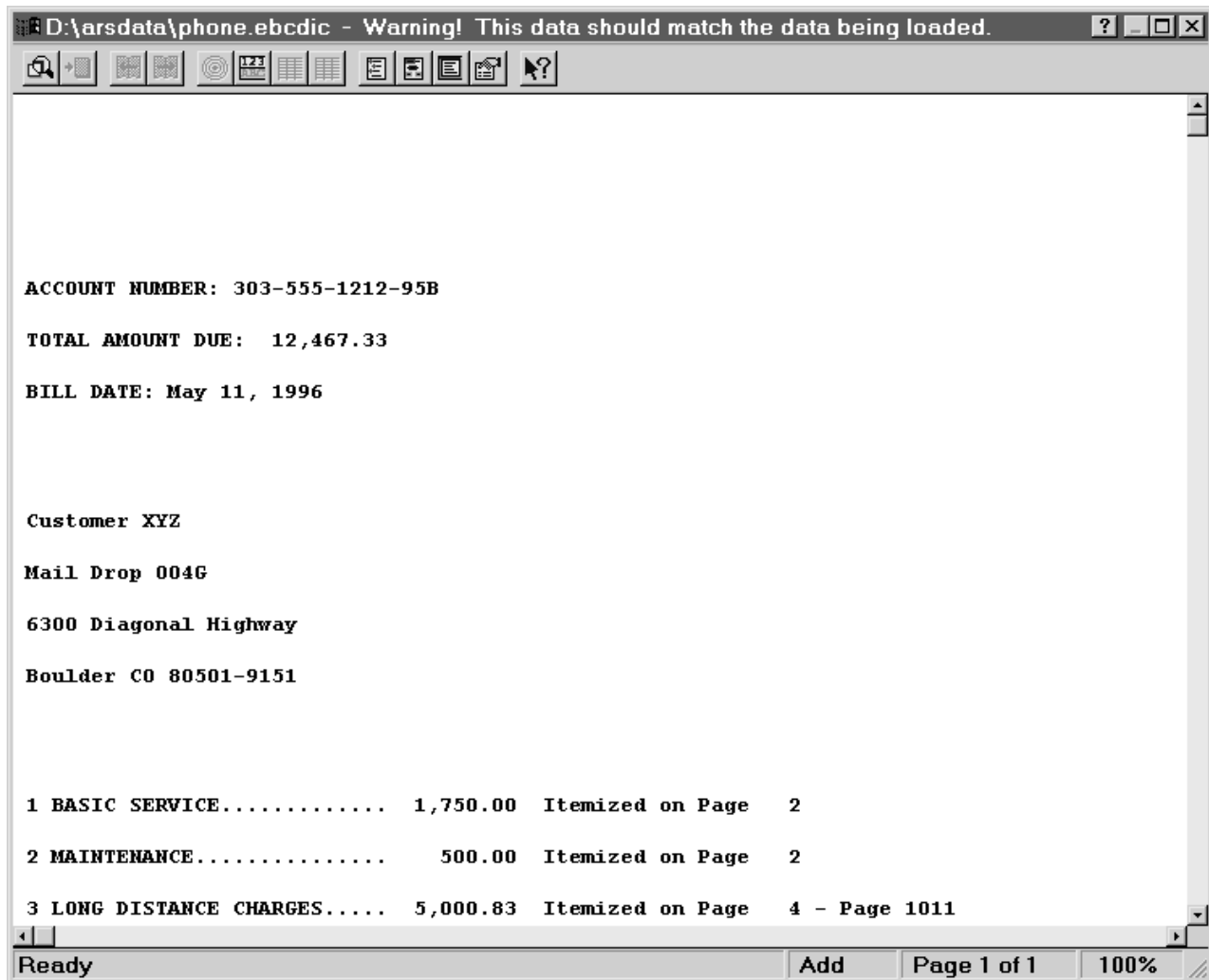
Figure 25. Report window

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.)

   Figure 26 on page 136 shows the completed General page.

| Add an Application Group | ✕ |

| Field Definition | Field Information | Advanced Index Information |
| General | Message Logging | Storage Management | Permissions |

Application Group Name

Name Telephone Bill Reports

Description Monthly telephone statements

Advanced...

| OK | Cancel | Help |

Figure 26. General page

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.
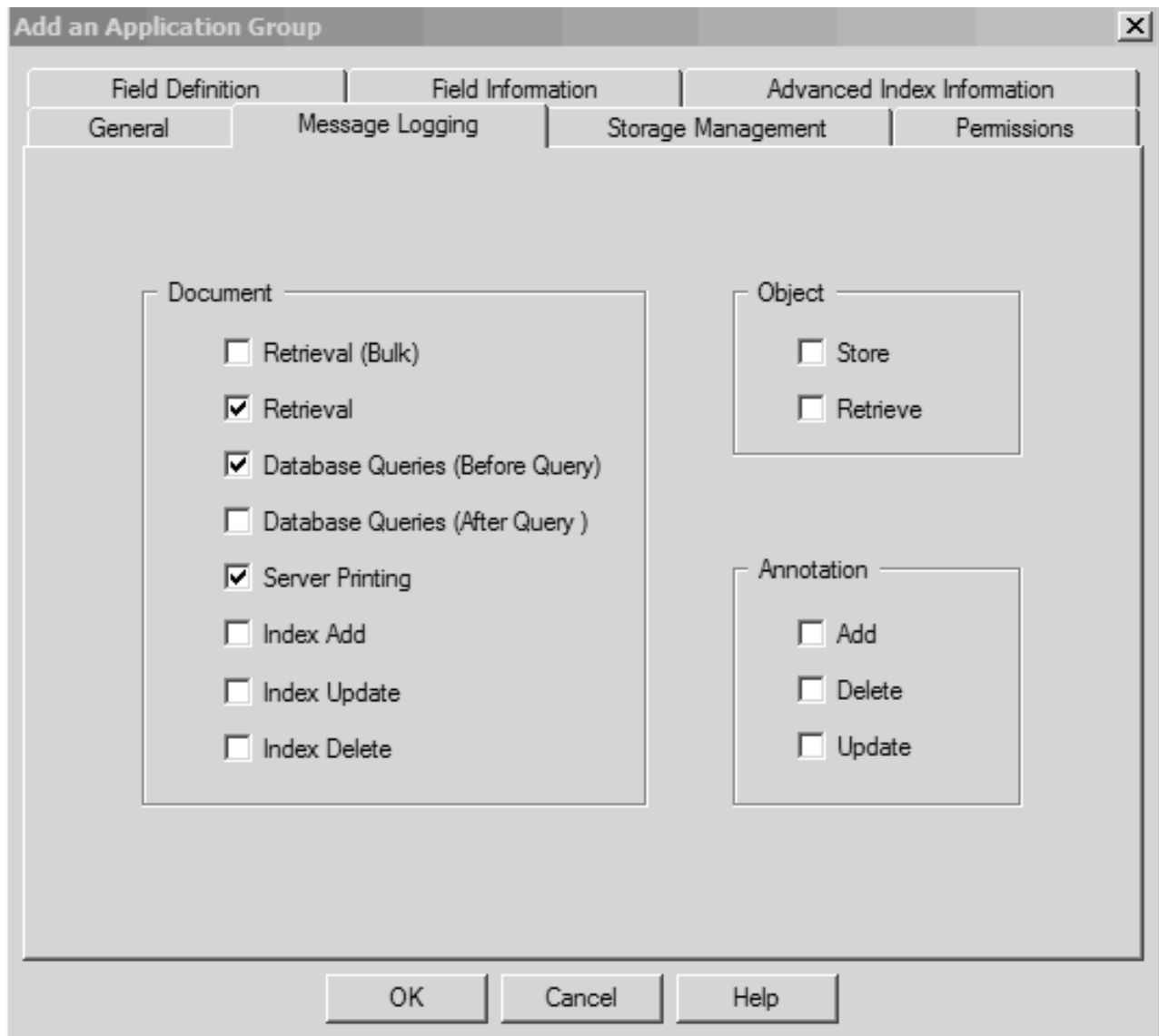
Figure 27. Message Logging page

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 125 and "Adding the Customer XYZ group" on page 126.
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.

   Figure 28 on page 139 shows the completed Field Definition page.
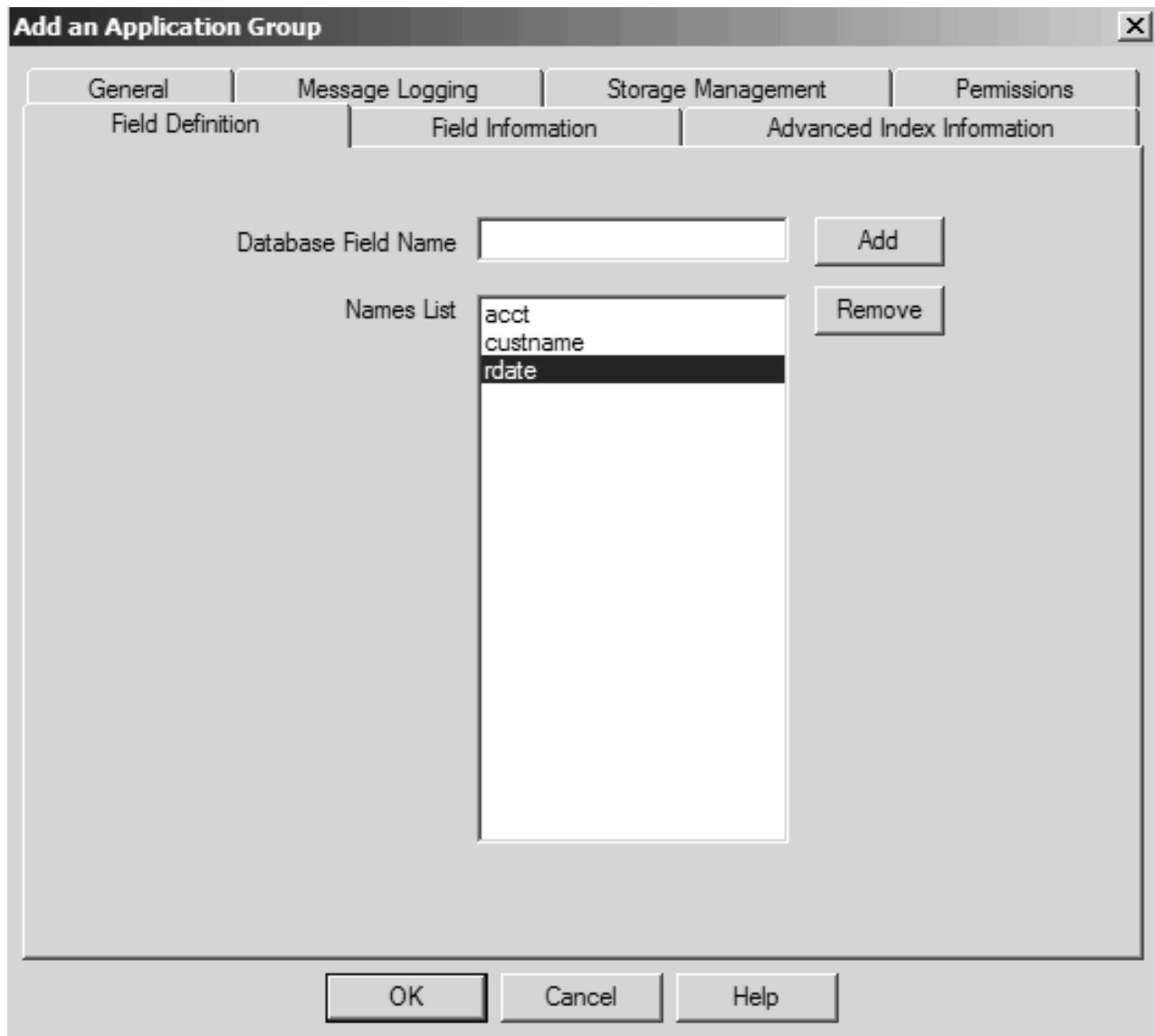
Figure 28. Field Definition page

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.

Figure 29 shows the completed General page.

Figure 29. General page

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.

Figure 30 on page 142 shows the View Information page.

Figure 30. View Information page

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. Figure 31 on page 143 shows an example of the Indexer Information page.
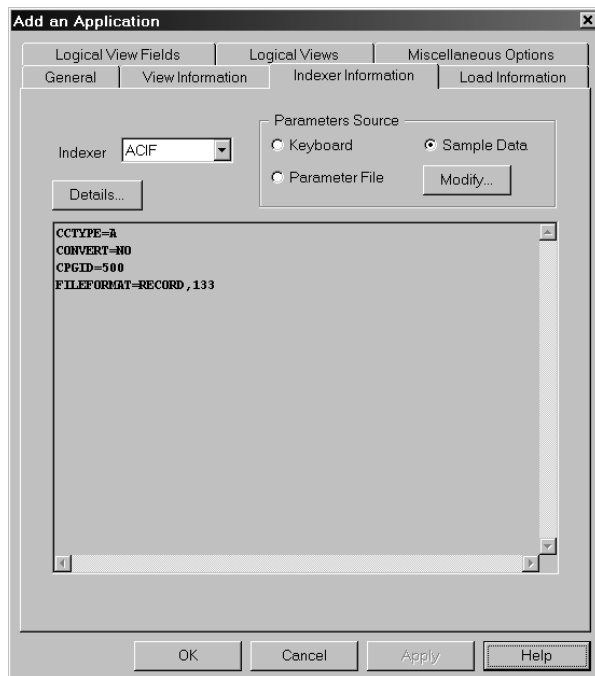
Figure 31. Indexer Information page

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. Figure 32 on page 144 shows the report window.
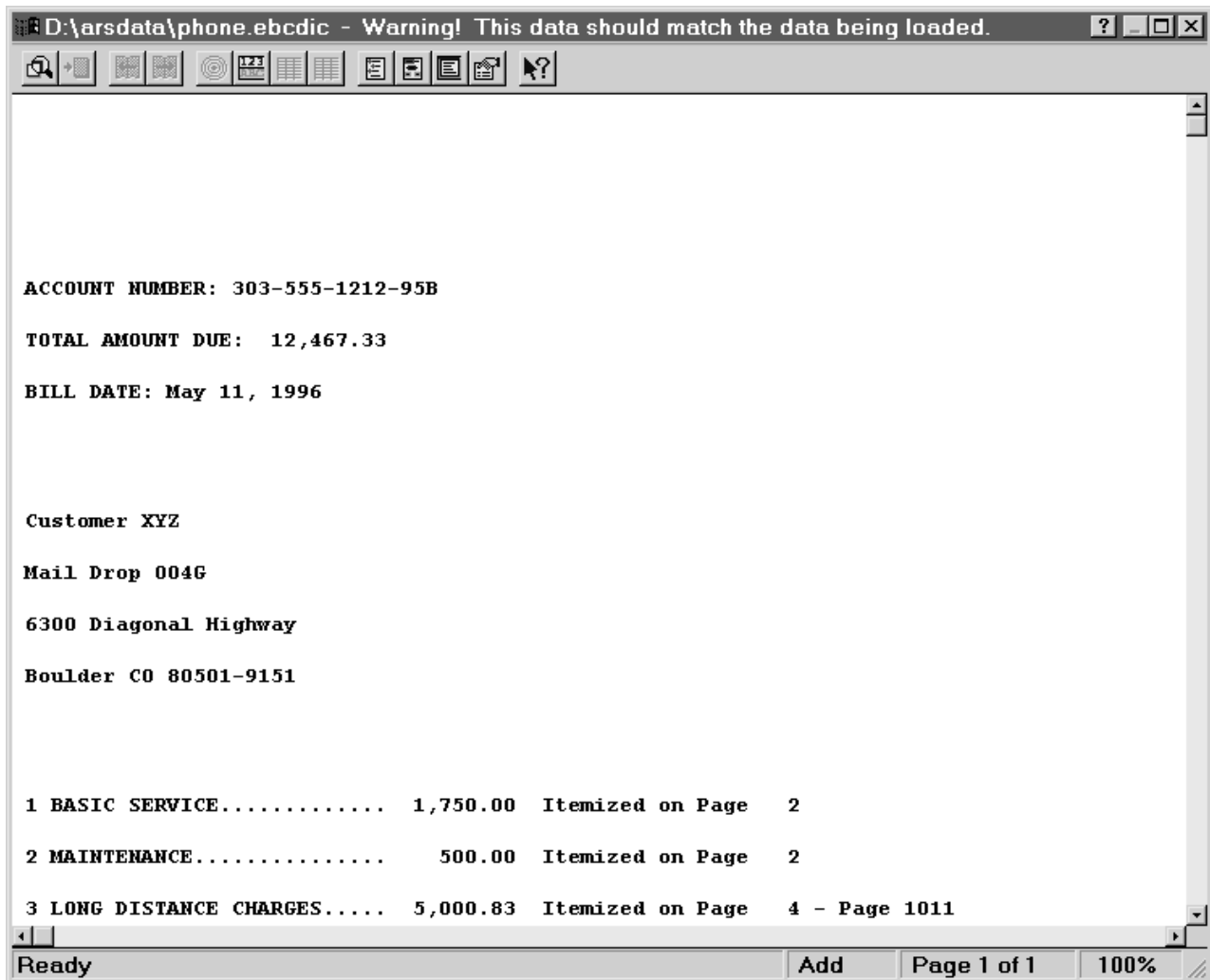
Figure 32. Report window

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.

        The following shows the completed Add a Trigger dialog box for Trigger1.
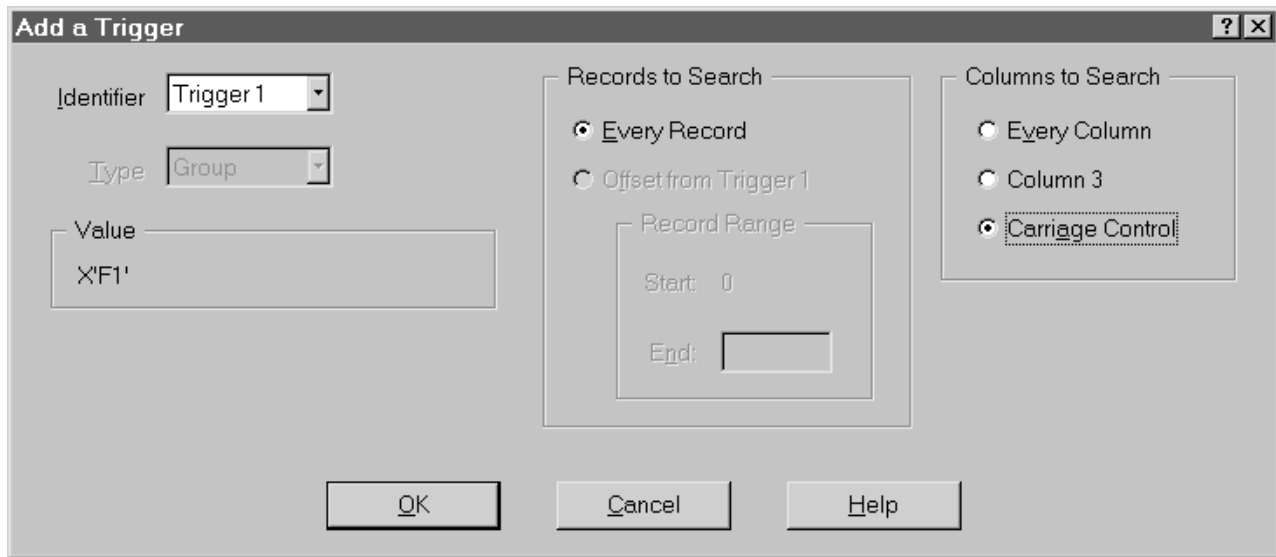
Figure 33. Add a Trigger dialog box

    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.

       Figure 34 on page 146 shows the completed Add a Field dialog box for Field1.
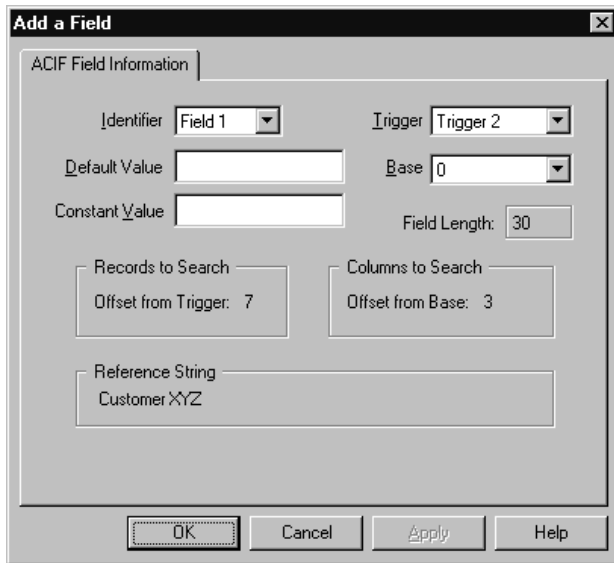
Figure 34. Add a Field dialog box

    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.

       Figure 35 on page 147 shows the Add an Index dialog box for Index1.
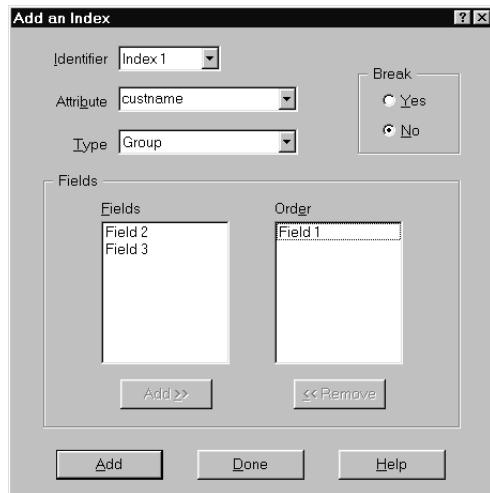
Figure 35. Add an Index dialog box

   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.

   Figure 36 on page 148 shows the completed Resource Information page.
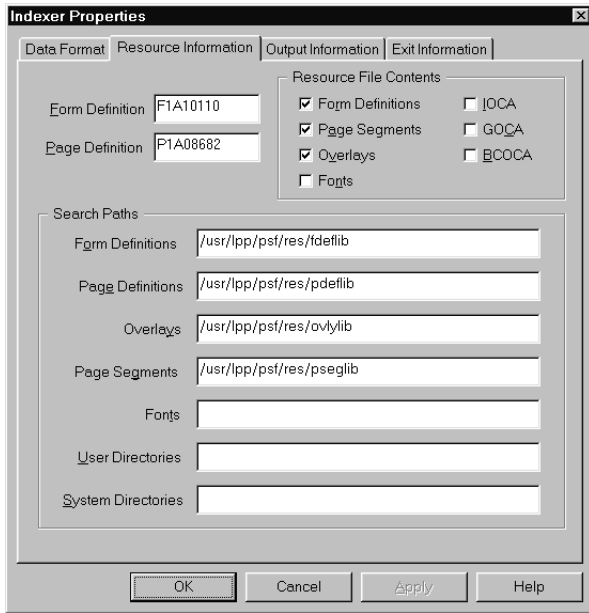
Figure 36. Resource Information page

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 DB Name list, select rdate to specify the format of the date as it appears in the report.
3. From the format list, select %Y

Figure 37 on page 149 shows the Load Information for the report date field.

Figure 37. Load Information page

4. In the Application Group DB 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.

   Figure 38 shows the Load Information for the account number field.

Figure 38. Load Information page

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 128.

Figure 39 shows the completed Miscellaneous Options page.
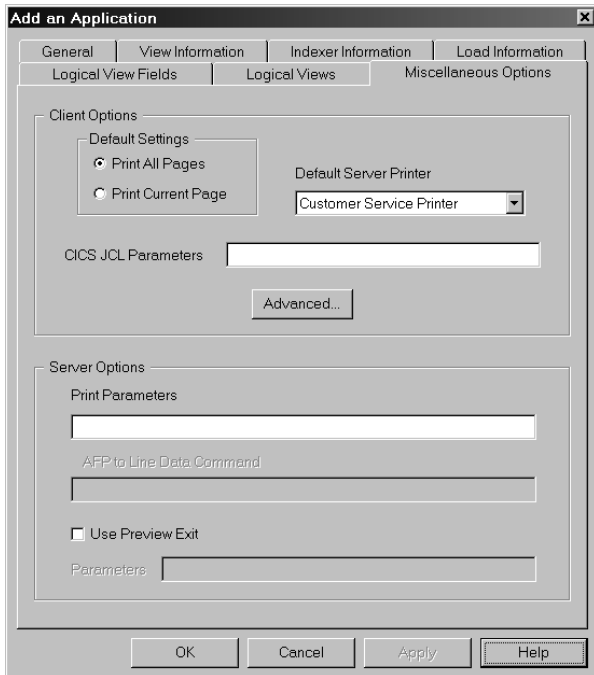
Figure 39. Miscellaneous Options page

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 125 and "Adding the Customer XYZ group" on page 126.
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.

3. Figure 40 shows the completed Field Definition page.
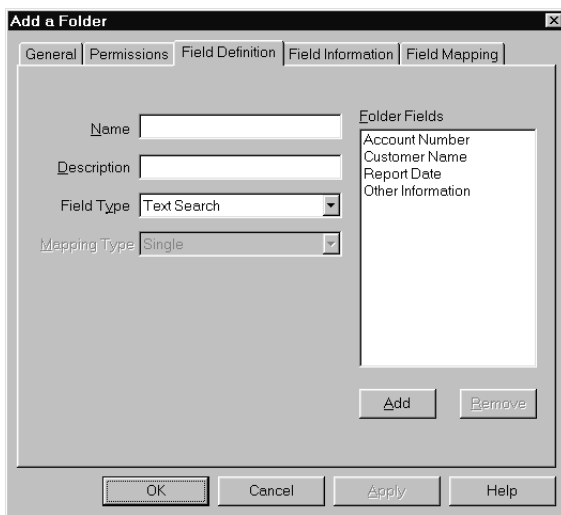
Figure 40. Field Definition page

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 Fmt list, select %Y

   e. From the Defaults Fmt list, select %Y

f. In the Interval area, select Last, type a 3 (three) in the entry field, and select Months

Figure 41 shows the Field Information page for the Report Date field.

Figure 41. Field Information page

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

   Figure 42 on page 154 shows the Field Information page for the Customer Name field.
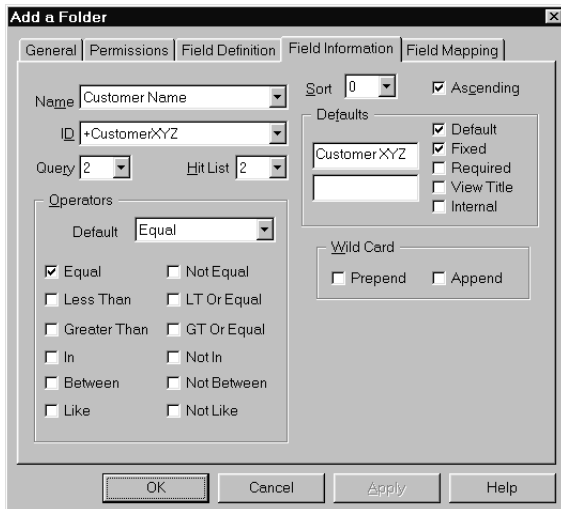
Figure 42. Field Information page

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 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.

Web administrator

You can use a Web administrator to administer OnDemand. This client provides advantages to both OnDemand administrators and individuals who are not full-time administrators but might perform certain administrative tasks:

- **For administrators:** The OnDemand Web administrator allows you to add, view, update, and delete users, groups, applications, application groups, folders, printers, storage sets, and cabinets. You can create custom administrative forms by using the Web administrator and IBM Workplace Forms Designer. Individuals who are not full-time administrators can use the custom forms to perform administrative tasks.
- **For individuals who are not full-time administrators:** The Web administrator enables users who do not have indepth knowledge of Content Manager OnDemand to complete administrative tasks.

This section describes two business scenarios of using the Web administrator, the requirements for the client, and how to install the client.

Scenario 1: Managing user access and permission levels

In this scenario, the manager of the customer service department needs to monitor the service representatives and make changes to the users and user permissions in Content Manager OnDemand. The manager's tasks include:

- A new customer service representative joined the department, and the manager needs to add that person as `CSR6` to the Content Manager OnDemand system and to the `CustomerService` user group, and grant this user the same permissions as those of all other customer service reps.
- A customer service rep, `CSR1`, left the company. The manager needs to remove her user ID from the `CustomerService` user group and from the system.
- Before she left, `CSR1` was the team lead of the service representatives, and the user administrator in Content Manager OnDemand. After she left, the manager decided to name `CSR2` as the user administrator, and needs to make this change in the system.

By using the Web administrator, the manager can use a browser to access the users and user groups in a form format, and make appropriate updates.

Scenario 2: Managing user access

In this scenario, all customer access to phone bill reports is controlled by Director of Customer Service. At this time, a group of users from XYZ can access XYZ's phone bill reports through ODWEK.

There is a new customer BCD. The director wants to add them as a new customer group and let them access their phone bill reports through ODWEK. By using the Web administrator, the director can add a new user group and grant this group access to the appropriate data.

You can certainly use the installed &admc; to complete all above tasks, however, if you use the Web administrator and IBM Workplace Forms Designer to simplify the forms, you can enable people who are not Content Manager OnDemand administrators to complete administrative tasks.

Requirements for the Web administrator

For the Web administrator requirements, see http://www.ibm.com/support/docview.wss?rs=129&uid=swg27016455 or search for 7016455 at http://www.ibm.com

Installing the Web administrator

For Web administrator installation procedures, see: http://www.ibm.com/support/docview.wss?rs=129&uid=swg21257664 or search for 1257664 at http://www.ibm.com

Loading AFP and line data

About this task

This section provides an overview of the data loading process.

The process begins with configuring storage on the system and configuring the programs that transmit, index, and load data. You need to decide where to index your reports and define the application groups and applications for the reports. The next step is to load the data – the process of adding the index data to the database and loading the report data and resources on storage volumes. When a load process completes, you can view the messages that were saved in the system log. To complete the data loading process, backup the Content Manager OnDemand and archive storage manager databases.

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 OnDemand, the print output of an application program is called a report. Content Manager OnDemand provides programs that can automatically receive reports from z/OS systems, index the reports, 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 an OnDemand server. If you generate your reports on an z/OS system, then you would typically transfer the reports to the server and use the data indexing and loading programs to process them. 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 transfer the resources to the server or you can mount a resource library using the Network File System (NFS). You can also index your reports on the z/OS system and then transmit the report, index, and resource files to the server to load them.

You can create up to 32 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. If you use Tivoli Storage Manager to maintain report data in archive storage, you can configure the system to maintain a backup copy of the reports that are stored in 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
About this task

When you initially configure an OnDemand system, you calculate the total amount of disk, optical, and tape storage required to hold the OnDemand database, database log files, reports, and the Tivoli Storage Manager database and recovery log. 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 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 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 that you need, you should organize the physical storage devices for the various components of the system, such as the database, Download, and cache storage. If you plan to maintain data in archive storage, then you also need to prepare storage devices for use by Tivoli Storage Manager. Physically separating program directories, the database, cache storage, and temporary work spaces can improve performance.

The *IBM Content Manager OnDemand for Multiplatforms: Introduction and Planning Guide* provides information, formulas, and worksheets that can help you calculate your storage requirements. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* contains recommendations for organizing storage devices, describes how to define the database, cache storage, and temporary storage areas, and describes how to configure storage libraries and devices and define them to Tivoli Storage Manager.

Defining the application group
About this task

When you archive reports in Content Manager OnDemand, 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 OnDemand maintains the report data and how long it maintains the index data and the report data. You can use the 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, then the application group must use a storage set that includes a primary node that identifies a client node in Tivoli Storage Manager. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides information about configuring Tivoli Storage Manager.

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 OnDemand administrative client to define an application. See the online help for the administrative client for more information about defining applications.

Accessing reports, index data, and resources
About this task

Most customers process reports that were created by programs that run on z/OS systems. You can use several methods to transfer report data from the z/OS system to the Content Manager OnDemand server. For example:

- You can use Download for z/OS to transmit a print data set from the JES Spool to file systems on the server. The z/OS component of Download operates as one or more JES writers. You configure the writers to interpret JCL parameters, such as CLASS and DEST, and route spool files to an Content Manager OnDemand server. You can specify other JCL parameters, such as FORM and DATASET, and invoke a user-written program to process additional job information that is sent with the data. The ARSJESD program starts the server component of Download and monitors a specific TCP/IP port for data from a z/OS system. You can start one or more occurrences of the ARSJESD program, each monitoring a different port, receiving data sets from different Download writers, and storing data on different file systems. You can configure Download to automatically transmit all of the reports that you need to archive on the system. See *PSF for z/OS: Download for z/OS* for information about installing, configuring, and operating Download on the z/OS system. *PSF for z/OS: Download for z/OS* lists the

supported JCL parameters and provides information about the user exit point at which you can invoke a user-written program. See "Download exits and processing" on page 193 for more information about using the ARSJESD program to invoke a user-written program to process a downloaded file, the JCL parameters, and the additional job information. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* shows how to configure the server to start the ARSJESD program whenever the system is initialized.

- You can provide network access to files that reside on a z/OS system. For example, if you plan to index reports on the server, you can use NFS to mount the resource library on the server so that the ARSLOAD program can access the resources required by a report. However, for performance reasons, IBM recommends against using NFS to access the data sets that contain the actual report or index data.

- You can save a copy of your reports on tape storage volumes. For example, if a system that generates a report is not connected to the same network as the OnDemand server, then you may need to save a copy of the report on a tape storage volume and have it delivered to a location where it can be processed.

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 either run the ARSLOAD program from the command line or configure it to run as a daemon (UNIX servers) or service (Windows servers) to periodically check for input data to process.

Note: The OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) must be running, otherwise the ARSLOAD program will fail.

See "ARSLOAD" on page 279 for more information about the ARSLOAD program, including the parameters that you can specify to process your reports.

Specifying input data to process
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:

```
MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD
```

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

Unless you specify otherwise, the ARSLOAD program uses the FORM 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 (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 (MVS, JOBNAME, DATASET, or FORMS) that identifies the application.

Specifying 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
- For UNIX servers, you can specify the **-U** parameter each time that you run the ARSLOAD program and name a file that contains the userid and password for a user with administrator authority for the application group
- For UNIX servers, you can specify a system administrator userid and password in the ARSLOAD.CFG configuration file. The ARSLOAD program will use the values in the ARSLOAD.CFG file, unless you specify otherwise (with the **-u** and **-p** parameters or if you specify a different configuration file with the **-U** parameter).
- For Windows servers, the Load Data service uses the userid and password of the instance owner. If you run the ARSLOAD program from the command line, then you should use Windows unified logon.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* describes how to configure the ARSLOAD.CFG file.

Note: After you configure the userid and password in the ARSLOAD.CFG file, remember to change the password any time that you change the user's password in OnDemand. Otherwise the load will fail. The ARSLOAD program can accept an expired password. However, the ARSLOAD program will fail if an incorrect password is specified.

Running ARSLOAD on UNIX servers
About this task

You can use the INIT facility to start the ARSLOAD program as a daemon. The INIT facility makes sure that the ARSLOAD program starts each time that the system is restarted. Each record that you add to the INIT facility defines a process to run and its parameters.

The following shows an example of an INIT record for an AIX server to automatically start the ARSLOAD program. The ARSLOAD program checks the specified directories for input data to process and stores temporary data in the file system specified with the **-c** parameter. The ARSLOAD program will process any input file that has a file type of ARD or PDF.

Important: OnDemand does not support loading files that have spaces in the file names. The space character is used as a delimiter for the command line of arsload and the functions it calls.

Unless otherwise specified, the ARSLOAD program uses the FORM part of the input file name to determine the application group to load. By default, an application group must contain one and only one application. The ARSLOAD

program checks for input data to process every thirty minutes. By default, the ARSLOAD program logs on to the server and accesses an application group with the userid and password from the ARSLOAD.CFG file.

```
ars6:2:once:/usr/lpp/ars/bin/arsload -d /arsacif/acif2
-d /arsacif/acif3 -d /arsacif/acif4 -c /arsacif/acif1 -t 1800 -v
```

See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about configuring the system to automatically load data. See your operating system documentation for information about the INIT facility, automating tasks, and operating system initialization.

Running ARSLOAD on Windows servers
About this task

You should configure the ARSLOAD service to start automatically when the system is started. To configure ARSLOAD service:

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

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

Indexing input data

About this task

You must generate index data for a report before you load the report into the system. OnDemand supports several programs to help you 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), 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.
- IBM Content Manager OnDemand PDF Indexer for Multiplatforms, to index Adobe PDF files
- The Xenos d2e Platform, an optional and separately-priced set of programs to index and convert AFP, Metacode/DJDE, and PCL print files.
- 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 *IBM Content Manager OnDemand for Multiplatforms: Indexing Reference*.

The information in the following sections shows examples of using ACIF to index reports. You can use ACIF to index reports and collect resources into a resource group. ACIF processes a report and produces an output file with indexed groups of pages to load on storage volumes, an index file for the database manager, and a resource group file that contains the resources required to view and reprint pages of the report. ACIF can run on an OnDemand server or a z/OS system. Before you run ACIF on an OnDemand server, you must transfer the report to the server and provide access to the resources that may be required to process the report. After you run ACIF on a z/OS system, you must transfer the output file, index file, and resources created by ACIF to the OnDemand server before you can run the load process.

Indexing reports on OnDemand servers
About this task

If you plan to index your reports on the server, then you must first transfer the report to the server. You must also provide access to the resources required by the report. Figure 43 on page 164 depicts one method that you can use to index reports on the server.
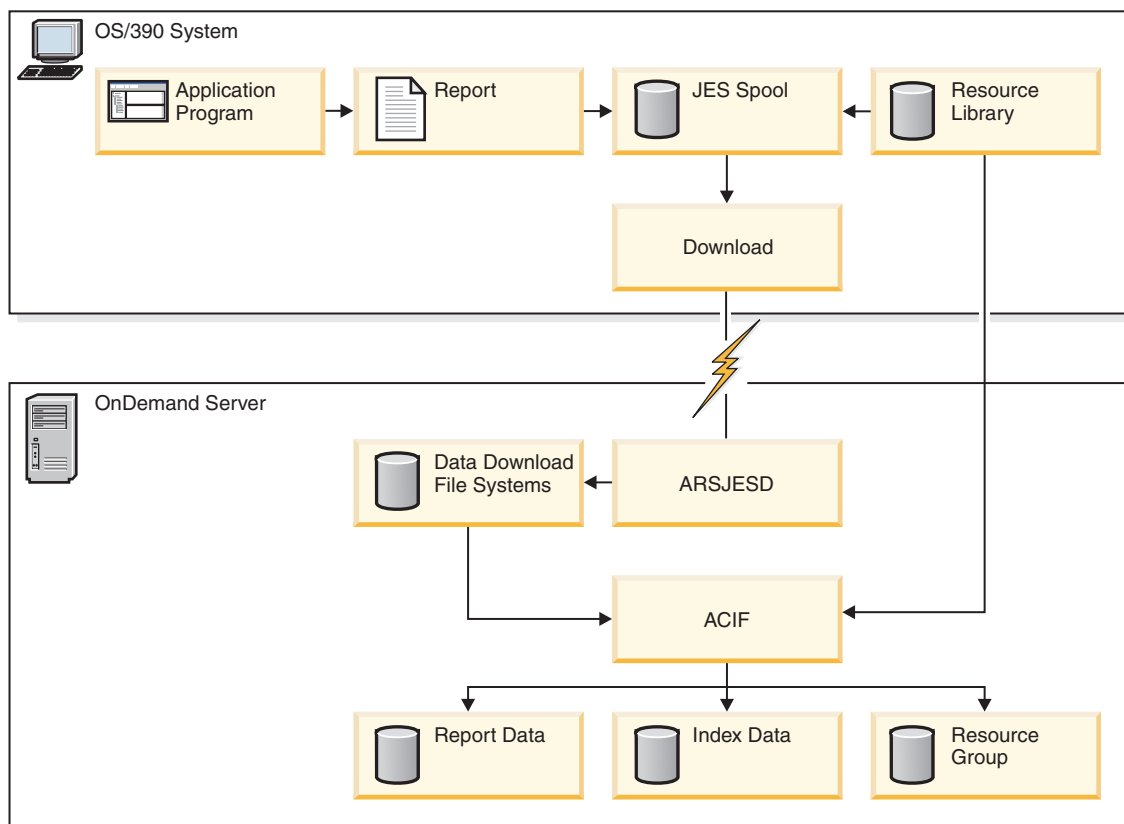
Figure 43. Indexing reports on OnDemand servers

In the example, Download is used to automatically transmit the report to the OnDemand server. After the report has been successfully transmitted to the server, the ARSLOAD program can process the data. You can either invoke the ARSLOAD program from the command line to process a specific file or use facilities of the server operating system to run the ARSLOAD program and automatically check for input data to process. Since the report must be indexed before it can be loaded into the system, the ARSLOAD program automatically calls the data indexing program. In the example, the ARSLOAD program calls ACIF to index the report. In the example, ACIF uses NFS to access the library that contains the resources.

The *IBM Content Manager OnDemand for Multiplatforms: Indexing Reference* contains details about indexing reports and shows examples of indexing several types of input data.

Indexing reports on z/OS systems
About this task

If you plan to index your reports on z/OS systems, then after you index a report, you must transfer the output file, index file, and resource group file generated by ACIF to the OnDemand server where you plan to load the data. Figure 44 on page 165 depicts one method that you can use to index reports on z/OS systems.
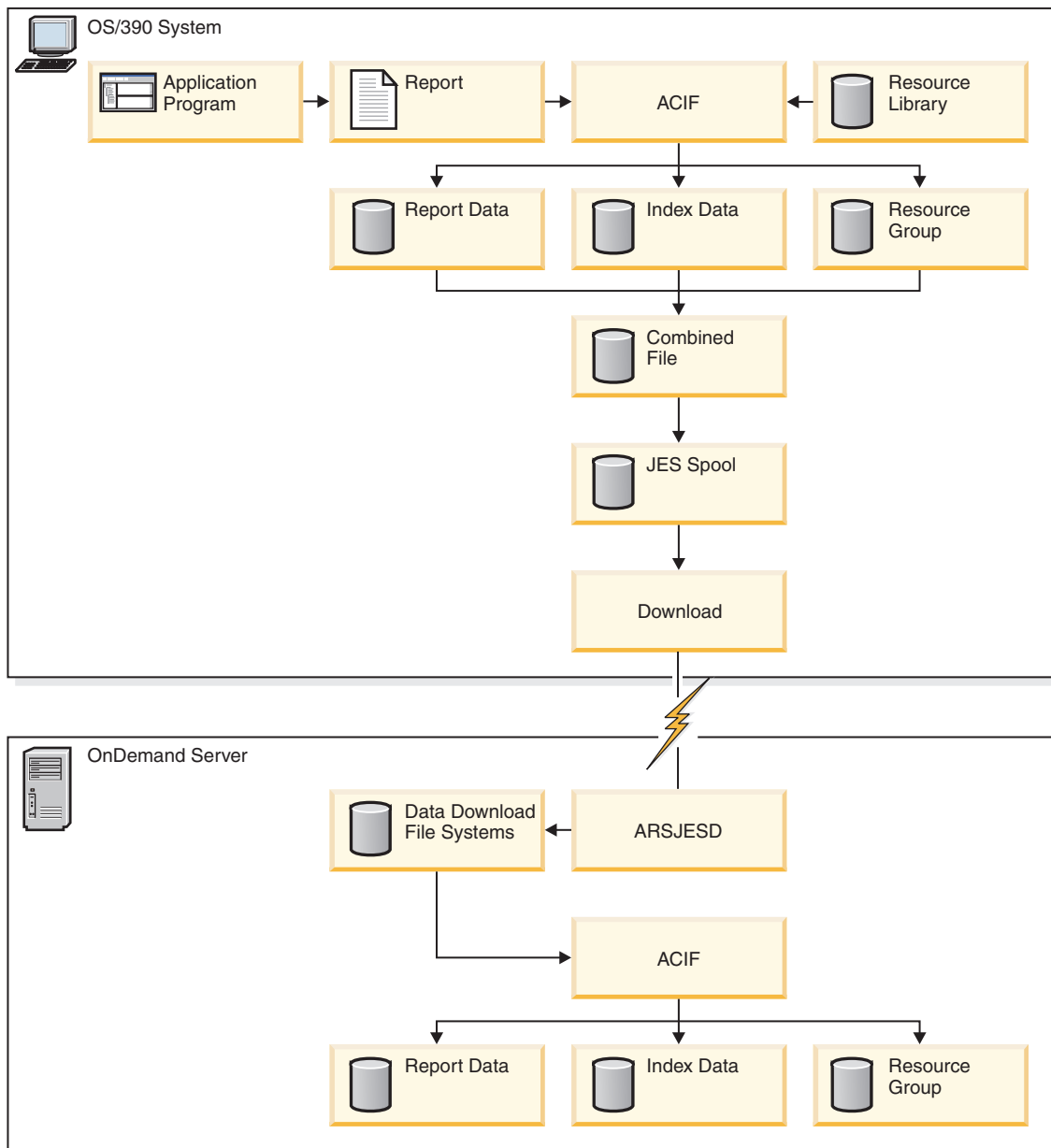
Figure 44. Indexing Reports on z/OS systems

In the example, ACIF processes a line data report, generates index data, converts the input to AFP data, and collects resources from a library on the system. After running ACIF, you must combine the three output files generated by ACIF into one file and then store the combined file in the JES Spool. Download can be used to automatically transmit the combined file to the OnDemand server. After the combined file has been successfully transmitted to the server, you must run ACIF on the OnDemand server to process the combined file. Using parameters to convert the data and name a default form definition (it is not necessary to specify any trigger, field, or index parameters when you run ACIF on the server), ACIF splits the combined file into separate index, resource, and output files that can be processed by the ARSLOAD program. Contact the IBM support center if you have questions about indexing data on z/OS systems and processing the output on an OnDemand server. "Download exits and processing" on page 193 provides more information about processing files transmitted by Download.

The *IBM Content Manager OnDemand for Multiplatforms: Indexing Reference* contains details about indexing reports and shows examples of indexing several types of input data with ACIF.

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. For example, the ARSLOAD program may need to drop duplicate index records. 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

The ARSLOAD program calls the storage manager to copy storage objects to cache storage and archive storage. The storage manager extracts information from the application group to determine where and when to copy the storage objects.

OnDemand uses an object called a storage set to determine the locations that can hold report data. A storage set contains one or more storage nodes. A storage node identifies an object server. A storage node can specify cache storage, archive storage, or both. A storage set can write data to one and only one storage node at a time (the *active* storage node).

Cache storage

The primary purpose of cache storage is for short-term, high-speed retrieval of report data. Cache storage can be located on one or more object servers, depending on how you configure your system and define your storage nodes. 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 OnDemand copies documents to cache storage.

Archive storage

A storage node can identify a client node in storage that is managed by Tivoli Storage Manager. OnDemand works with Tivoli Storage Manager to maintain storage objects in archive storage 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.

Copy storage pool

You can define a copy storage pool in Tivoli Storage Manager to maintain a backup copy of files that are stored in archive storage. Tivoli Storage Manager maintains the backup copy independently of OnDemand. If the primary copy of a file is unavailable or becomes corrupted, then Tivoli Storage Manager can automatically retrieve the backup copy of the file. See your Tivoli Storage Manager information for help with configuring a copy storage pool.

If you define a copy storage pool on your system, you must copy files from the primary storage pool to the copy storage pool. Tivoli Storage Manager provides a scheduling component that you can use to automate the copy process. IBM recommends that you schedule the copy process to take place soon after you load reports into the system.

Resources

If caching of data is enabled, OnDemand stores resources in the cache to provide fast retrieval when a user selects an item for viewing. The ARSLOAD program saves only one copy of a resource on the system, even if several reports use the same resource. 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 storage that is managed by Tivoli Storage Manager, 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. Table 9 lists the fields in a Load ID.

Table 9. Content Manager OnDemand Load ID

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

Backing up databases

About this task

After you load reports into the system, IBM recommends that you create a backup copy of the Content Manager OnDemand database and the Tivoli Storage Manager 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.
- When you backup the Tivoli Storage Manager database, you protect control information that Tivoli Storage Manager uses to maintain report data in archive storage.

IBM recommends that you backup the databases at least once a week, and more often, if you load reports every day. If you are using DB2, then you can use the ARSDB program to backup the database. If you are using Oracle, then you should use the Oracle procedures to backup the database. If you are using SQL Server, then IBM recommends that you use the Database Maintenance Plan Wizard to configure and schedule your database backups. For Tivoli Storage Manager, you can run or schedule the Tivoli Storage Manager database backup using the Tivoli Storage Manager administrative client.

Backup the Content Manager OnDemand database
About this task

For DB2, Content Manager OnDemand provides the ARSDB program so that you can create a backup image of the Content Manager OnDemand database. When you use the ARSDB program to backup the database and you maintain the archived log files on disk, Content Manager OnDemand automatically removes the log files from the archived log file directory when the backup completes. If you use Tivoli Storage Manager to maintain the DB2 archived log files, the Tivoli Storage Manager policy domain determines when the archived log files are removed from Tivoli Storage Manager storage.

The ARSDB program supports two types of database backups, an online backup and an offline backup:

- An online backup can be taken when other applications or processes are connected to the database. That is, other applications and processes can continue to read or modify data while the backup is in progress. You can use facilities of the server operating system to schedule an online backup.
- During an offline backup, only the backup task is connected to the database. Before starting an offline backup, IBM recommends that you stop the Content Manager OnDemand server programs and other related processes.

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

See "ARSDB" on page 235 for information about how to use the ARSDB program to create a backup image of the Content Manager OnDemand database.

Backup the Tivoli Storage Manager database

About this task

After you load reports into the system, IBM recommends that you backup the Tivoli Storage Manager database. A backup image can be used to recover the database, in the event that a hardware failure occurs or the database becomes corrupted.

The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* describes how to define a backup device to Tivoli Storage Manager and create a full backup of the Tivoli Storage Manager database. See the *Tivoli Storage Manager Administrator's Guide* for details about protecting the Tivoli Storage Manager database.

Protecting cache storage

About this task

Cache storage is the primary, short-term storage location for reports. If you do not configure the system to copy reports to archive storage when you load them into the system, then you need to consider how you can recover the reports in the event that you need to do so (for example, if a cache storage device fails).

Cache storage can be protected by maintaining it on high-availability storage devices. If no high-availability storage is available, IBM recommends that backups of reports in cache storage (the file systems) be taken on a regular basis.

Loading a previously indexed AFP file

About this task

If AFP files are previously indexed, that is, the `*.out`, `*.ind`, `*.res` files are already in the indexing directory, you can use ARSLOAD to load the `.out` files into the system. When you load the files, specify the file names without the `.out` extension. That is, instead of specifying `filename`.out, just specify `filename`.

Loading metacode documents in large object format

About this task

Large object documents are stored in groups of pages called segments, which can be retrieved individually to improve performance. You can set the number of pages in a segment in the application. In order to load documents in large object format, the loading process must create an index file that contains, in addition to the group information, offset and length information for each page in the document. OnDemand uses Xenos to create the index file when it is loading metacode documents, the Xenos script must be written to create the index file with the necessary information.

For more information, see the "JS program reference" section in the *Indexing Reference*.

Loading image files

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, Metacode/DJDE, 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 32 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 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
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 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 173.

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 a TIFF to 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 = >5242-3-0-27FAA-12461-12466<
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. If you need to save scanned image files as uncompressed TIFF images, then you should select one of the compression methods supported by Content Manager OnDemand

so that OnDemand can compress the input files to improve storage efficiency and retrieval performance.

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 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 173. 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, Metacode/DJDE, PCL, 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 45 on page 178 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 for Multiplatforms: 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 Unicolas
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 45. 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 (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) 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 "ARSLOAD" on page 279 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 from the command line or configure it to run as a daemon (UNIX servers) or service (Windows servers) 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 command line. For example:

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

Specify the following parameters to the ARSLOAD program:

-u admin
> The name of an 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 file to process.

In the example, the ARSLOAD program locates 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 177 (which is a parameter file for the Generic indexer). Because the full path names of the input files were not specified in the parameter file, the input files must be in the directory from which the ARSLOAD program is started.

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 = >6927-0-0-1FAA-0-0<
  Loaded 4 rows into the database
arsload: Mon Dec 18 13:05:14 MST 1995 Loading Completed
arsload: Processing successful for file >letters<
```

You can verify the number of 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.

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, Metacode/DJDE, 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 32 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 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 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 181.

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 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 181. 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 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 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, Metacode/DJDE, 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 46 on page 186 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 for Multiplatforms: 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 46. 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 OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) 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 "ARSLOAD" on page 279 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 a file or set of files into the system. You can either run the ARSLOAD program from the command line or configure it to run as a daemon (UNIX servers) or service (Windows servers) 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 command line. For example:

```
/usr/lpp/ars/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 an 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 the index data" on page 185 (which is a parameter file for the Generic indexer). Because the full path names of the input files were not specified in the Generic index file, the input files must reside in the directory from which you start the ARSLOAD program.

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.

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.

Deleting a report

About this task

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 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. (Search for message number 87 in the system log.) 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. By default, the UNLOAD function deletes the storage objects from all locations. The UNLOAD function also deletes the rows of index data that were added to the database during the original load process.

The following shows how to run the ARSADMIN program on a UNIX server from the command line to delete a load from all of the storage locations:

```
OnDemand install directory/bin/arsadmin unload -h instance
-g Application Group name -L loadID -u user -p password
```

To UNLOAD a report on a Windows server:

Procedure

1. Start → Programs → IBM Content Manager OnDemand for Windows → Command Window.
2. Run the ARSADMIN program from the prompt. For example:
   ```
   arsadmin unload -h instance -g Application Group name
   -L loadID -u user -p password
   ```

Results

When the ARSADMIN program completes, you should open the System Log folder to view the messages that were generated during the UNLOAD process. (Search for message number 85 in the system log.) See *IBM Content Manager OnDemand: Messages and Codes* for more information about messages.

The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) must be running, otherwise the ARSADMIN program will fail.

See "ARSADMIN" on page 215 for more information about the ARSADMIN program and the parameters that you can specify to UNLOAD data.

Download exits and processing

Using Download

You can use Download to transmit reports from z/OS systems to Content Manager OnDemand servers. If you use Download to automate the data loading process, then you may need to provide a user-written program to process the files transmitted by Download before they can be processed by the ARSLOAD program. You must provide a user-written program if the file name does not contain information that can be used to identify the application group to load.

For example, suppose that you use a report distribution system to place the output of your application programs on the spool data set. Download selects the output and transmits the data to an Content Manager OnDemand server. However, after the output has been processed by the report distribution system, the resulting file name on the spool data set can no longer be associated with the application program that generated the output. Therefore, the ARSLOAD program cannot use the file name to determine the application group to load. (And because the file name does not contain information that can use used to identify the application group and application to load, you cannot run the ARSLOAD program with the **-A** and **-G** parameters.) You must find some other way to identify the application group and application to load.

Download provides a user exit (APSUX15) that allows you to provide additional job information to Content Manager OnDemand. Download includes the additional job information in the data stream that is transmitted from the spool data set to the server. The additional job information is installation dependent. See *PSF for z/OS: Download for z/OS* for details about the APSUX15 user exit and the content, format, and purpose of the additional job information. The processing done by the user-written program is also installation dependent. See your Infoprint Manager or PSF information for information about processing the additional job information with a user-written program.

The ARSJESD program is the component of Download for OS/390® that runs on the workstation. The **-x** parameter of the ARSJESD program may be used to specify the name of a user-written program to process additional job information sent by PSF through the APSUX15 user exit. If the ARSJESD program was invoked with the **-x** parameter, it calls the specified user-written program. The ARSJESD program passes the file name and the additional job information to the user-written program.

Invoking the Download user exit

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

The ARSJESD program receives the data sets into file systems on the server. If you start the ARSJESD program with the **-x** parameter, the ARSJESD program invokes the specified user-written program. The program specified with the **-x** parameter can be any user-written program.

For example, you could provide a user-written program that parses the additional job information transmitted by Download and the APSUX15 user exit program. The user-written program could extract the value of the WRITER parameter to identify the application to load. Using this value, the user-written program could then query the database to determine the name of the application group to which the application belongs. The user-written program could then run the ARSLOAD program with the **-a** parameter to identify the application to load and the **-g** parameter to identify the application group to load. (The user-written program could also rename the input file and then run the ARSLOAD program with the **-A** parameter to specify the part of the new file name that identifies the application and the **-G** parameter to specify the part of the new file name that identifies the application group.)

By using the Download user exit program, the **-x** parameter with the ARSJESD program, and a user-written program, you can configure the system so that each file that Download transmits to the server is automatically processed and loaded into the correct application group and application.

See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about using the ARSJESD program and the Download user exit.

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 an OnDemand system. The administrative objects that can be imported and exported include:

- users
- groups
- printers
- storage sets
- application groups
- applications
- folders
- cabinets
- holds

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.

If you create an XML file from scratch, ensure that you include objects following this order:

1. users
2. groups
3. printers
4. storage sets
5. application groups
6. applications
7. folders
8. cabinets
9. holds

Installing batch system administration

This section discusses how to install and setup the prerequisites for OnDemand batch system administration, and guides you through a short installation verification process.

Prerequisites

For batch administration requirements, see http://www.ibm.com/support/
docview.wss?rs=129&uid=swg27016455 or search for 7016455 at
http://www.ibm.com

The following files are included with the OnDemand batch system administration:
- /bin/arsxml
- /bin/arsxml.exe (Windows)
- /bin/xml/ondemand.xsd - OnDemand XML Schema file
- /bin/xml/samples/addgroups.xml
- /bin/xml/samples/addusers.xml
- /bin/xml/samples/deletegroups.xml
- /bin/xml/samples/deleteusers.xml
- /bin/xml/samples/exportgroups.xml
- /bin/xml/samples/exportusers.xml
- /bin/xml/samples/updategroups.xml
- /bin/xml/samples/updateusers.xml - Sample OnDemand XML files

Installation verification

Before you run the batch system administration functions, make sure that the
OnDemand environment is properly set up:

On Windows systems:
>   Open an OnDemand command prompt by selecting **Start -> Programs ->
>   IBM OnDemand for Windows -> Command Window**.

On UNIX systems:
>   Set up the PATH environment variable to include the OnDemand bin
>   directory `/usr/lpp/ars/bin`. Use this command:
>
>   `export PATH=${PATH}:/usr/lpp/ars/bin`

On other systems:
>   Set up the PATH environment variable to include the OnDemand bin
>   directory `/opt/ondemand/bin`. Use this command:
>
>   `export PATH=${PATH}:/opt/ondemand/bin`

To run the ARSXML command, change directories to the OnDemand XML samples
directory.

Run this command:

`arsxml add -h <hostname> -u <user> -p <password> -i addusers.xml -v`

where

<hostname>
>   Hostname for the OnDemand system

<user>
>   An OnDemand user which has authority to add users and printers

<password>
>   The password for the OnDemand user. If the user does not have a
>   password, they meed to specify -password " "

The XML file contains user passwords that are between 6 and 8 characters long. If the OnDemand system where these users are created has different password restrictions, these passwords might need to be changed before you run the command.

After you run the command, you should receive several messages stating that a printer, and five users have been added successfully.

If the command is properly run, you should be able to use the OnDemand system administrator GUI to view the newly added users and printer.

To remove the newly added objects, run this command:

```
arsxml delete –h <hostname> -u <user> -p <password> -i deleteusers.xml –v
```

There are several other sample XML files that can be used. However, they depend on the users that are created by the `addusers.xml` file.

Importing an XML file into an OnDemand system

About this task

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

Procedure

1. Preparing an XML file for the import process
2. Importing the XML file by using the ARSXML command

Results

Preparing an XML file for the import process
About this task

Before importing data into an 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 from scratch. If you decide to develop an XML file from scratch, you must follow the format of the OnDemand XML schema file.

The OnDemand XML schema file defines the syntactical format for all 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.

The following is an example of a portion of the OnDemand XML file, which contains an OnDemand administrative user named `bill` and a user group named `SpecialGroup`. `SpecialGroup` is owned by the user `admin`, and contains a single user `bill`.

```
<user name="bill"
  password=""
  description="This is an Admin user."
  email="Sample@us.ibm.com"
  userType="User Admin">
</user>
```

```
<group name="SpecialGroup"
   description="A Special Group for special users"
   ownerUser="admin">
  <user name="bill"/>
</group>
```

Creating an XML file

Occasionally, you need to create an 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 OnDemand. For more detailed information on XML and specific XML
syntax information, reference XML documentation.

Overall file structure

An OnDemand XML file is a basic text file that can be created by the user.

Every OnDemand XML file will contain the following:
- An XML identifier tag
- An OnDemand identifier tag
- One or more OnDemand objects
- An 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"?>
> ```

OnDemand identifier tag
> The OnDemand identifier tag specifies what the XML file is used for and
> what schema file to use. The standard 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 OnDemand XML schema file which is located
> in the OnDemand bin/xml directory.

OnDemand objects
> See "Objects and data model used in the OnDemand XML file" on page
> 351 for objects and data model used in the XML file.

OnDemand ending tag
> The OnDemand ending tag indicates the end of the OnDemand XML file.
> The ending tag is:
> ```
> </onDemand>
> ```

Every standard 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">
   ... OnDemand Objects ...
</onDemand>
```

OnDemand objects

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

The following objects can be included within the OnDemand XML file:

- user
- group
- printer
- storageSet
- applicationGroup
- application
- folder
- cabinet
- hold

These objects as well as all of the child objects are shown in detail in the XML data tables in "XML objects in the OnDemand XML file" on page 351. The OnDemand objects can occur in any order within the XML file, but can not be imbedded within each other.

Ensure that you include the objects in this order:

1. user
2. group
3. printer
4. storageSet
5. applicationGroup
6. application
7. folder
8. cabinet
9. hold

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 OnDemand XML file" on page 351.

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 time out 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 OnDemand XML file" on page 351.
2. Unless you set Case Sensitivity as one of the system parameters, when you add a user, 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, OnDemand automatically converts it to SAMPLEUSER.

Child objects

Child objects are constructed the same way as the nine main 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 OnDemand XML file" on page 351 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="84000">
</group>
```

is equivalent to:

```
<group name="Sample" gid="84000"/>
```

Examples

The following is a complete 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" >
  <permission group="*PUBLIC">
    <fieldInfo name="cost" user="SampleUser"
    sortOrder="2" greaterThan="Default" lessThan="Yes"/>
  </permission>
 </folder>
</onDemand>
```

Creating 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:

Table 10. XML specification for special characters

XML code	Character
&	&
'	'
"	"
<	<
>	>

Your input file might look like this:

```
...
<user
    name="DBRYANTDEU"
    acctInfo="Engraving &amp; 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 OnDemand XML files for delete and export

When creating an 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 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<printer name="_ALL"/>
```

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 OnDemand system. For syntax, description, and parameters of this command, see "Purpose" on page 307.

Exporting OnDemand administrative objects to an XML file
About this task

You can use the ARSXML command to export objects from 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 an 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 "Purpose" on page 307.

If you are exporting objects that have dependencies on other objects: Some objects in 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 "Purpose" on page 307 for more example XML files that are generated in different export scenarios.

Command overview

The commands contained in this reference are Content Manager OnDemand programs that you can use to perform server, archive storage manager, and database manager functions. The commands are presented in alphabetical order. Each command contains a description of its purpose and syntax (including descriptions of the parameters that can be used). Examples and general information about using the commands have also been included.

Examples of commands in this book may appear on one or more lines, for example:

```
arsdate -z "09/01/99 04:00:00"

09/01/99 04:00:00 -> 936180000
```

However, when you enter the command information at the prompt, you must type the entire command, parameters, and values on a single line.

The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) must be running; otherwise the commands will fail.

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) in uppercase letters; however, at the command prompt, you can enter keywords in either uppercase or lowercase. An example of a keyword is a command name. On UNIX servers, command names are case sensitive. On Windows servers, command names are not case sensitive.
- Variables represent names or values you supply and are shown (in syntax) in lowercase letters; however, at the command prompt, you can enter variables in either uppercase or lowercase unless the syntax directions explicitly state the case restrictions. An example of a variable is a file name.

  On UNIX servers, file names are case sensitive. On Windows servers, file names are not case sensitive.

A parameter can be a combination of a keyword and a variable.

Required parameters are displayed on the main path.

```
►►──COMMAND──required parameter──────────────────────────────────────────►◄
```

Optional parameters are displayed below the main path.

```
►►──COMMAND──────────────────────────────────────────────────────────────►◄
            └─optional parameter─┘
```
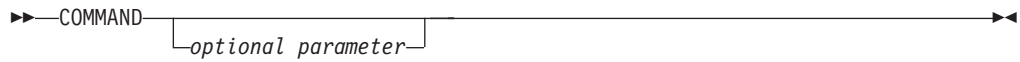
A stack of parameters, with the first parameter displayed on the main path, shows that you must choose one of the parameters.

```
►►──COMMAND──┬─required choice 1─┬────────────────────────────────────────►◄
             └─required choice 2─┘
```
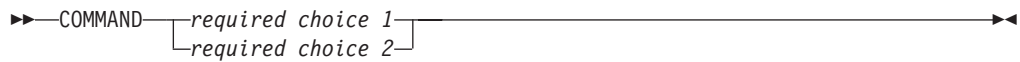
A stack of parameters, with the first parameter displayed below the main path, shows that you can choose one of the parameters.

```
►►──COMMAND──────────────────────────────────────────────────────────────►◄
            ├─optional choice 1─┤
            └─optional choice 2─┘
```

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.

```
            ┌─────────────────────┐
►►──COMMAND─▼─repeatable parameter─┴──────────────────────────────────────►◄
```

- If the repeat arrow contains a comma, the item can be repeated in a list with the items separated by commas.

```
            ┌─,───────────────────┐
►►──COMMAND─▼─repeatable parameter─┴──────────────────────────────────────►◄
```

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).

```
                      ┌──────────,─────────────────────────────────┐
  ►►──COMMAND──────────▼────────────────────────────────────────────┴──►◄
                      ├─parameter choice1────────────────────────────┤
                      ├─parameter choice2────────────────────────────┤
                      │                  ┌─parameter choice2a─┐       │
                      │                  ├─parameter choice2b─┤       │
                      │                  └─parameter choice2c─┘       │
                      └─parameter choice3────────────────────────────┘
```
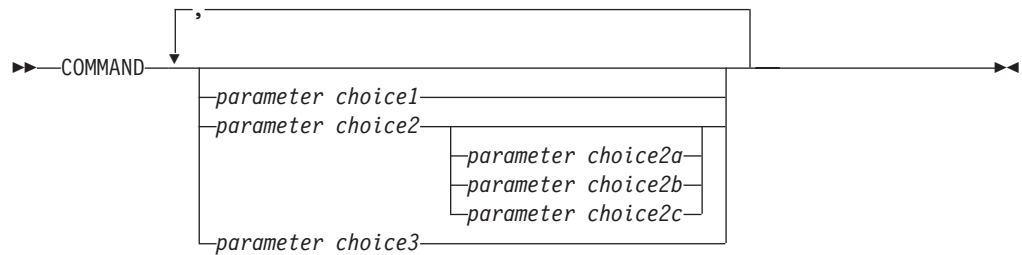
Some commands are preceded by an optional path parameter.

```
  ►►──────────COMMAND──────────────────────────────────────────────►◄
      └─path─┘
```

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:

```
  ►►──┬─COMMAND FORM1─┬──────────────────────────────────────────────►◄
      └─COMMAND FORM2─┘
```

The description of the command directs you to the correct format to use.

Using quotes in commands

About this task

When you run an 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 quote characters.

For example, when running the ARSDOC GET program from the command line
under UNIX 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 quote characters. For
example:

```
  -o "(sdate)(student)"
```

However, if you were to run the same program under Windows, then you would
not need the double quote 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.

Removing trailing blank characters

About this task

Trailing blank characters are trimmed from the name when adding or updating an application, application group, folder, printer, storage set, or 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 (UNIX servers; the registry for Windows servers) 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.

Working with instances

An Content Manager OnDemand instance is a logical server environment made up of a database, a library server, and one or more object servers. An instance is defined in the ARS.INI file by specifying a unique name for the instance, identifying the userid of the instance owner, and identifying the library server on which the database will be maintained.

You can run multiple instances on the same workstation or node, with each instance configured differently:

- To have separate test and production environments
- To have databases using different code pages

Each instance has different security from other instances on the same machine. You must define users and groups to each instance and set application group and folder permissions for users of each instance. Each instance has its own system log.

When you work with more than one instance, you must identify the instance name when you run Content Manager OnDemand programs (such as ARSDB, ARSLOAD, and ARSSOCKD) and database commands (such as connecting to the database).

When the OnDemand client programs connect to an instance, they must specify the TCP/IP port number over which the instance listens for requests from clients. Each instance that is operating on the same workstation must use a different TCP/IP port number.

See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for information about configuring and creating instances, including:
- Adding an instance to the ARS.INI file
- Modifying the server configuration in the ARS.CFG file
- Defining table space file systems in the ARS.DBFS file
- Defining cache file systems in the ARS.CACHE file
- Automating instance operations

The following topics in this section describe how to work with Content Manager OnDemand instances:
- Manually starting and stopping instances
- Connecting to instances
- Working with instances

Manually starting and stopping instances

You must start an instance before clients can connect to databases on the instance.

To start an instance:
1. Log on as the `root` user or instance owner.
2. Start the instance:

   ```
   arssockd test arssockd
   ```

   Where `test` is the name of the instance that you want to start. The name `test` is the value of the SRVR_INSTANCE parameter in the ARS.INI file.
3. Use the PS command to verify that the instance started:

   ```
   ps -ef ¦ grep ars

   root 6014 28991 0 60 13:30:13    -  0:00 arssockd -test:  (accepting)
   ```

To stop an instance:
1. Log on as the `root` user or the instance owner.
2. Use the ARSSOCKD command to stop the initial ARSSOCKD process. If there is more than one ARSSOCKD process running, determine the initial ARSSOCKD process. The initial ARSSOCKD process has a status of `accepting`:

   ```
   ps -ef ¦ grep ars

   root 6022 28991 0 60 13:43:03    -  0:00 arssockd:  db  (USERID)
   root 6018 28991 0 60 13:33:03    -  0:00 arssockd:  db  (idle)
   root 6014 28991 0 60 13:30:13    -  0:00 arssockd -arssockd:  (accepting)
   ```
3. Then stop the `accepting` process. For example:

   ```
   arssockd stop 6014
   ```

The initial process and all of its children will be stopped.

Note:

1. When you stop the server process, all users who are connected over the network to the OnDemand system are disconnected. Therefore, it is a good idea to warn connected users before stopping the server process.

2. The ARSSOCKD STOP command is for UNIX servers only. For Windows servers, the server process runs as a service. The service can be stopped by using the Services administrative tool in Windows or the OnDemand Configurator client.

3. To stop an instance when you are running more than one instance on the same workstation or node, make sure that you kill the correct accepting ARSSOCKD process. There is no instance name associated with the accepting ARSSOCKD process for the default instance (the default instance name is `archive`). The following shows an example of the processes for two instances: the default instance and an instance named `test`:

   ```
   ps -ef ¦ grep ars

   root 3326 25955 0 60 10:03:59    - 0:00 arssockd:  db  (USERID)
   root 3303 25955 0 60 10:03:31    - 0:00 arssockd:  db  (idle)
   root 3299 25955 0 60 10:01:45    - 0:00 arssockd:  (accepting)
   root 6022 28991 0 60 13:43:03    - 0:00 arssockd:  db  (USERID)
   root 6018 28991 0 60 13:33:03    - 0:00 arssockd:  db  (idle)
   root 6014 28991 0 60 13:30:13    - 0:00 arssockd -TEST:  (accepting)
   ```

   In the example, process 3299 belongs to the default instance and process 6014 belongs to the instance named `test`.

Connecting to instances

Connecting from Content Manager OnDemand clients
To connect to a particular instance, the client must log on to the correct library server. Use the Update Servers dialog box to add library servers to the client. When you add a library server, you identify the host name of the library server and the port number and communications protocol used to communicate with the instance. The port number that you specify in the Update Servers dialog box must be the same as the value that you specified for the PORT parameter in the ARS.INI file.

Connecting to a database
You can connect to a database using the DB2 command line processor or other tools provided with DB2. To connect to and use a database, log on as `root`, the instance owner, or some other user that belongs to the instance owner's primary group.

You can connect to only one database at a time. If you are currently connected to a database, you must disconnect from that database before you can connect to a different database.

Working with instances

Content Manager OnDemand provides programs that you can use to work with an instance. For example, Content Manager OnDemand provides programs to create databases, start instances, load data, and maintain index data. One of the parameters that you can specify to a program is the name of the instance. If you do not specify otherwise, the name of the instance is `archive`. If you name your first or only instance something other than `archive` or you work with more than one instance, you should always specify the name of the instance when you run a

program. That way you can make sure you are working with the correct instance. Depending on the program that you need to run, there are different ways that you can specify the name of the instance.

The ARSADMIN, ARSDB, ARSLOAD, ARSMAINT, and ARSSYSCR programs use the –I instance parameter. Content Manager OnDemand retrieves the host name of the library server that is associated with the instance from the HOST parameter in the ARS.INI file. For example:

```
arsdb -I test -gcv
arssyscr -I test -l
arsload -I test -d /arsacif/test/acif1 -c /arsacif/test/acif2
```

Where test is the name of the instance, and matches the value of a SRVR_INSTANCE parameter in the ARS.INI file.

The ARSSOCKD and ARSOBJD programs use the following syntax:

```
arssockd test arssockd
arsobjd test arsobjd
arssockd stop test
```

Where test is the name of the instance, and matches the value of a SRVR_INSTANCE parameter in the ARS.INI file.

ARSADMIN

Purpose

The ARSADMIN program is the primary Content Manager OnDemand input data processing program. The ARSLOAD program calls the Load function of the ARSADMIN program to process the input file, index file, and optionally, resource group file to create index data and storage objects. The ARSADMIN program processes the output files that are produced by the various data indexing and conversion programs (output files, index files, and optionally, resource group files) to load index data into the Content Manager OnDemand database and copy documents and resources to cache storage and archive storage.

The Import function can be used to import migrated index data from archive storage to the database.

The ARSADMIN program can also be run from the command line by an experienced Content Manager OnDemand administrator to help with problem determination, error recovery, and other administrative tasks. For example, an experienced Content Manager OnDemand administrator can create, delete, and restore storage objects, retrieve individual documents and resources from storage volumes, and calculate data compression ratios.

Description and syntax

The ARSADMIN program provides the following functions:

Load. The Load function creates storage objects and index data. The storage manager copies the storage objects to cache storage and archive storage. The database manager loads the index data into the database. The input data to the Load function can be one or more of the following: an input file (specify with the **-s** parameter), an index file (specify with the **-i** parameter), and a resource group file (specify with the **-r** parameter). If you use ACIF to index a report, then you can concatenate the three input files into a single input file. If you do so, then the **-s**, **-i**, and **-r** parameters must specify the same file name. The index data that the Load function creates can be saved to a file.

```
►►──arsadmin load──────────────────────────────────────────────────────────►
                   └─-a──application─┘  └─-d──directory─┘  └─-e──resourceID─┘

►──────-g──applGroup──-h──instance───────────────────────────────────────────►
    └─-f─┘                       └─-i──indexFile─┘

►────────────────────────────────────────────────────────────────────────────►
    └─-n──prinid-secnid─┘  └─-o──outputFile─┘  └─-p──password─┘

►────────────────────────────────────────────────────────────────────────────►
    └─-r──resourceFile─┘  └─-s──sourceFile─┘  └─-u──userid─┘  └─-1──trace_file─┘

►────────────────────────────────────────────────────────────────────────────►◄
    └─-2──level─┘  └─-y──delimiter─┘  └─-Z──userExit─┘
```

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 administrative 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.

```
►►──arsadmin unload──-g──applGroup──-h──instance──-L──loadID──────────────────►
                                                          └─-p──password─┘

►──┬──────┬──┬──────────────┬──┬──────────────────┬──┬───────────┬──────────►◄
   └─-Q─┘   └─-u──userid─┘    └─-1──trace_file─┘     └─-2──level─┘
```

Load_db. The Load_db function stores the index data that was created by the Load function into the database. If you specify Postprocessing information for the application (on the Load Information page), the index data is processed as specified before the Load_db function stores the index data in the database.

```
►►──arsadmin load_db──┬──────────────────┬──-g──applGroup──-h──instance──────►
                      └─-a──application─┘

►──-L──loadID──┬──────────────┬──-s──sourceFile──┬──────────────┬────────────►
               └─-p──password─┘                  └─-u──userid─┘

►──┬────────────────┬──┬───────────┬──┬────────────────┬──┬──────────────┬───►◄
   └─-1──trace_file─┘   └─-2──level─┘   └─-y──delimiter─┘    └─-Z──userExit─┘
```

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 (UNIX servers) or the console (Windows servers) 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 retry the query.

Note: Importing migrated index data requires additional space in the database, additional log file storage, and additional temporary space.

```
►►──arsadmin import──-g──applGroup──-h──instance──┬──────────────┬───────────►
                                                  └─-p──password─┘

►──┬──────────────┬──────tableName──┬──────────────┬──┬───────────┬──────────►◄
   └─-u──userid─┘                   └─-1──trace_file─┘   └─-2──level─┘
```

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.

```
►►──arsadmin store─────────────────────────-g──applGroup──-h──instance──────────────────►
                    └─-d──directory─┘
```

```
►──────────────────────────-n──prinid-secnid──────────────────────────────────────────►
      ┌─7─┐                                  └─-p──password─┘  └─-u──userid─┘
   -m─┼─1─┤
      ├─2─┤
      ├─3─┤
      ├─4─┤
      ├─5─┤
      └─6─┘
```

```
►──▼─filename──────────────────────────────────────────────────────────────────────►◄
                 └─-1──trace_file─┘  └─-2──level─┘
```
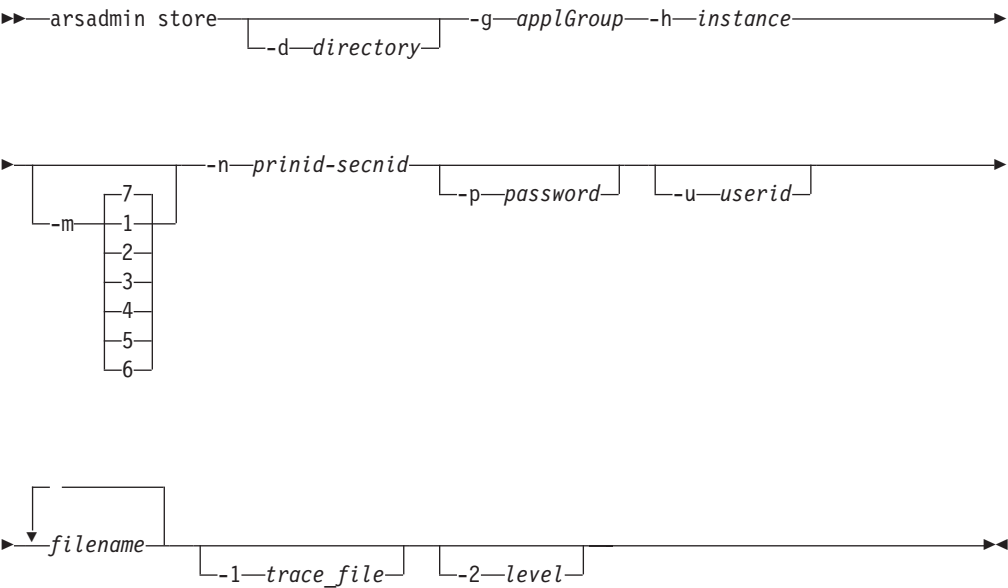
Retrieve. The Retrieve function can be used to retrieve a storage object from a specified location. For example, you could retrieve a storage object from cache storage or a primary storage node. After retrieving a storage object, you could use the Decompress function to extract a specific document or resource from the storage object file.

```
►►──arsadmin retrieve──────────────────────────────────────-g──applGroup──────────────►
                       └─-b──offset─┘  └─-d──directory─┘
```

```
►──-h──instance─────────────────────────────-n──prinid-secnid─────────────────────────►
                 └─-l──length─┘      ┌─7─┐
                                  -m─┼─1─┤
                                     ├─2─┤
                                     ├─3─┤
                                     ├─4─┤
                                     ├─5─┤
                                     └─6─┘
```

```
►──────────────────────────▼─filename──────────────────────────────────────────────►
      └─-p──password─┘  └─-u──userid─┘          └─-1──trace_file─┘
```

```
►────────────────────────────────────────────────────────────────────────────────►◄
   └─-2──level─┘
```
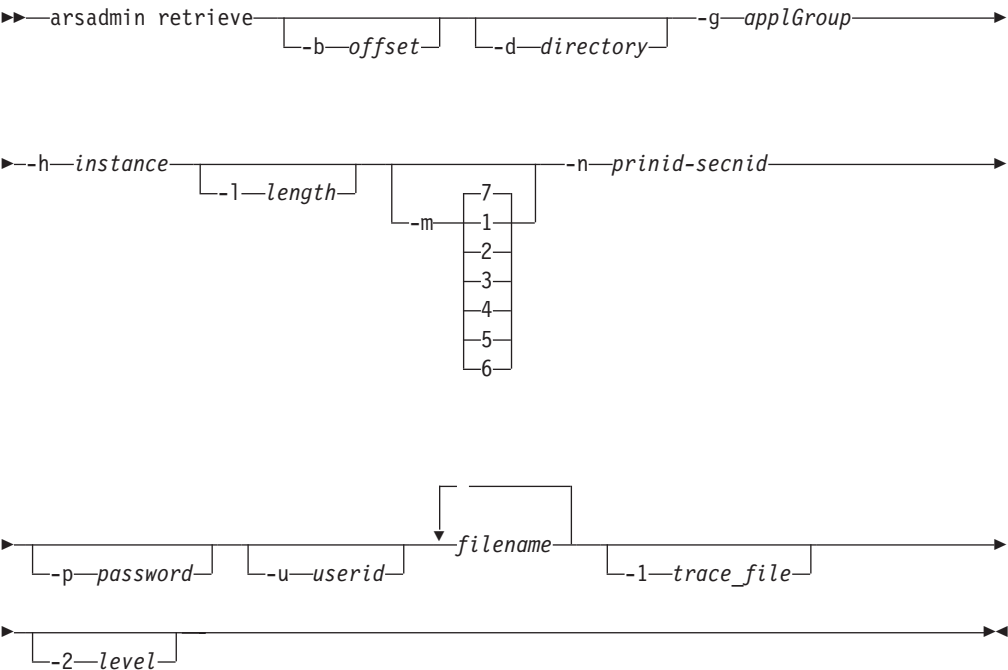
Compress. The Compress function is used to compress input files and resources in preparation for creating storage objects.

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 **-l** 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 OnDemand (LZW12, LZW16, OD77, and OD77Lite). Compare the compressed file sizes and the time required to compress the data to determine the best compression method for the input data.
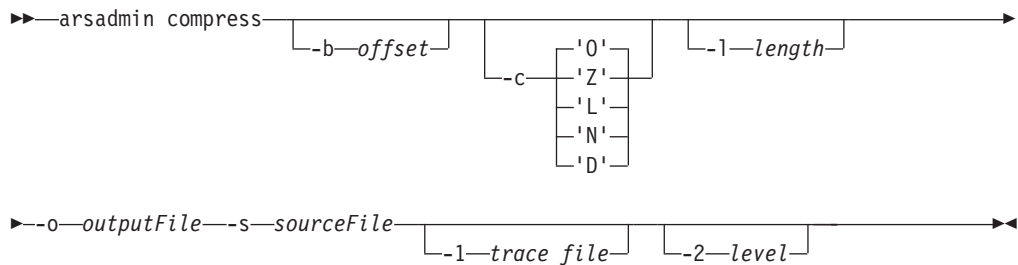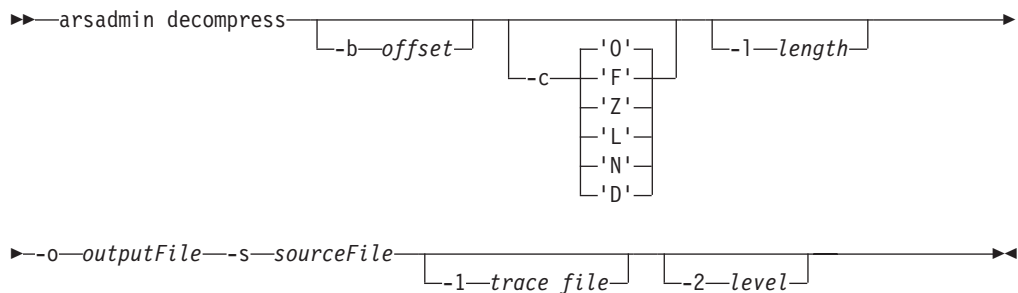
```
►►──arsadmin compress──────────────────────────────────────────────────────►
                          └─-b─offset─┘    ┌─'O'─┐    └─-l─length─┘
                                        -c─┼─'Z'─┤
                                           ├─'L'─┤
                                           ├─'N'─┤
                                           └─'D'─┘

►──-o─outputFile──-s─sourceFile────────────────────────────────────────────►◄
                               └─-1─trace_file─┘  └─-2─level─┘
```

Decompress. The Decompress function can be used to extract a specific document or resource 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.

```
►►──arsadmin decompress─────────────────────────────────────────────────────►
                           └─-b─offset─┘    ┌─'O'─┐    └─-l─length─┘
                                         -c─┼─'F'─┤
                                            ├─'Z'─┤
                                            ├─'L'─┤
                                            ├─'N'─┤
                                            └─'D'─┘

►──-o─outputFile──-s─sourceFile────────────────────────────────────────────►◄
                               └─-1─trace_file─┘  └─-2─level─┘
```

Res_diff. The Res_diff function can be used to determine whether the resources contained in the specified resource group file already exist on the server.

```
►►──arsadmin res_diff──────────────────-g─applGroup──-h─instance─────────────►
                        └─-a─application─┘
```

```
►►──-n──prinid-secnid─────────────────────────────────────▼──filename─────────►
                    └─-p──password─┘  └─-u──userid─┘
```
```
►──────────────────────────────────────────────────────────►◄
   └─-1──trace_file─┘  └─-2──level─┘
```
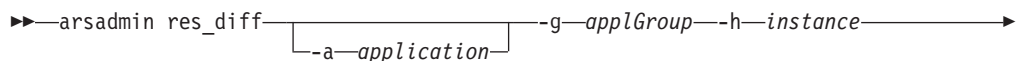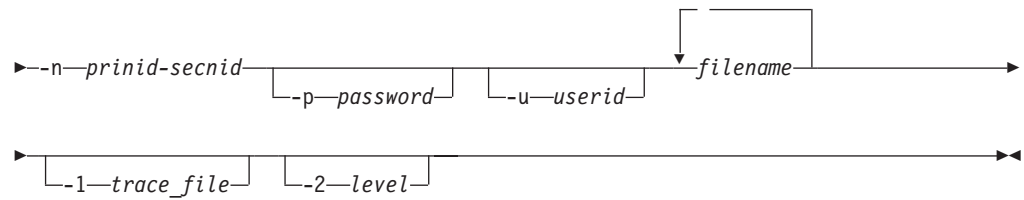
Postprocess_cmd. The Postprocess_cmd function can be used to print the Postprocessor information for the specified application. (The Postprocessor information is specified on the Load Information page in the application.) The information is printed to stdout (UNIX servers) or the console (Windows servers).

```
►►──arsadmin postprocess_cmd───────────────────────-g──applGroup──-h──instance──►
                            └─-a──application─┘
```
```
►───────────────────────────────────────────────────────────►◄
   └─-p──password─┘  └─-u──userid─┘  └─-1──trace_file─┘  └─-2──level─┘
```

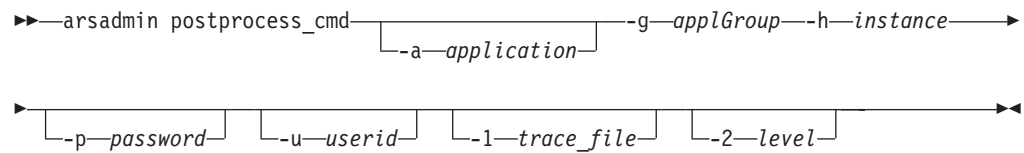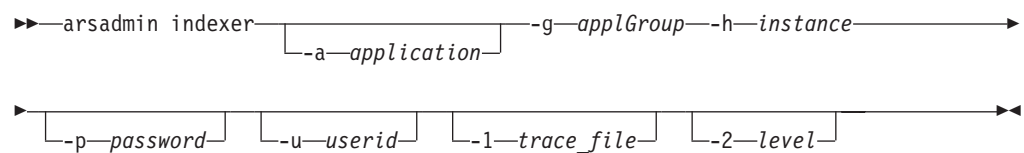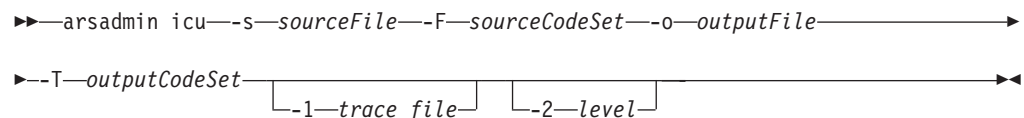**Indexer**. The Indexer function can be used to print the indexing parameters for the specified application. (The indexing parameters are specified on the Indexer Information page in the application.) The indexing parameters are printed to stdout (UNIX servers) or the console (Windows servers).

```
►►──arsadmin indexer────────────────────-g──applGroup──-h──instance──────────►
                      └─-a──application─┘
```
```
►───────────────────────────────────────────────────────────►◄
   └─-p──password─┘  └─-u──userid─┘  └─-1──trace_file─┘  └─-2──level─┘
```

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.

```
►►──arsadmin icu──-s──sourceFile──-F──sourceCodeSet──-o──outputFile─────────────►
```
```
►──-T──outputCodeSet────────────────────────────────────────►◄
                     └─-1──trace_file─┘  └─-2──level─┘
```

Msgs. Adds messages to the System Log.

```
►►──arsadmin msgs──-h──hostname──-p──password──-s──src_file──-u──user ID────────►
```
```
►──-M──msg_num──────────────────────────────────────────────►◄
               └─-1──trace_file─┘  └─-2──level─┘  └─input string argument─┘
```
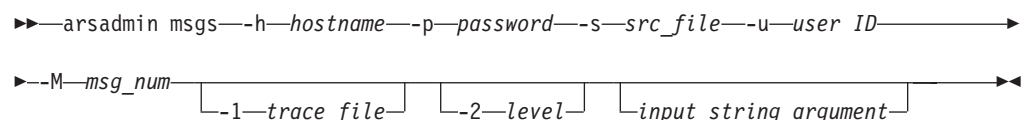
Psf_options. Writes the PSF Print Options on stdout.

```
►►──arsadmin psf_options──-a─name──-g─name──-h─hostname──-p─password──────────►

►──-u─use login name────────────────────────────────────────────────────────►◄
            └─-1─trace_file─┘  └─-2─level─┘
```

Trace. Converts the trace that is generated by any Content Manager OnDemand
command into a readable XML file.

```
►►──arsadmin trace──-o─outputFile──-s─sourceFile──────────────────────────────►◄
```

Parameters

-a application
> The name of the Content Manager OnDemand application. If there is more
> than one application contained in the application group, then you must
> specify this parameter.

-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'**
> no compression, and **'D'** disable compression.

-d directory
> For the LOAD function, if you specified the **-f** parameter to create storage
> object files, you can use this parameter to specify the directory in which
> the ARSADMIN program writes the storage object files. The default is the
> directory from which the ARSADMIN program was invoked.
>
> 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.

-e res_ID
> For AFP files, this parameter and the **-r** parameter determine how Content
> Manager OnDemand processes the resource group file. The default
> resource ID is 0 (zero).

- If you specify a resource ID and a resource file (with the **-r** parameter), then Content Manager OnDemand loads the resource from the file and overrides the resource specified in the document with the resource ID you specified.
- If you specify a value of 0 (zero) and a resource file (with the **-r** parameter), then Content Manager OnDemand checks the server to determine if the resource specified in the document exists. If the resource does exist, then Content Manager OnDemand does not load the resource from the file. Otherwise, Content Manager OnDemand loads the resource from the file.
- If you specify a value of 0 (zero) and do not specify a resource file (with the **-r** parameter), then Content Manager OnDemand does not assign a resource ID to the document.
- If you specify a value other than 0 (zero) and do not specify a resource file (with the **-r** parameter, then Content Manager OnDemand assigns the resource ID that you specified to the document.

-f   For the LOAD function, write storage objects to file, rather than to storage volumes. By default, Content Manager OnDemand creates storage objects in memory and copies them to cache storage and archive storage. You can specify this parameter when you want Content Manager OnDemand to process an input file and create storage objects, but you do not want to copy the storage objects to storage volumes. You may want to use this option when you are testing the data loading process.

-F   For the ICU function, specifies the code set for the document specified by the **-s** 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 OnDemand instance to process. The ARSADMIN program will attempt to locate the specified instance name in the ARS.INI file (UNIX servers) or the registry (on Windows servers) 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, DECOMPRESS, and ICU.

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 Multiplatforms: Installation and Configuration Guide* for information about configuring instances.

-i indexFile

The name of the input index file created by the data indexing program. Optionally, for AFP data, the name of a concatenated input file that also contains resource group and report data. (See also the **-r** and **-s** parameters.)

-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.

For the DECOMPRESS function, the number of bytes (from the offset) of data to decompress. The default is zero bytes.

-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 six parts. For example:

```
1220-1-0-8FAA-9339-9345
```

Where 1220 is the internal application group identifier, 1 is the primary storage node, 0 is the secondary storage node, 8FAA represents the name of the object that was stored during the load process, 9339 is the first (internal) date in the report, and 9345 is the last (internal) date in the report. 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.

For the LOAD_DB function, the Load ID that represents the input file from which the index data was generated.

In the administrative 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** (seven), 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** (one) for cache storage, **2** (two) for the primary storage node, or **4** (four) for the secondary storage node. The values are additive; that is, if you specify a value of **3** (three), then the storage manager processes the storage objects in cache storage and the primary storage node.

-M msg_num

Specifies the message number of the message that should be added to the system log. You can use multiple input string arguments at the end of the command.

-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 LOAD function, optionally create a file that contains the index information that the database manager uses to update the Content Manager OnDemand database. You can use this option to create the index data, but not load it into the database; for example, if you want to test the index process or inspect the index data.

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 or resource.

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

The password for the Content Manager OnDemand userid specified with the **-u** parameter. If the userid is not assigned a password, then enter a null password (that is, specify `-p ""`). If you do not specify this parameter, then the ARSADMIN program will prompt you to enter the password. If the userid 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.

-r resourceFile

The name of the input resource group file. Created by the data indexing program. Optionally, for AFP data, the name of a concatenated input file that also contains index and report data. (See also the **-i** and **-s** parameters.)

-s sourceFile

For the LOAD function, the input file or a concatenated file that contains the report data, index data, and resource group data. (See also the **-i** and **-r** parameters.)

For the LOAD_DB function, the name of the file that contains the index data to load.

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.

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.

Important: On the Windows platform, this file cannot be a link to another file.

| 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
An Content Manager OnDemand userid with administrator authority for the application group. If you do not specify this parameter, then the ARSADMIN program prompts you to enter the userid.

For the UNLOAD function, the userid that you specify must have permission to delete documents from the application group.

-y delimiter
The column delimiter. Optionally specify the character that separates the columns (fields) in the index records that are in the index file to be processed by the database manager. The default delimiter is the pipe character.

Content Manager OnDemand uses the delimiter when you specify the **-o** parameter to save the index records to an output file.

-Z userExit
For the LOAD and LOAD_DB functions, specifies a user-defined string that is passed to the load user exit program (ARSULOAD).

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(s) to retrieve. Also determines the name of the file that the ARSADMIN program creates in the directory specified with the **-d** parameter. Specify a valid six-part load ID. For example:

```
1220-1-0-8FAA-9339-9345
```

Where 1220 is the internal application group identifier, 1 is the primary storage node, 0 is the secondary storage node, 8FAA represents the name of the of the object that was stored during the load process, 9339 is the first (internal) date in the report, and 9345 is the last (internal) date in the report.

For the RES_DIFF function, the name of the resource group file(s) to process.

tableName
For the IMPORT function, the name of the migrated index table to import into the database.

-1 trace_file
Specifies the fully qualified trace file name.

-2 level
Specifies trace level.

1 Errors

2 Warnings

4 Info

8 Flow

Examples

1. The following example shows how to use the LOAD function to load the
   `credit.out` input file, the resources in the `credit.res` resource group file, and
   the index data in the `credit.ind` index file into the `Credit` application group.

   ```
   arsadmin load -g Credit -h instance -i credit.ind -s credit.out -r credit.res
   -u user -p password
   ```

2. The following example shows how to use the LOAD function to load a
   concatenated input file that contains the input data, index data, and resource
   group file.

   ```
   arsadmin load -g Credit -h instance -i credit.cat -s credit.cat -r credit.cat
   -u user -p password
   ```

3. The following example shows how to use the UNLOAD function to delete an
   input file from the system.

   ```
   arsadmin unload -g Credit -h instance -L loadID
   -u user -p password
   ```

4. 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
   ```

5. 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
   ```

6. 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 SL27 -h instance -u user -p password
   ```

7. 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
   ```

8. The following example shows how to use the MSGS function to display a
   message in the system log:

   ```
   arsadmin msgs -h instance -u user -p password -M 88 Name LoadId File
   ```

   in which

   Name First input string argument

   LoadId
         Second input string argument

   File   Third input string argument

   This message is displayed in the system log:

   ```
   08/15/2008 12:41:23 TRANG    1573048 Error No    88
   ApplGroup Failed Load: Name LoadId File
   ```

Notes

The OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) must be running, otherwise the ARSADMIN program will fail.

Files

/usr/lpp/ars/bin/arsadmin
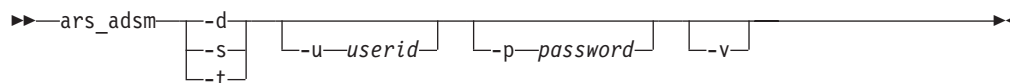> The AIX executable program.

/opt/ondemand/bin/arsadmin
> The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsadmin
> The Windows executable program.

ARS_ADSM

Syntax

```
►►──ars_adsm──┬──-d──┬────────────────┬──────────────────┬──────┬──►◄
              ├──-s──┤  └──-u──userid──┘  └──-p──password──┘  └──-v──┘
              └──-t──┘
```

Description

The ARS_ADSM program provides a command line interface to Tivoli Storage Manager for the following functions:

- Start the Tivoli Storage Manager server as a background process.
- Stop the Tivoli Storage Manager server.
- Create a backup image of the Tivoli Storage Manager database.

When you run Tivoli Storage Manager administrative tasks, you must log on to the Tivoli Storage Manager server using an administrative userid and password. A set of default values for the userid and password are in the ARS_ADSM program provided by IBM. You can use the values in the program or specify different values using the **-u** and **-p** command line parameters. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* describes how to change the Tivoli Storage Manager administrative userid and password in the ARS_ADSM program.

Parameters

-d   Create a backup image of the Tivoli Storage Manager database. Writes the backup image to the backup device that you configured when you installed and configured Tivoli Storage Manager. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides help with configuring the database backup device.

        If you plan to create a backup image of the database on tape, mount a tape storage volume that was formatted by Tivoli Storage Manager in the tape drive before you start the ARS_ADSM program.

-s   Start the Tivoli Storage Manager server as a background process.

-t   Stop the Tivoli Storage Manager server.

-u userid
        The userid of a Tivoli Storage Manager administrative user.

-p password
        The password for the Tivoli Storage Manager administrative userid specified with the **-u** parameter. If the user is not assigned a password, enter a null password. For example, **-p ""**.

-v   Enables verbose mode, to display all messages (informational and error). By default, the ARS_ADSM program displays error messages.

Examples

1. To start the Tivoli Storage Manager server, enter:

   ```
   ars_adsm -sv
   ```

2. To stop the Tivoli Storage Manager server, enter:

   ```
   ars_adsm -tv
   ```

3. To create a backup image of the Tivoli Storage Manager database, enter:

   ```
   ars_adsm -dv
   ```

   Note: If you plan to backup the database to tape, put a tape storage volume that was formatted by Tivoli Storage Manager in the drive before you start the ARS_ADSM program.

Notes

Before you use the ARS_ADSM program for the first time, you must configure it for your system. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring the ARS_ADSM program.

The ARS_ADSM program can be used to invoke the Tivoli Storage Manager processes that start and stop the Tivoli Storage Manager server and backup the Tivoli Storage Manager database. The ARS_ADSM program does not monitor the Tivoli Storage Manager processes or report on the success or failure of the Tivoli Storage Manager processes. **Important:** If you use the ARS_ADSM program to start the Tivoli Storage Manager process that backs up the Tivoli Storage Manager database, query the Tivoli Storage Manager server to determine whether the backup was successful before you continue with other OnDemand or Tivoli Storage Manager work. See your Tivoli Storage Manager documentation for details.

Content Manager OnDemand server programs can remain running while you create a backup copy of the Tivoli Storage Manager database. However, IBM recommends that you stop all OnDemand processes before you backup the database.

If you stop the Tivoli Storage Manager server while the Content Manager OnDemand processes are running, errors may occur. For example, if a client attempts to retrieve a document that resides on a storage volume that is managed by Tivoli Storage Manager, an error will occur and Content Manager OnDemand will send a message to the client and the system log.

Files

/usr/lpp/ars/bin/ars_adsm
       The AIX executable program.
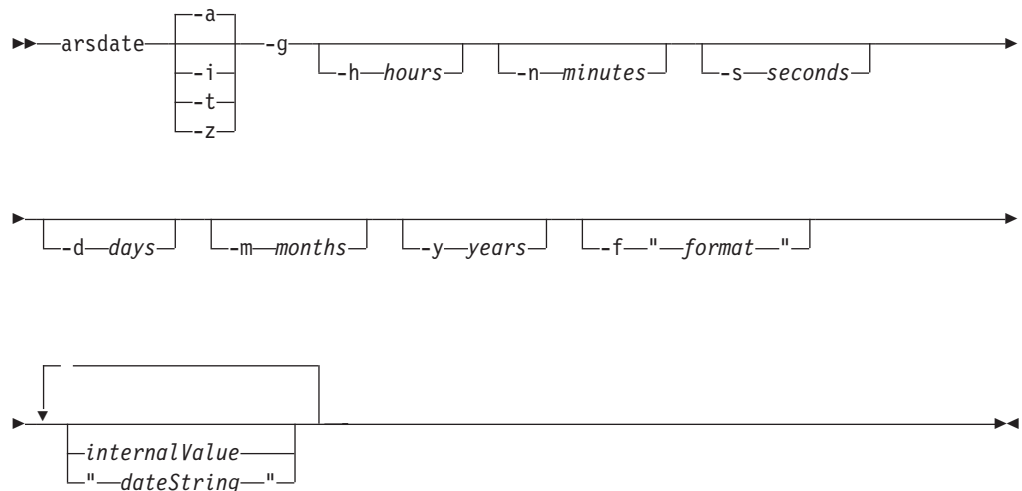
/opt/ondemand/bin/ars_adsm
       The HP-UX and Sun Solaris executable program.

ARSDATE

Purpose

Display the IBM Content Manager OnDemand internal database value for a given date and time string or display the date and time string for a given Content Manager OnDemand internal database value.

Syntax

```
>>--arsdate--+------+--  -g--+-------------+--+---------------+--+----------------+-->
             +- -a -+        +- -h--hours -+  +- -n--minutes -+  +- -s--seconds --+
             +- -i -+
             +- -t -+
             +- -z -+
```

```
>--+-----------+--+-------------+--+------------+--+- -f--"--format--"-+-->
   +- -d--days -+  +- -m--months-+  +- -y--years -+
```

```
>--+-----------------------+----------><
   |  +----------------+    |
   +--+- -internalValue-+---+
      +-"--dateString--"+
```

Description

The ARSDATE program can be used to display the IBM Content Manager OnDemand internal database value for a given date and time string or display the date and time string for a given Content Manager OnDemand internal database value.

Values whose data types are Date, Time, Date/Time, or Date/Time (TZ) are represented in an internal form that is transparent to the casual user of Content Manager OnDemand. Casual users enter date and time values the same way that they appear 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
```

-i   Use to display database values and date and time strings for Date/Time fields. The time part of a Date/Time field 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 a Date/Time database field? You can use the result to search a Date/Time field with an SQL string.
- Given a value stored in a Date/Time database field, what would be the date and time printed on a report?

-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
```

-z   Use to display database values and date and time strings for Date/Time (TZ) fields. The time part of a Date/Time (TZ) 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) 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) 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.

-g   Display the current system date.

-d days

Add the specified number of days to the specified internal value.

-h hours

Add the specified number of hours to the specified internal value.

-m months

Add the specified number of months to the specified internal value.

-n minutes

Add the specified number of minutes to the specified internal value.

-s seconds

Add the specified number of seconds to the specified internal value.

-y years

Add the specified number of years to the specified internal value.

-f "format"

Determines the format of the date and time string that Content Manager OnDemand displays.

Table 11 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 blank character).

Table 11. Date and time format specifications

Date Format Specifier	Date Format	Example
%m/%d/%y	mm/dd/yy	01/31/95
%d/%m/%y	dd/mm/yy	31/01/95
%f/%e/%y	m/d/yy	1/31/95
%e/%f/%y	d/m/yy	31/1/95
%m-%d-%y	mm-dd-yy	01-31-95
%d-%m-%y	dd-mm-yy	31-01-95
%m%d%y	mmddyy	013195
%m%d%Y	mmddyyyy	01311995
%y%m%d	yymmdd	950131
%Y%m%d	yyyymmdd	19950131
%b %e, %Y	Mth d, yyyy	Jan 31, 1995
%B %e, %Y	Month d, yyyy	January 31, 1995
%m/%d/%y %H:%M	mm/dd/yy hh:mm	01/31/95 10:50
%H:%M	hh:mm	10:50
%T	hh:mm:ss	10:50:59

internalValue

The internal date and time value from the Content Manager OnDemand database. Enter one or more internal values when you want to display formatted date strings.

"dateString"

The date and time string. Enter one or more strings when you want to display internal date values.

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**.

   ```
   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** 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**. 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). 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. 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
   ```

2. 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
```

3. The upper limit of the date is September 17, 2059. To format the output for a
   date, specify the **-f** parameter as follows:

```
arsdate -f "%m/%d/%Y" 32767
32767   ->    09/17/2059
```

Files

/usr/lpp/ars/bin/arsdate
   The AIX executable program.

/opt/ondemand/bin/arsdate
   The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsdate
   The Windows executable program.

ARSDB

Purpose

Provide an interface to the database manager for specific Content Manager OnDemand database functions, such as creating the database, starting and stopping the database manager, activating the database, creating a backup image of the database, and reorganizing the Content Manager OnDemand system tables. The Content Manager OnDemand database resides on the library server.

Note to Oracle users: See "Using the ARSDB program with Oracle" on page 244.

Syntax
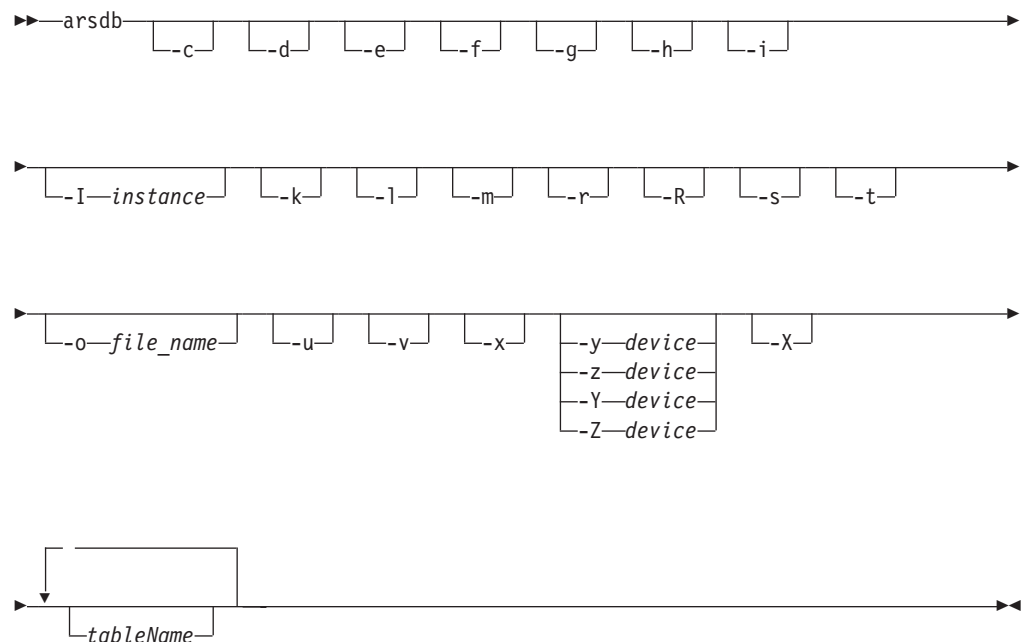
Description

The ARSDB program provides an interface to the database manager for the following database functions:

- Create and initialize the Content Manager OnDemand database.
- Start the database manager.
- Stop the database manager.
- Create backup images of the Content Manager OnDemand database. You can use the ARSDB program to create backup images of table spaces and the full database. When you run the ARSDB program to create backup images of DB2 table spaces and databases, you can specify that you want Tivoli Storage Manager to maintain the backup image files. However, to do so, you must first configure Tivoli Storage Manager. See *IBM Content Manager OnDemand for*

Multiplatforms: Installation and Configuration Guide for assistance with configuring Tivoli Storage Manager to maintain DB2 backup image files.

- Reorganize and optimize the Content Manager OnDemand system tables.
- Import and export tables and databases.

The ARSDB program reads the ARS.CFG file (UNIX servers) or the registry (Windows servers) to obtain the database configuration information. Before creating the database, you should verify the database configuration information. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* provides details.

Parameters

-c   For DB2 and SQL Server, create the (initial) Content Manager OnDemand database structure. (Not supported for Oracle.)

        The **-c** parameter performs the same functions as the **-r**, **-s**, and **-t** parameters.

-d   The arsdb -d command drops all OnDemand system tables from the database. To drop a specific table, use the `-d tableName` form of the command, where `tableName` is the name of the Content Manager OnDemand system table.

-e   The arsdb -e command will drop all the OnDemand system indexes for the specified Content Manager OnDemand System table (using `-e tableName`). The default, if no tableName is specified, is to drop all the system indexes for all OnDemand System tables.

-f   Ignore database manager failures.

-g   For DB2 and SQL Server, start the database manager. (Not supported for Oracle.)

-h   For DB2 and SQL Server, stop the database manager. (Not supported for Oracle.)

-i   Import data to the Content Manager OnDemand system 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 where the ARSDB program was invoked.

-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 (UNIX servers) or the registry (Windows servers) 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.

        Note: 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 Multiplatforms: Installation and Configuration Guide* for information about configuring instances.

-k   For DB2, activate the connection to the Content Manager OnDemand database. (Not supported for Oracle or SQL Server.)

-l   For DB2, import or export tables in the IXF file format instead of the DEL file format (the default). For SQL Server, import or export tables in the native file format. (Not supported for Oracle.)

The IXF file format is the PC version of the Integrated Exchange Format. A DEL file is a sequential ASCII file with row and column delimiters.

This flag is valid only with the **-i** and **-x** parameters.

-m   For DB2 and SQL Server, run maintenance on the Content Manager OnDemand database, reorganizing the Content Manager OnDemand system tables. (Not supported for Oracle.) This option refreshes the tables and optimizes access to information in the database. The system must be idle when you run the ARSDB program with this option. (Users should not be using the system and other applications should not be connected to the database.)

-r   Create indexes for the OnDemand system tables. 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 Content Manager OnDemand system table.

-R   Create the IBM Content Manager OnDemand Report Distribution for Multiplatforms Facility (RDF) tables and indexes.

-s   Run statistics on the Content Manager OnDemand system tables. This parameter is used 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 the Content Manager OnDemand system tables. The default is all Content Manager OnDemand system tables. To configure a specific table, use the `-s tableName` form of the command, where `tableName` is the name of the Content Manager OnDemand system table.

-o file_name
Specifies the file name for the log file that captures output from ARSDB commands.

-u   Updates the Content Manager OnDemand system tables with new changes every time when 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 data from the Content Manager OnDemand system 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.

-X   Exclude the USERSPACE1 table space from the backup image. The USERSPACE1 table space is created by DB2 when you initialize the OnDemand database. It contains system tables and application group

tables that have not been migrated to their own table spaces. This parameter is only valid with the **-Y** and **-Z** parameters.

A typical use of this parameter is to test the table space backup process. For example, assume that you migrated a table of application group data to its own table space. All other tables of application group data remains in the USERSPACE1 table space. To create an online backup image of the migrated application group table, use the `arsdb –XZ` form of the command.

-y device

For DB2, create an offline backup image of the Content Manager OnDemand database on the specified device. For SQL Server, create a complete database backup to the specified device. (Not supported for Oracle.)

Before you start this type of backup, you must make sure that no other users or applications are connected to the Content Manager OnDemand database. IBM recommends that you run this type of backup at a time when you are certain that no users will access the system. Otherwise, you should stop the Content Manager OnDemand server programs and other related processes before you start the backup.

If you specify a tape device, you must place a blank tape volume in the device before starting the ARSDB program.

If the specified device is `ADSM`, it means that the backup image file will be stored in storage that is managed by Tivoli Storage Manager. However, before you use the ARSDB program to store backup image files in Tivoli Storage Manager storage, you must configure Tivoli Storage Manager. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring Tivoli Storage Manager to maintain DB2 backup image files. The *DB2 Administration Guide* also contains information about Tivoli Storage Manager support and applicability for your system.

When backing up the database to devices that are managed by the operating system, the maximum number of backup devices that may be specified with the ARSDB program is 100. To specify backup devices, use the following format:

```
arsdb -y device1 [,device2, ... ,device100]
```

For example, the following shows how to initiate a backup by using three devices:

```
arsdb -y device1,device2,device3
```

Where:

- **-y** is the backup option.
- `device1,device2,device3` is a comma-separated list of devices to which the database manager will concurrently write the backup image of the database. Examples:

  ```
  /dev/rmt0,/dev/rmt1,/dev/rmt2
  ```

  ```
  \\.\Tape0,\\.\Tape1,\\.\Tape2
  ```

  ```
  /arsdb/backup1,/arsdb/backup2,/arsdb/backup3
  ```

DB2 provides support for backing up the Content Manager OnDemand database to Tivoli Storage Manager storage. The ARSDB program may be used to backup the database by using multiple connections to Tivoli

Storage Manager. Multiple connections to Tivoli Storage Manager can improve the performance of the backup task and reduce the time that it takes to complete the backup. Use the following format of the ARSDB program:

```
arsdb -y ADSM [,n]
```

For example, the following shows how to initiate a backup to Tivoli Storage Manager storage by using three connections to Tivoli Storage Manager:

```
arsdb -y ADSM,3
```

Where:

-y is the backup option.

ADSM specifies that the backup image file will be created in Tivoli Storage Manager storage.

3 is the number of connections that DB2 establishes with Tivoli Storage Manager. The number can be from 1 (one) to 10 (ten).

-Y device

For DB2, create an incremental, offline backup image of table spaces on the specified device. (Not supported for Oracle or SQL Server.) An incremental backup means that Content Manager OnDemand backs up only those tables that have changed since the last time they were backed up. Content Manager OnDemand keeps track of table space backups and can determine which table spaces have changed and need to be backed up.

Before you start an offline backup of the database, you must make sure that no other users or applications are connected to the Content Manager OnDemand database. IBM recommends that you run the offline backup at a time when you are certain that no users will access the system. Otherwise, you should stop the Content Manager OnDemand server programs and related processes before you start the offline backup.

If you specify a tape device, you must place a blank tape volume in the device before starting the ARSDB program.

If the specified device is ADSM, it means that the backup image file will be stored in storage that is managed by Tivoli Storage Manager. However, before you use the ARSDB program to store backup image files in Tivoli Storage Manager storage, you must configure Tivoli Storage Manager. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring Tivoli Storage Manager to maintain DB2 backup image files. The *DB2 Administration Guide* also contains information about Tivoli Storage Manager support and applicability for your system.

When backing up the database to devices that are managed by the operating system, the maximum number of backup devices that may be specified with the ARSDB program is 100. To specify backup devices, use the following format:

```
arsdb -Y device1 [,device2, ... ,device100]
```

For example, the following shows how to initiate a backup of the database by using three devices:

```
arsdb -Y device1,device2,device3
```

Where:

- **-Y** is the backup option.
- `device1,device2,device3` is a comma-separated list of devices to which the database manager will concurrently write the backup image of the database. Examples:

  `/dev/rmt0,/dev/rmt1,/dev/rmt2`

  `\\.\Tape0,\\.\Tape1,\\.\Tape2`

  `/arsdb/backup1,/arsdb/backup2,/arsdb/backup3`

DB2 provides support for backing up the Content Manager OnDemand database to Tivoli Storage Manager storage. The ARSDB program may be used to backup the database by using multiple connections to Tivoli Storage Manager. Multiple connections to Tivoli Storage Manager can improve the performance of the backup task and reduce the time that it takes to complete the backup. Use the following format of the ARSDB program:

`arsdb -Y ADSM [,n]`

For example, the following shows how to initiate a backup of the database to Tivoli Storage Manager storage by using three connections to Tivoli Storage Manager:

`arsdb -Y ADSM,3`

Where:

-Y is the backup option.

ADSM specifies that the backup image file will be created in Tivoli Storage Manager storage.

3 is the number of connections that DB2 establishes with Tivoli Storage Manager. The number can be from 1 (one) to 10 (ten).

-z device

For DB2, create an online backup image of the Content Manager OnDemand database on the specified device. For SQL Server, create a differential database backup to the specified device. (Not supported for Oracle.)

This type of backup may run while other users and applications are connected to the database. That is, other users and applications can read and modify the database while the backup is in process.

If you specify a tape device, then you must place a blank tape volume in the device before starting the ARSDB program.

If the specified device is ADSM, it means that the backup image file will be stored in storage that is managed by Tivoli Storage Manager. However, before you use the ARSDB program to store backup image files in Tivoli Storage Manager storage, you must configure Tivoli Storage Manager. See the *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring Tivoli Storage Manager to maintain DB2 backup image files. The *DB2 Administration Guide* also contains information about Tivoli Storage Manager support and applicability for your system.

When backing up the database to devices that are managed by the operating system, the maximum number of backup devices that may be specified with the ARSDB program is 100. To specify backup devices, use the following format:

```
arsdb -z device1 [,device2, ... ,device100]
```

For example, the following shows how to initiate an online backup of the
database by using three devices:

```
arsdb -z device1,device2,device3
```

Where:

- **-z** is the backup option.
- `device1,device2,device3` is a comma-separated list of devices to which
  the database manager will concurrently write the backup image of the
  database. Examples:

  ```
  /dev/rmt0,/dev/rmt1,/dev/rmt2
  ```

  ```
  \\.\Tape0,\\.\Tape1,\\.\Tape2
  ```

  ```
  /arsdb/backup1,/arsdb/backup2,/arsdb/backup3
  ```

DB2 provides support for backing up the Content Manager OnDemand
database to Tivoli Storage Manager storage. The ARSDB program may be
used to backup the database by using multiple connections to Tivoli
Storage Manager. Multiple connections to Tivoli Storage Manager can
improve the performance of the backup task and reduce the time that it
takes to complete the backup. Use the following format of the ARSDB
program:

```
arsdb - z ADSM [,n]
```

For example, the following shows how to initiate an online backup of the
database to Tivoli Storage Manager storage by using three connections to
Tivoli Storage Manager:

```
arsdb -z ADSM,3
```

Where:

> **-z** is the backup option.

> **ADSM** specifies that the backup image file will be created in Tivoli
> Storage Manager storage.

> 3 is the number of connections that DB2 establishes with Tivoli Storage
> Manager. The number can be from 1 (one) to 10 (ten).

-Z device

For DB2, create an incremental, online backup image of table spaces on the
specified device. (Not supported for Oracle or SQL Server.) An incremental
backup means that Content Manager OnDemand backs up only those
tables that have changed since the last time they were backed up. Content
Manager OnDemand keeps track of table space backups and can determine
which table spaces have changed and need to be backed up.

An online backup of the database can run while other users and
applications are connected to the database. That is, other users and
applications can read and modify the database while the backup process is
running.

If you specify a tape device, you must place a blank tape volume in the
device before starting the ARSDB program.

If the specified device is ADSM, it means that the backup image file will be
stored in storage that is managed by Tivoli Storage Manager. However,
before you use the ARSDB program to store backup image files in Tivoli

Storage Manager storage, you must configure Tivoli Storage Manager. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring Tivoli Storage Manager to maintain DB2 backup image files. The *DB2 Administration Guide* also contains information about Tivoli Storage Manager support and applicability for your system.

When backing up the database to devices that are managed by the operating system, the maximum number of backup devices that may be specified with the ARSDB program is 100. To specify backup devices, use the following format:

```
arsdb -Z device1 [,device2, ... ,device100]
```

For example, the following shows how to initiate a backup of the database by using three devices:

```
arsdb -Z device1,device2,device3
```

Where:
- **-Z** is the backup option.
- `device1,device2,device3` is a comma-separated list of devices to which the database manager will concurrently write the backup image of the database. Examples:

  ```
  /dev/rmt0,/dev/rmt1,/dev/rmt2
  ```

  ```
  \\.\Tape0,\\.\Tape1,\\.\Tape2
  ```

  ```
  /arsdb/backup1,/arsdb/backup2,/arsdb/backup3
  ```

DB2 provides support for backing up the Content Manager OnDemand database to Tivoli Storage Manager storage. The ARSDB program may be used to backup the database by using multiple connections to Tivoli Storage Manager. Multiple connections to Tivoli Storage Manager can improve the performance of the backup task and reduce the time that it takes to complete the backup. Use the following format of the ARSDB program:

```
arsdb -Z ADSM [,n]
```

For example, the following shows how to initiate an online backup of the database to Tivoli Storage Manager storage by using three connections to Tivoli Storage Manager:

```
arsdb -Z ADSM,3
```

Where:

  -Z is the backup option.

  ADSM specifies that the backup image file will be created in Tivoli Storage Manager storage.

  3 is the number of connections that DB2 establishes with Tivoli Storage Manager. The number can be from 1 (one) to 10 (ten).

Restriction:

If you want to restore a backup database image, and roll it forward to a certain point, do not use the `arsdb -z` command, because the archive logs, which are needed for the operation, are removed after the backup is completed. In this situation, a message `SQL4970N` is displayed. For example:

```
SQL4970N  Roll-forward recovery on database
"TEST2" cannot reach the specified stop point
(end-of-log or point-in-time) because
of missing log file(s) on node(s) "0".
```

It is recommended that you use the DB2 backup command. When you issue the DB2 command line to do backup, DB2 does not delete the archive logs.

tableName

Optionally specify the name of one or more Content Manager OnDemand system tables for the drop (**-d** and **-e**), import (**-i**), create configuration (**-r** and **-t**), run statistics (**-s**), and export (**-x**) parameters. If you do not specify the **tableName** parameter, then Content Manager OnDemand runs the command for all of the Content Manager OnDemand system tables.

For the import (**-i**) parameter, Content Manager OnDemand reads the imported tables from a file in the directory in which the ARSDB program was invoked. For the export (**-x**) parameter, Content Manager OnDemand writes the exported tables to a file in the directory in which the ARSDB program was invoked.

Examples

1. The following shows how to start the database manager:

   ```
   arsdb -gkv
   ```

2. The following shows how to stop the database manager:

   ```
   arsdb -hv
   ```

3. The following shows how to reorganize the Content Manager OnDemand system tables and run the database statistics command, which optimizes access to the tables:

   ```
   arsdb -msv
   ```

4. The following shows how to reorganize a specific Content Manager OnDemand system table. In the example, the ARSDB program reorganizes the indexes for the Application Group table:

   ```
   arsdb -mv arsag
   ```

5. The following shows how to export a specific Content Manager OnDemand system table to an IXF format file:

   ```
   arsdb -xl arsag
   ```

6. The following shows how to create an offline backup image of the Content Manager OnDemand database on tape:

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

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

   Note: You must place a blank, formatted tape in the device before you start the ARSDB program.

7. The following shows how to create an online, incremental backup of all table spaces in the Content Manager OnDemand database. The backup image is written to storage that is managed by Tivoli Storage Manager.

   ```
   arsdb -v -Z ADSM
   ```

8. The following shows how to create an offline, incremental backup of all table spaces in the Content Manager OnDemand database. The backup image is written to storage that is managed by Tivoli Storage Manager.

   ```
   arsdb -v -Y ADSM
   ```

9. The following shows how to create an online, full backup of the Content Manager OnDemand database. The backup image is written to storage that is managed by Tivoli Storage Manager.

   ```
   arsdb -v -z ADSM
   ```

10. The following shows how to create an offline, full backup of the Content Manager OnDemand database. The backup image is written to storage that is managed by Tivoli Storage Manager.

    ```
    arsdb -v -y ADSM
    ```

Notes

The ARSDB program should be run only on the library server.

IBM recommends that you stop the Content Manager OnDemand server programs and related processes before you run the ARSDB program with the create (**-c**), import (**-i**), reorganize (**-m**) or offline database backup (**-y**, **-Y**) parameters.

The Content Manager OnDemand server program (or service) can continue to run even after you stop the database manager. However, Content Manager OnDemand sends an error message to the client and the system log when a user attempts to query the database and the database manager has been stopped.

The online database backup command (**-z**, **-Z** parameters) can be run while other applications are connected to the database.

IBM recommends that you do not load data into the system at the same time that you run the ARSDB program with the **-m** or **-s** parameters.

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

Using the ARSDB program with Oracle

If you plan to use Oracle with Content Manager OnDemand, then you will need to use the standard Oracle procedures to start, stop, maintain, and optimize the database, to make backup copies of the database, and so forth. In addition, some of the other options for the ARSDB program do not apply when you use Oracle with Content Manager OnDemand. Table 12 on page 245 lists the options of the ARSDB program that are not valid when you use Oracle with Content Manager OnDemand.

Table 12. ARSDB program and Oracle

Option	Purpose	Notes
arsdb –g	Start the database	Use the Oracle procedures instead
arsdb –h	Stop the database	Use the Oracle procedures instead
arsdb –k	Activate the database	Does not apply to Oracle
arsdb –l	Import/Export tables	Does not apply to Oracle
arsdb –m	Reorganize the database	Does not apply to Oracle
arsdb –y	Backup the database	Use the Oracle procedures instead
arsdb –Y	Backup the database	Use the Oracle procedures instead
arsdb –z	Backup the database	Use the Oracle procedures instead
arsdb –Z	Backup the database	Use the Oracle procedures instead

You can use the ARSDB program to collect statistics on the Content Manager OnDemand system tables. The Content Manager OnDemand system tables include the user table, the group table, and the application group table. See "System control tables" on page 329 for a complete list of the Content Manager OnDemand system tables. For most customers, the Content Manager OnDemand system tables require very little maintenance. You can probably schedule the ARSDB program to collect statistics once a month (or less often).

The syntax is:

```
/opt/ondemand/bin/arsdb <options>
```

The options are:

–e   Drop configuration indexes

–r   Create configuration indexes

–s   Collect statistics

Files

/usr/lpp/ars/bin/arsdb
      The AIX executable program.

/opt/ondemand/bin/arsdb
      The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsdb
      The Windows executable program.

ARSDOC

Purpose

The ARSDOC program is a multi-purpose document processing program. You can use the ARSDOC program to query the database and generate a list of items that match a query, retrieve documents from the system, add, delete, and update documents, and send documents to a server printer.

Description and syntax

The ARSDOC program provides the following functions:

- ADD
- DELETE
- GET
- PRINT
- QUERY
- UPDATE
- HOLD_ADD
- HOLD_RELEASE
- CFSOD-FED

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.

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 date and 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 three ways:

- A document file
- An SQL query that contains clauses, database field names, index values, and operators
- A public named query

When you specify an SQL query or public named query, 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.
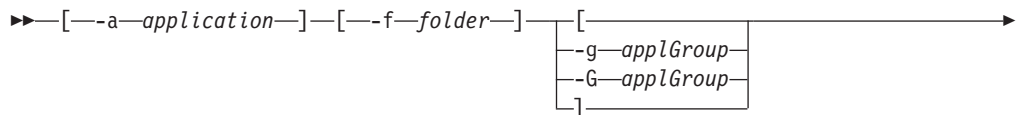
Important: The ADD function will fail unless the Database Organization for the application group named with the **-g** parameter is Multiple Loads per Database Table and the Expiration Type is Segment or Document. If only index data is added, and no document data is added, the Expiration Type can be `Segment` or `Document`. You can add index only data for an existing document when the expiration type is Load.
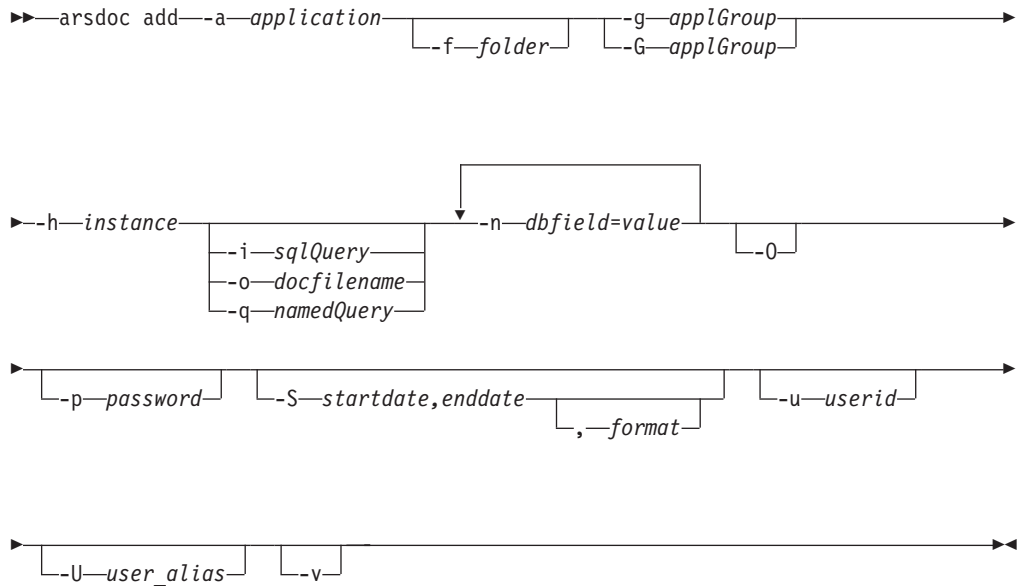
```
►►──arsdoc add──-a─application──────────────────────┬─-g─applGroup─┬─────────►
                              └─-f─folder─┘          └─-G─applGroup─┘
```

```
►──-h─instance─┬───────────────┬──►─┬──-n─dbfield=value─┬────────────►
               ├─-i─sqlQuery───┤                        └─-O─┘
               ├─-o─docfilename┤
               └─-q─namedQuery─┘
```

```
►─┬──────────────┬─┬─-S─startdate,enddate──────────────┬─┬──────────┬─►
  └─-p─password─┘  │               └─,─format─┘         │ └─-u─userid─┘
```

```
►─┬────────────────┬─┬────┬──────────────────────────────────────►◄
  └─-U─user_alias─┘  └─-v─┘
```

When you use the parameter file option, you must specify the **-F**, **-h**, **-p**, **-u**, **-U**, and **-v** 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:
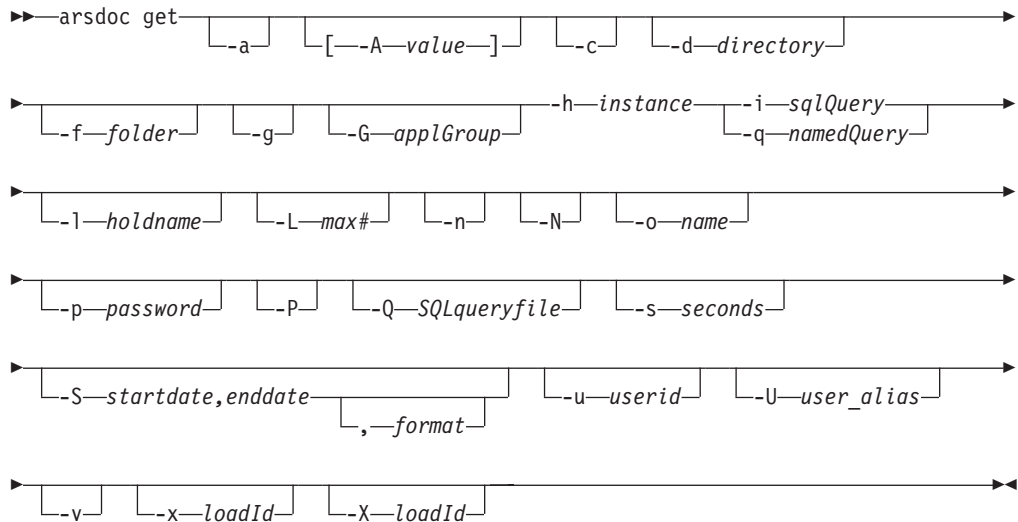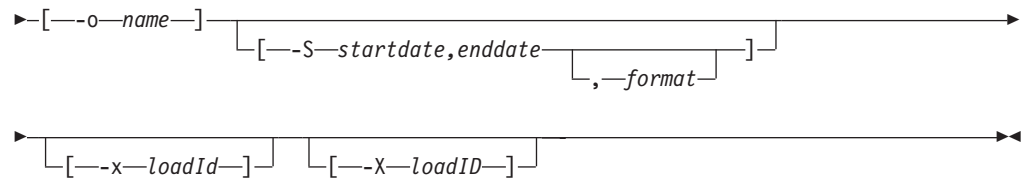
```
►►──arsdoc add──-F─parmfile──-h─instance─┬────────────┬─┬──────────┬─►
                                         └─-p─password─┘ └─-u─userid─┘
```

```
►─┬────┬────────────────────────────────────────────────────────►◄
  └─-v─┘
```
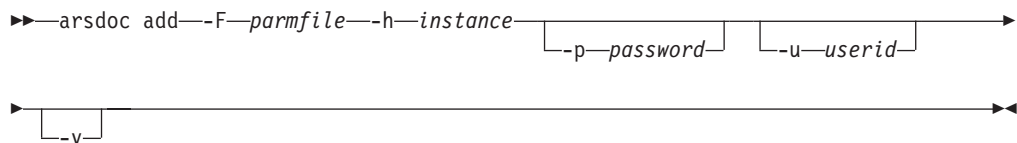
The syntax of the parameter file for the ADD function is:

```
►►──[──-a─application─]──[──-f─folder─]──┬─[────────────────┬────────►
                                         │  ├─-g─applGroup─┤
                                         │  ├─-G─applGroup─┤
                                         │  └─]
```

```
►►──┬─[──-i─sqlQuery──]─┬──────┬─[──-n─dbfield=value──]─┬──────┬─[──-O──]─┬─►
    ├─[──-o─docfilename──]─┤                            │      └──────────┘
    └─[──-q─namedQuery──]──┘

►──┬─[──-S─startdate,enddate──────────────]─┬─►◄
   │                  └──,──format──┘         │
```
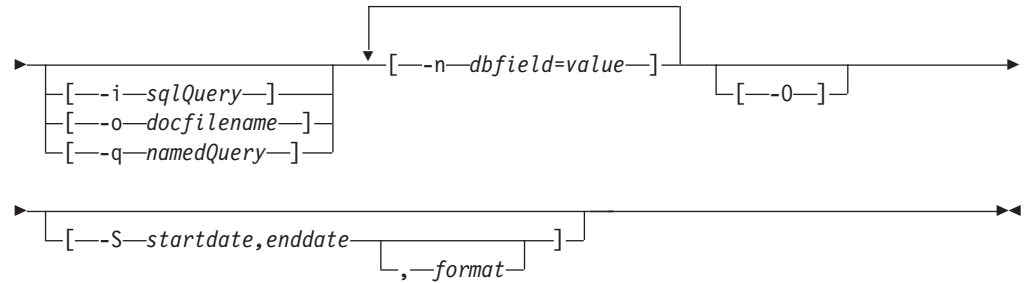
DELETE function

Use to delete documents from the system.

To identify the documents that you want to delete , you must enter an SQL query or specify the name of a public named query. The SQL query must contain clauses, database field names, index values, and operators. The DELETE function deletes all documents that match the query.
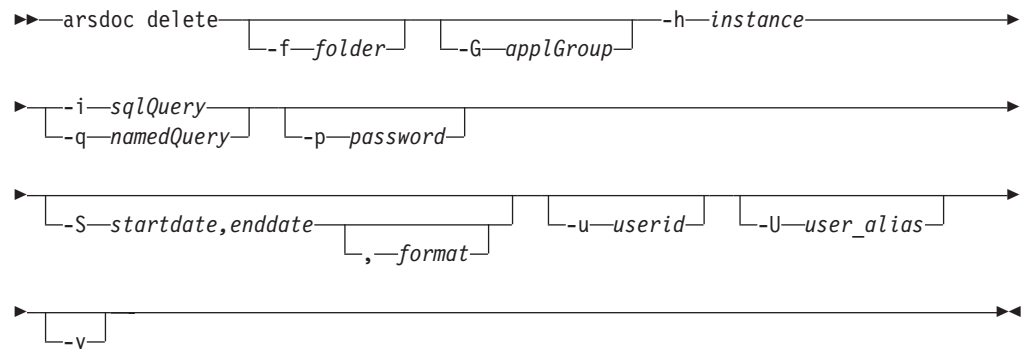
```
►►──arsdoc delete──┬────────┬──┬──────────┬──-h─instance──────────►
                   └─-f─folder─┘  └─-G─applGroup─┘

►──┬─-i─sqlQuery──────┬──┬──────────────┬────────────────────►
   └─-q─namedQuery────┘  └─-p─password──┘

►──┬────────────────────────────┬──┬──────────┬──┬─────────────┬─►
   └─-S─startdate,enddate────────┘  └─-u─userid─┘  └─-U─user_alias─┘
            └──,──format──┘

►──┬──────┬──────────────────────────────────────────►◄
   └─-v──┘
```

When you use the parameter file option, you must specify the **-F**, **-h**, **-p**, **-u**, **-U**, and **-v** 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:
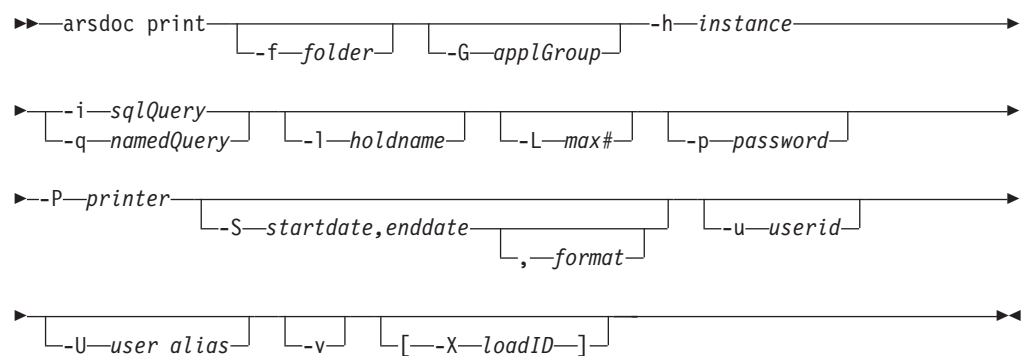
```
►►──arsdoc delete──-F─parmfile──-h─instance──┬──────────────┬──►
                                             └─-p─password──┘

►──┬──────────┬──┬──────┬──────────────────────────────────►◄
   └─-u─userid─┘  └─-v──┘
```
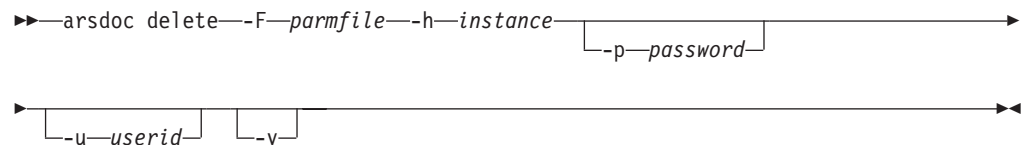
The syntax of the parameter file for the DELETE function is:

```
►►──┬─[──-f─folder──]─┬──────────────────────┬──┬─[──-i─sqlQuery──]─┬────►
                      └─[──-G─applGroup──]────┘  └─[──-q─namedQuery──]─┘

►──┬─[──-S─startdate,enddate──────────────]─┬──────────────────►◄
   │                  └──,──format──┘         │
```
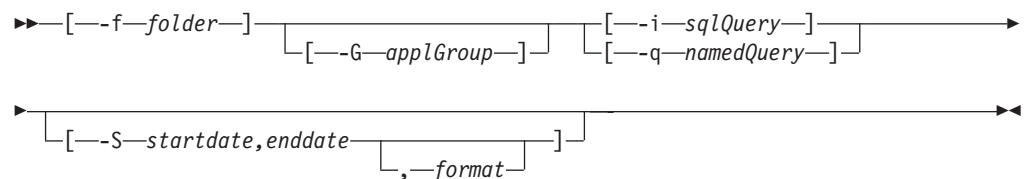
GET function

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. To query the database, you can enter an SQL query or specify the name of a public named query. The SQL query 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 (UNIX servers) or the console (Windows servers). You can also choose to write the output to a file. To retrieve documents in a sorted order, you must specify the **-n** parameter.

```
►►──arsdoc get────────────────────────────────────────────────────────────────►
                    └─-a─┘   └─[──-A─value─]─┘   └─-c─┘   └─-d─directory─┘

►───────────────────────────────────────────────────-h─instance───┬─-i─sqlQuery──┬──►
    └─-f─folder─┘  └─-g─┘  └─-G─applGroup─┘                        └─-q─namedQuery─┘

►─────────────────────────────────────────────────────────────────────────────►
    └─-l─holdname─┘  └─-L─max#─┘  └─-n─┘  └─-N─┘  └─-o─name─┘

►─────────────────────────────────────────────────────────────────────────────►
    └─-p─password─┘  └─-P─┘  └─-Q─SQLqueryfile─┘  └─-s─seconds─┘

►─────────────────────────────────────────────────────────────────────────────►
    └─-S─startdate,enddate──────────┘  └─-u─userid─┘  └─-U─user_alias─┘
                       └─,─format─┘

►─────────────────────────────────────────────────────────────────────────────►◄
    └─-v─┘  └─-x─loadId─┘  └─-X─loadId─┘
```
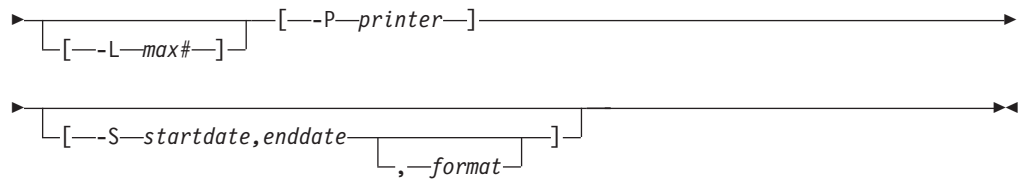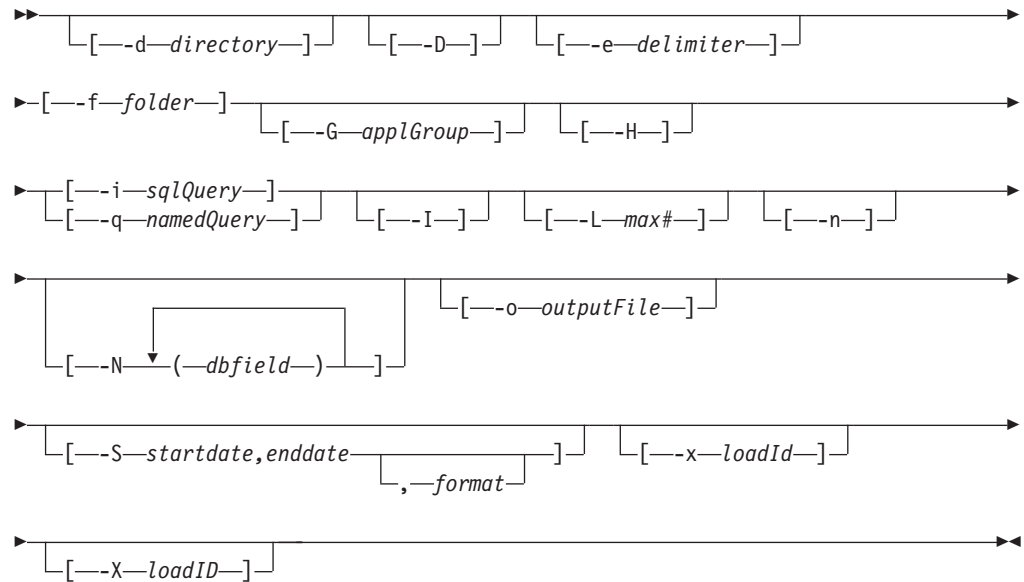
When you use the parameter file option, you must specify the **-F**, **-h**, **-p**, **-u**, **-U**, and **-v** parameters on the command line. All other parameters must be specified in the parameter file. The following diagram shows the syntax of the GET function when you use the parameter file option:
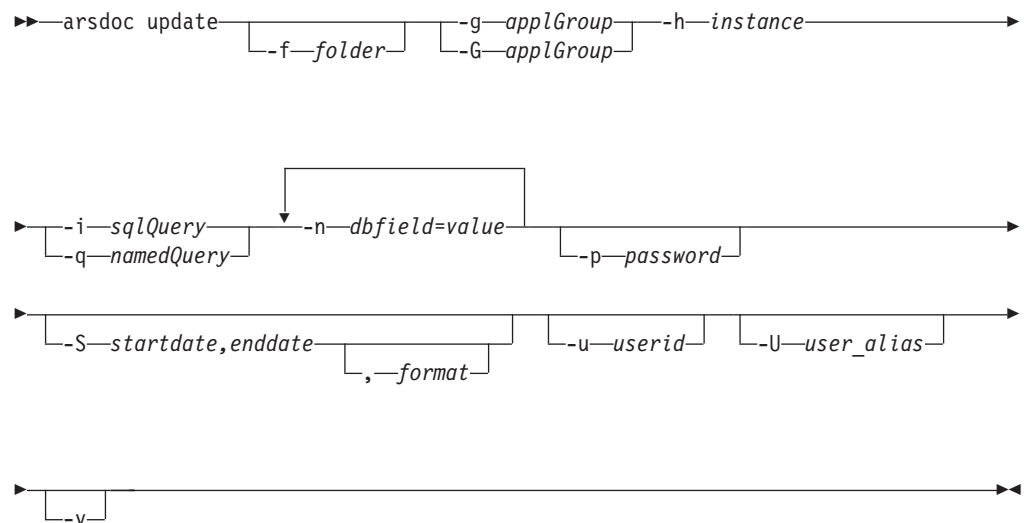
```
►►──arsdoc get──-F─parm_file──-h─instance────────────────────────────────────────►
                                              └─-p─password─┘  └─-s─seconds─┘

►─────────────────────────────────────────────────────────────────────────────►◄
    └─-u─userid─┘  └─-v─┘
```
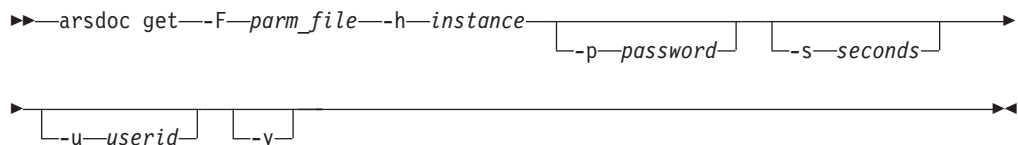
The syntax of the parameter file for the GET function is:

```
►►─────────────────────────────────────────────────────────────────────────────►
    └─[──-a─]─┘  └─[──-A─value─]─┘  └─[──-c─]─┘  └─[──-d─directory─]─┘

►─[──-f─folder─]──────────────────────────────────────────────────────────────►
                 └─[──-g─]─┘  └─[──-G─applGroup─]─┘

►─┬─[──-i─sqlQuery──]──┬───────────────────────────────────────────────────────►
  └─[──-q─namedQuery─]─┘  └─[──-L─max#─]─┘  └─[──-n─]─┘  └─[──-N─]─┘
```

```
►─[──-o─name──]──────────────────────────────────────────────────────────►
                    └─[──-S─startdate,enddate─────────────────]─┘
                                          └─,─format─┘

►────────────────────────────────────────────────────────────────────────►◄
      └─[──-x─loadId──]─┘  └─[──-X─loadID──]─┘
```
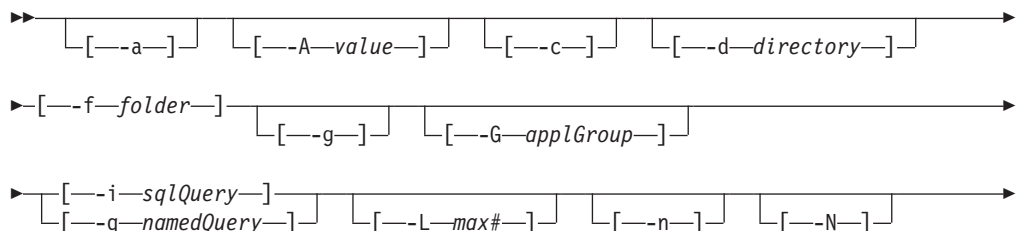
PRINT function

Use to send documents to an IBM Content Manager OnDemand server printer. You must name the Content Manager OnDemand library server and the folder that you want to search and specify the query to run. The items that match the query are sent to the server printer named with the **-P** parameter. The server printer must be defined to Content Manager OnDemand using the administrative client. To query the database, you can enter an SQL query or specify the name of a public named query. The SQL query 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. You can limit the number of database tables searched, and possibly increase the performance of a query, by specifying the **-S** parameter and specifying a start date and an end date. The PRINT function does not currently support server FAX.

```
►►──arsdoc print────────────────────────────────────── -h─instance─────────►
                  └─-f─folder─┘  └─-G─applGroup─┘

►──┬─-i─sqlQuery──┬───────────────────────────────────────────────────────►
   └─-q─namedQuery─┘  └─-l─holdname─┘  └─-L─max#─┘  └─-p─password─┘

►──-P─printer─────────────────────────────────────────────────────────────►
             └─-S─startdate,enddate──────────────┘  └─-u─userid─┘
                                 └─,─format─┘

►─────────────────────────────────────────────────────────────────────────►◄
   └─-U─user_alias─┘  └─-v─┘  └─[──-X─loadID──]─┘
```

When you use the parameter file option, you must specify the **-F**, **-h**, **-p**, **-u**, **-U**, and **-v** 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:
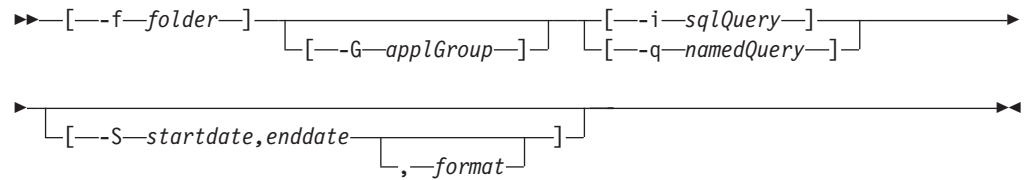
```
►►──arsdoc print──-F─parmfile──-h─instance───────────────────────────────────►
                                           └─-p─password─┘  └─-u─userid─┘

►─────────────────────────────────────────────────────────────────────────►◄
   └─-v─┘
```
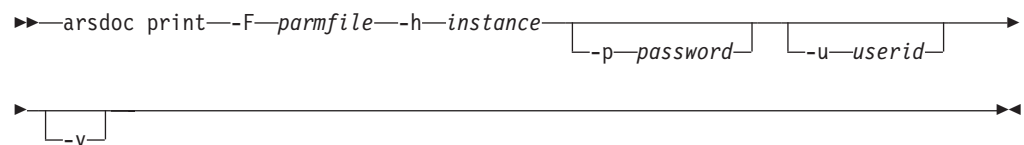
The syntax of the parameter file for the PRINT function is:

```
►►──[──-f─folder──]──────────────────────┬─[──-i─sqlQuery──]──┬──────────────►
                   └─[──-G─applGroup──]─┘  └─[──-q─namedQuery──]─┘
```

```
►──────┬────────────┬───┬──[──-P─printer──]──────────────────────────────►
       └[──-L─max#──]┘

►──────────────────────────────────────────────────────────────────────►◄
       └[──-S─startdate,enddate──┬──────────────┬──]─┘
                                 └,──format─┘
```
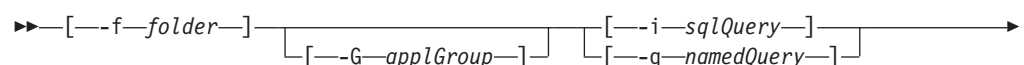
QUERY function

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. To query the database, you can enter an
SQL query or specify the name of a public named query. The SQL query 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.

By default, the ARSDOC program sends the list of items that match the SQL query
to stdout (UNIX servers) or the console (Windows servers). You typically redirect
the output to a printer or another process. You can also choose to write the output
to a file.
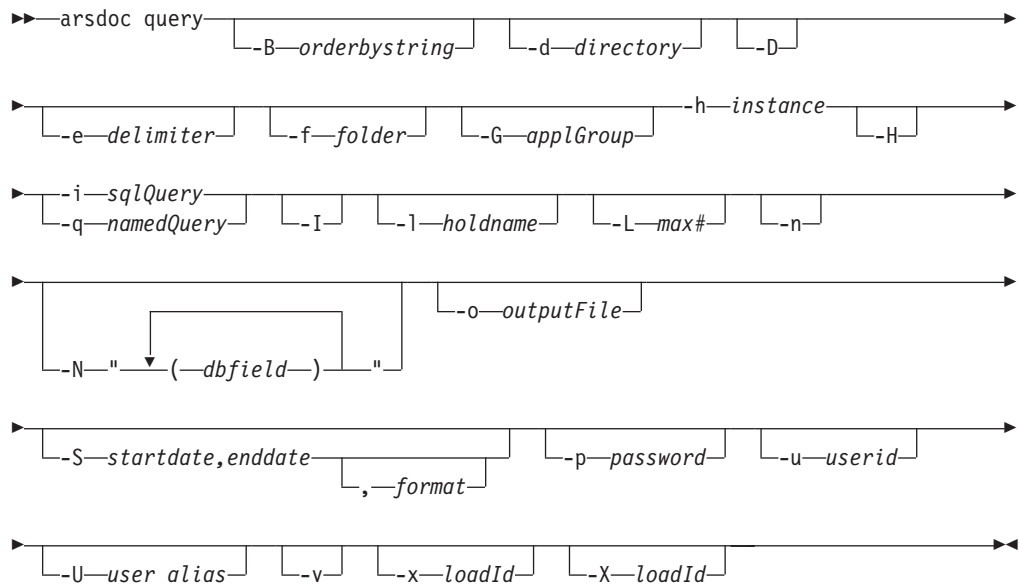
```
►►──arsdoc query──┬──────────────────┬──┬──────────────┬──┬────┬──►
                  └-B─orderbystring──┘  └-d─directory──┘  └-D──┘

►──┬───────────────┬──┬──────────┬──┬───────────────┬──-h─instance──┬─────┬──►
   └-e─delimiter───┘  └-f─folder─┘  └-G─applGroup───┘               └-H──┘

►──┬──-i─sqlQuery─────┬──┬────┬──┬──────────────┬──┬──-L─max#──┬──┬────┬──►
   └──-q─namedQuery───┘  └-I──┘  └-l─holdname───┘  └──────────┘  └-n──┘

►──┬──────────────────────────────┬──┬───────────────┬──────────────────────►
   │         ┌◄───────────┐       │  └-o─outputFile──┘
   └──-N─"──┴─(─dbfield─)─┴──"─────┘

►──┬──────────────────────────────────┬──┬──────────────┬──┬───────────┬──►
   └-S─startdate,enddate─┬──────────┬──┘  └-p─password───┘  └-u─userid──┘
                         └,──format─┘

►──┬──────────────────┬──┬────┬──┬──────────────┬──┬──────────────┬──►◄
   └-U─user_alias─────┘  └-v──┘  └-x─loadId─────┘  └-X─loadId──────┘
```
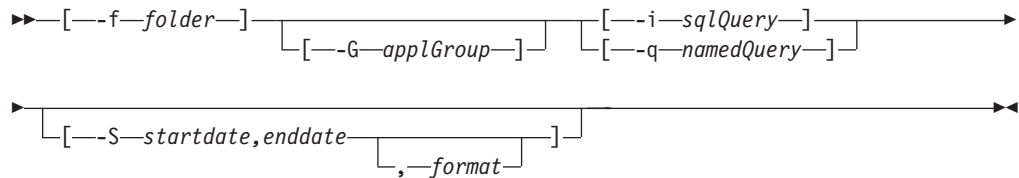
When you use the parameter file option, you must specify the **-F**, **-h**, **-p**, **-u**, **-U**,
and **-v** parameters on the command line. All other parameters must be specified in
the parameter file. The following shows the syntax of the QUERY function when
you use the parameter file option:

```
►►──arsdoc query──-F─parm_file──-h─instance──┬──────────────┬──►
                                             └-p─password───┘

►──┬──────────────┬──┬────┬──────────────────────────────────►◄
   └-u─userid─────┘  └-v──┘
```
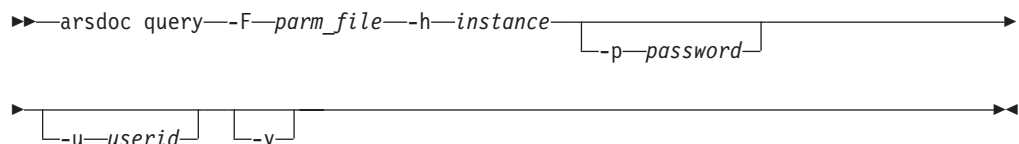
The syntax of the parameter file for the QUERY function is:

```
►►─┬─────────────────────┬──┬─────────┬──┬──────────────────────┬────────►
   └─[──-d─directory──]──┘  └─[──-D──]─┘  └─[──-e─delimiter──]──┘

►─┬─[──-f─folder──]──┬─────────────────────────────────────────────────────►
                     └─[─-G─applGroup──]──┘  └─[──-H──]──┘

►─┬─[──-i─sqlQuery──]────┬──┬─────────┬──┬───────────┬──┬─────────┬─────────►
  └─[──-q─namedQuery──]──┘  └─[──-I──]─┘  └─[──-L─max#──]─┘  └─[──-n──]──┘

►─┬──────────────────────────────┬──┬──────────────────────┬──────────────►
  │            ◄─────────┐       │  └─[──-o─outputFile──]──┘
  └─[──-N──┬─(──dbfield──)─┬──]──┘

►─┬──────────────────────────────────────┬──┬────────────────┬────────────►
  └─[─-S─startdate,enddate──┬──────────┬──]─┘  └─[──-x─loadId──]─┘
                            └─,─format─┘

►─┬─────────────────────┬──────────────────────────────────────────────────►◄
  └─[──-X─loadID──]──┘
```

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 must enter an SQL query or specify the name of a public named query. The SQL query must contain clauses, database field names, index values, and operators. The UPDATE function updates all of the rows that match the query.

```
►►──arsdoc update──┬────────────┬──┬─-g─applGroup─┬──-h─instance──────────►
                   └─-f─folder──┘  └─-G─applGroup─┘
```

```
         ◄───────────────────────┐
►─┬─-i─sqlQuery────┬──┬─-n─dbfield=value──┬─────────────────────────────────►
  └─-q─namedQuery──┘                      └─-p─password──┘
```

```
►─┬───────────────────────────────────────┬──┬─────────────┬──┬──────────────────┬─►
  └─-S─startdate,enddate──┬──────────┬─────┘  └─-u─userid───┘  └─-U─user_alias──┘
                          └─,─format─┘
```

```
►─┬──────┬──────────────────────────────────────────────────────────────────►◄
  └─-v──┘
```

When you use the parameter file option, you must specify the **-F**, **-h**, **-p**, **-u**, **-U**, and **-v** 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:

```
►►──arsdoc update──-F──parm_file──-h──instance──────────────────────────────►
                                              └─-p──password─┘

►─────────────────────────────────────────────────────────────────────────►◄
   └─-u──userid─┘  └─-v─┘
```
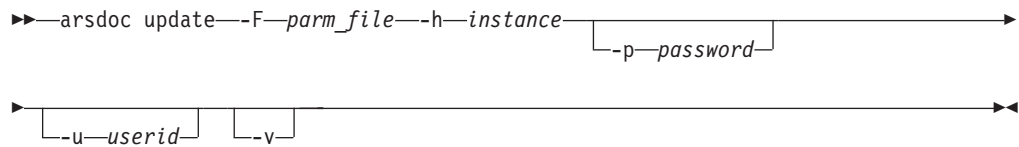
The syntax of the parameter file for the UPDATE function is:

```
►►──[──-f──folder──]──┬─────────────────┬──┬─[──-i──sqlQuery──]──┬─────────►
                      │   ┌─-g──applGroup─┐ │  └─[──-q──namedQuery──]─┘
                      └─[─┼─-G──applGroup─┼─┘
                          └─]─────────────┘
```

```
        ┌──────────────────────────┐
        ▼                          │
►─────[──-n──dbfield=value──]──────┴────────────────────────────────────►◄
                               └─[──-S──startdate,enddate────────────────]─┘
                                                      └─,──format─┘
```
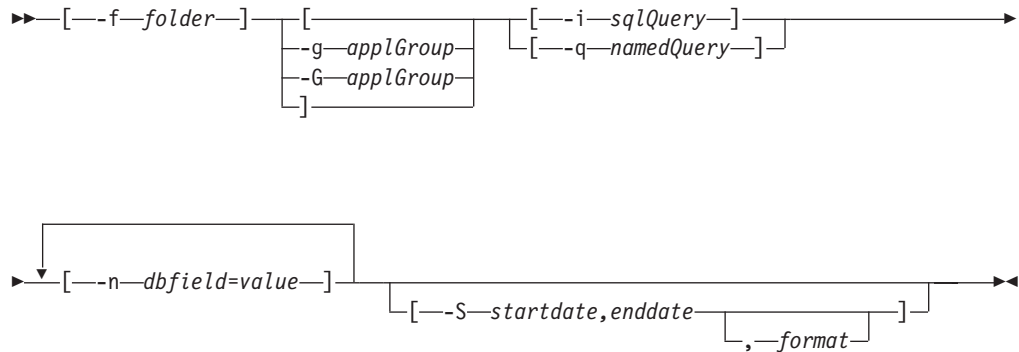
HOLD_ADD function

Use to add documents to a hold. Use an SQL string or Named Query to query the database and determine the documents to add to the hold.

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.
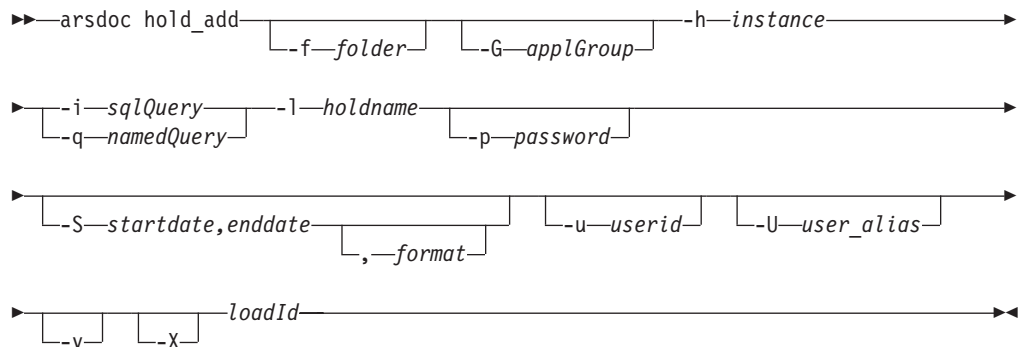
```
►►──arsdoc hold_add──┬──────────────┬──┬──────────────────┬──-h──instance──►
                     └─-f──folder─┘    └─-G──applGroup─┘

►──┬─-i──sqlQuery───┬──-l──holdname──────────────────────────────────────►
   └─-q──namedQuery─┘             └─-p──password─┘

►──┬──────────────────────────────────────┬──┬────────────┬──┬───────────────────┬──►
   └─-S──startdate,enddate────────────────┘  └─-u──userid─┘  └─-U──user_alias─┘
                         └─,──format─┘

►──┬──────────────┬──loadId──────────────────────────────────────────────►◄
   └─-v─┘  └─-X─┘
```

When you use the parameter file option, you must specify the **-f**, **-h**, **-p**, **-u**, **-U**, and **-v** 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:

```
►►──arsdoc hold_add──-F──parmfile──-h──instance──────────────────────────────►
                                            └─-p──password─┘
```

```
        ┌──-u──userid──┐  ┌──-v──┐ ┌──-X──┐ ──loadId──────────────────────────►◄
        └──────────────┘  └──────┘ └──────┘
```
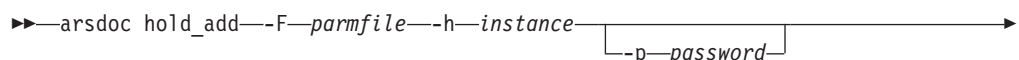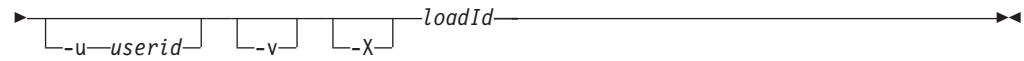
The syntax of the parameter file for the hold_add function is:

```
►►──[──-f──folder──]──────────────────────[──-i──sqlQuery──]────────────────────►
                    └──[──-G──applGroup──]─┘ [──-q──namedQuery──]─┘
```

```
►──────────────────────────────────────────────────────────────────────────────►◄
  └──[──-S──startdate,enddate─────────────────]─┘
                              └──,──format──┘
```

HOLD_RELEASE function

Use to remove documents from a hold. You can query the database by using an SQL string or Named Query to determine the documents that should be removed from a hold.

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.
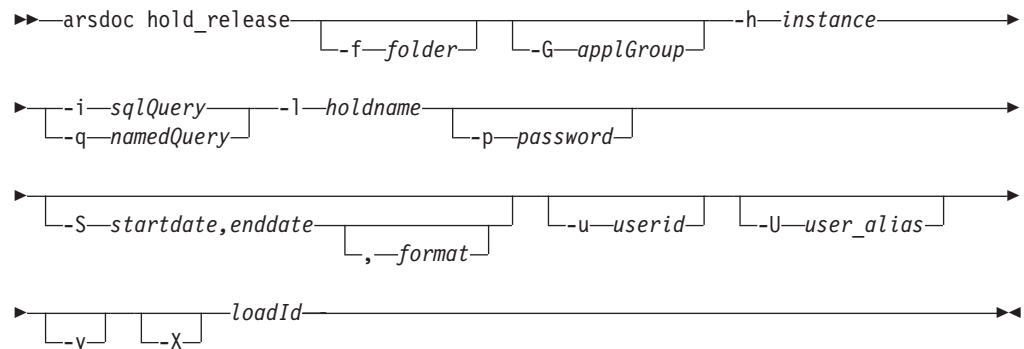
```
►►──arsdoc hold_release──────────────────────────────────────────-h──instance────►
                        └─-f──folder─┘ └─-G──applGroup─┘
```

```
►──┬─-i──sqlQuery────┬──-l──holdname──────────────────────────────────────────────►
   └─-q──namedQuery──┘                └─-p──password─┘
```

```
►──────────────────────────────────────────────────────────────────────────────►
  └─-S──startdate,enddate──────────────┘ └─-u──userid─┘ └─-U──user_alias─┘
                         └─,──format──┘
```

```
►──┌──-v──┐ ┌──-X──┐ ──loadId────────────────────────────────────────────────────►◄
   └──────┘ └──────┘
```

When you use the parameter file option, you must specify the **-f**, **-h**, **-p**, **-u**, **-U**, and **-v** 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:

```
►►──arsdoc hold_release──-F──parmfile──-h──instance──────────────────────────────►
                                                    └─-p──password─┘
```

```
┃          ┌─-u─userid─┐ ┌─-v─┐ ┌─-X─┐ loadId                              ►◄
```
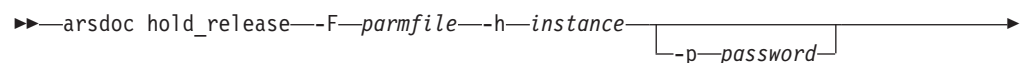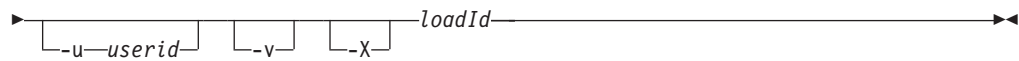
The syntax of the parameter file for the hold_release function is:

```
►►──[──-f─folder──]──────────────────────[──-i─sqlQuery──]──────────────────►
              └─[──-G─applGroup──]─┘       └─[──-q─namedQuery──]─┘

►──────────────────────────────────────────────────────────────────────────►◄
    └─[──-S─startdate,enddate──────────────]─┘
                          └─,─format─┘
```

CFSOD-FED function

Sends documents to CFS-OD and make them available to IBM FileNet® P8 features.
You can use an SQL string or Named Query to determine the documents to be sent
to CFS-OD.

To enable the CFS-OD feature, add the following line to the `ars.cfg` file:
`ARS_SUPPORT_CFSOD=1`

Ensure that the CFS-OD function is turned on for the application group and you
have permission to use CFS-OD before you attempt this function. Otherwise, a
permission error will occur.

If a duplicate attempt is made to send the same documents to CFS-OD, 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-OD.
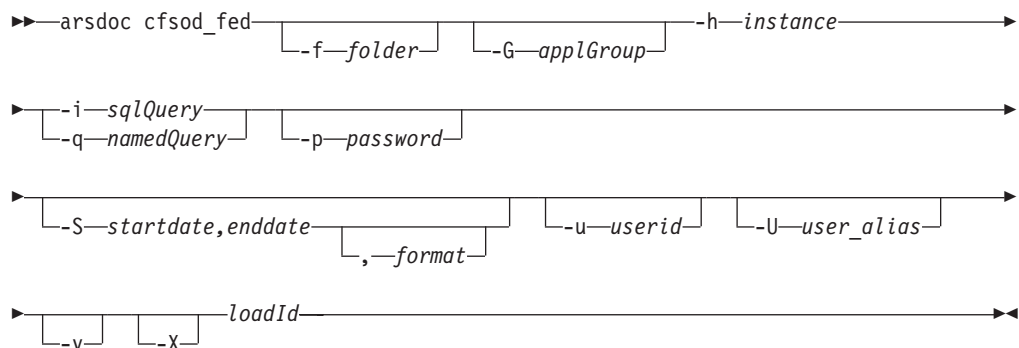
```
►►──arsdoc cfsod_fed─────────────────────────────────────-h─instance───────►
                     └─-f─folder─┘ └─-G─applGroup─┘

►──-i─sqlQuery──────────────────────────────────────────────────────────────►
   └─-q─namedQuery─┘ └─-p─password─┘

►───────────────────────────────────────────────────────────────────────────►
    └─-S─startdate,enddate──────────┘  └─-u─userid─┘  └─-U─user_alias─┘
                      └─,─format─┘

►──────────────────loadId───────────────────────────────────────────────────►◄
    └─-v─┘ └─-X─┘
```

When you use the parameter file option, you must specify the **-f**, **-h**, **-p**, **-u**, **-U**, and
-v 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:

```
►►──arsdoc cfsod_fed──-F─parmfile──-h─instance──────────────────────────────►
                                                └─-p─password─┘

►──────────────────loadId───────────────────────────────────────────────────►◄
    └─-u─userid─┘ └─-v─┘ └─-X─┘
```
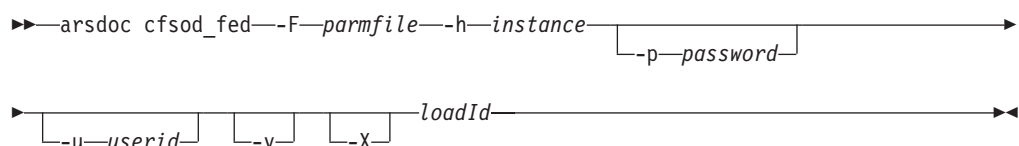
The syntax of the parameter file for the cfsod_fed function is:

```
►►──[──-f──folder──]──────────────────────────────────[──-i──sqlQuery──]────────────►
           └[──-G──applGroup──]┘   └[──-q──namedQuery──]┘

►──────────────────────────────────────────────────────────────────◄
  └[──-S──startdate,enddate────────────────]┘
                      └,──format──┘
```
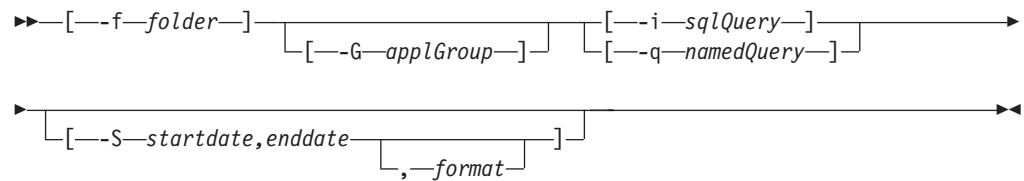
Parameters

-a
  For the GET function, when retrieving AFP documents, specify this parameter to include resources with the documents that are retrieved. If documents from the same application have different resource groups, 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.

-A *value*
  Use this parameter to retrieve annotations. 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 13. 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

  See "Examples" on page 268 for examples of using the **-A** parameter with the ARSDOC GET function.

-B orderbystring

For the QUERY function, use this parameter to permit SQL to reorder your table rows for the purpose of viewing.

-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.

For UNIX servers, directory names are case sensitive; for Windows servers, directory names are not case sensitive.

-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.

Ensure that you enclose the parameters and values that are specified in the parameter file in the left and right brackets, that is, [and]. The left and right brackets are used to identify each parameter in the file, and are required in the parameter file.

Important: In the parameter file, the parameter values cannot contain the left or right bracket.

Here is an example of using the ARSDOC program with the **-F** parameter:

```
arsdoc get -u oduser -p odpasswd -h odhost -F parmfile -v
```

Here is an example of the parameter values in the above parameter file:

```
[-f "Credit Card Statements"][-i "where account = '000-000-000'"]
[-a][-o credit.out]
```

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 for Multiplatforms: 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.

The ARSDOC program will attempt to locate the specified instance name in the ARS.INI file (UNIX servers; the registry on Windows servers) 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 ARSDOC program cannot locate the instance name in the ARS.INI file, the specified value is treated as a host name.

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 Multiplatforms: Installation and 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.)

The ARSDOC program writes the database field names to the specified output file or to stdout (UNIX servers) or the console (Windows servers), in the format used for the header record.

-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.

Restriction: If you specify the **-q** or the **-X** parameters, then you cannot specify the **-i** parameter.

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, you must use the Content Manager OnDemand internal format of the date. That is, the number of days since January 1, 1970. You can use the ARSDATE program to list the internal format for a given date string. For example, the following shows how to use the ARSDATE program on an AIX server to obtain the internal date for July 21, 1995:

```
/usr/lpp/ars/bin/arsdate -a 7/21/95
```

The ARSDATE program displays:

```
7/21/95 -> 9333
```

You would then enter 9333 as the index value for the date field.

-I

Important: When you use this parameter, you must specify the f or p variable. 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. You cannot specify the **-H**, **-n**, or **-N** parameters when you specify the **-I** parameter.

To use the **-I** parameter, the user running the query must have permission to access the System Log application group and folder.

If the Load ID is not found in the system log, 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

For the GET, QUERY, PRINT, HOLD_ADD, and HOLD_RELEASE functions, this parameter specifies the hold name. Do not specify this parameter when you specify the **-X** parameter. Do not specify this parameter if only database names are specified. 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 10 hits. If 2 of the 10 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 program uses a bulk retrieval method for high-speed retrieval of items from the server.

Tip: Specify the **-n** parameter 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. 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 QUERY function, include the line numbers.

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:

- If there is not a unique database value for the specified query, you can specify **.SEQ** after the database field names to ensure that a unique numeric extension is added to each of the file names that are generated. **.SEQ** is case sensitive. For example, if the account number is 000-000-000 and the name is JONES, after you use this command:

```
arsdoc get -u oduser -p odpasswd -h odserver -i
"where account = '000-000-000'" -f "Credit Card Statements"
-o (account)(name).SEQ -v
```

The following files will be created: `000-000-000.JONES.1`, `000-000-000.JONES.2, ... 000-000-000.JONES.n`.

- 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.

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 may never omit date or time fields. 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.

-p password
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 double quotes to show a null password. That is, specify `-p ""`. If you omit the **-p** parameter, then the ARSDOC program prompts you to enter the password when you run the program. If there is no password assigned to the user that you specify, press the **Enter** key when prompted.

-P   Indicates PDF files that are retrieved and should be stored in individual files even if concatenation has been requested.

-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.

Restriction: If you specify the **-i** or the **-X** parameters, then you cannot specify the **-q** parameter.

-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**.

-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 "ARSDATE" on page 229 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.

-u userid

The Content Manager OnDemand user that is permitted to perform the specified function. The ARSDOC program verifies that the user ID that you specify is a valid Content Manager OnDemand user for the library server that you name with the **-h** parameter, that the user ID is permitted to open the folder that you name with the **-f** parameter, and that 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.

-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 function, use to limit the documents that can be retrieved to the set of documents that were loaded into the system under the specified *loadId*.

For the QUERY function, 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. You can specify the search using the **-i** parameter or the **-G** parameter.
- Specify the **-g** parameter to search a specific application group. You must specify the search using the **-i** parameter.
- Specify both the **-f** parameter and the **-G** parameter. The ARSDOC program will verify that the application group can be searched from the folder.

When you specify the **-x** parameter, you cannot specify the **-X** parameter.

-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 PRINT function, the **-i SQL** parameter can be specified when the -X flag is used. If it is specified, it is used only if the retrieve fails using the loadID. If the -X flag is used, the application group name must be provided using the -G flag.

For the HOLD_ADD function, add documents to a hold by using the index file that was generated for the specified *loadId*.

For the HOLD_RELEASE function, remove documents from a hold by using the index file that was generated for the specified *loadId*.

For the CFSOD-FED function, send documents to CFS-OD and make them available to IBM FileNet P8 features 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 **-i**, **-q**, **-S**, and **-f** parameters.

Examples

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 rockies -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 rockies -F parmfile -v
   ```

   The parameter file contains two queries:

```
[-f "Student Information"] [-S 1/1/97,12/31/97] [-o (student)(type)] \
[-i "WHERE student='123420010056' AND type='B' OR type='G' OR type='T'"]

[-f "Student Information"] [-S 1/1/97,12/31/97] [-o (student)(type)] \
[-i "WHERE student='123450011917' AND type='B' OR type='G' OR type='T'"]
```

Assuming 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
```

3. 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 rockies -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.TRANSCRIPTS.ind

student.1.Student Information.TRANSCRIPTS.out

student.1.Student Information.TRANSCRIPTS.res
```

For AFP data, if there is more than one version of the resource group per application, then the ARSDOC program can generate additional output files. For example:

```
student.1.Student Information.TRANSCRIPTS.ind

student.1.Student Information.TRANSCRIPTS.out

student.1.Student Information.TRANSCRIPTS.res

student.2.Student Information.TRANSCRIPTS.out

student.2.Student Information.TRANSCRIPTS.ind

student.2.Student Information.TRANSCRIPTS.res
```

If the application group contains more than one application, 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
```

4. 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 rockies -f "Student Information" -o query1.out -H
-S 1/1/97,12/31/97 -i "WHERE student='0012-0034-0056' AND type='B' OR
type='G' OR type='T'" -v
```

5. The following shows how to use the QUERY function and a parameter file to run more than on query at a time. The parameter file is in the current directory. The output files are saved in the current directory.

```
arsdoc query -h rockies -F parmfile -v
```

The parameter file contains three queries:

```
[-f "Student Information"] [-i "WHERE type='B'"] [-o query2.out] \
[-S 1/1/97,12/31/97] [-H] [-N (student)(id)(p_date)]

[-f "Student Information"] [-i "WHERE type='G'"] [-o query3.out] \
[-S 1/1/97,12/31/97] [-H] [-N (student)(id)(p_date)]

[-f "Student Information"] [-i "WHERE type='T'"] [-o query4.out] \
[-S 1/1/97,12/31/97] [-H] [-N (student)(id)(p_date)]
```

6. The following example shows how to use the QUERY function to search a specific application group:

```
arsdoc query -h rockies -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 rockies -o /newdata/crd.dat -n "crd_date=01/21/98"
-n "account='000-000-000'" -n balance=123.45 -n "name='John Watpole'"
-f "Credit Card Statements" -g CRD -a CRD -v
```

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 rockies -i "where account='000-000-000' and
name='John Watpole'" -n "balance=0.00" -S 1/1/97,12/31/97 -f
"Credit Card Statements" -g CRD -v
```

9. The following example shows how to use the DELETE function to delete a document from the Credit Card Statements folder:

```
arsdoc delete -h rockies -i "where account='000-000-000' and
name='John Watpole'" -f "Credit Card Statements" -S 1/1/97,12/31/97
-v
```

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 rockies -P
svrprt1 -i "where account='000-000-000' and name='John Watpole'"
-f "Credit Card Statements" -S 1/1/97,12/31/97 -v
```

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 rockies -i "where sdate=10005 and account='000-000-000'
and name='Joe Smith'" -n "sdate=5/23/97" -n "account=000-000-000" -n
"name='Sally Smith'" -f "Credit Card Statements" -S 1/1/97,6/31/97 -v
```

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 rockies -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 rockies -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 rockies -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, the database is not queried; rather, documents are retrieved based on the information in the index file.

```
arsdoc get -h rockies -X 19867-025-0-3FAA-10136-10136
-G load-scanned-images-jpeg -o jpeg -v
```

16. 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 <date range>` is not specified, all tables will be searched, possibly resulting in poor performance.

17. **-A** parameter: The following are two examples of using the ARSDOC GET function with the **-A** parameter.

- To retrieve public text annotations:

```
arsdoc get -u oduser -p passwd -h odserver -f "CRD" -q named_query
 -o loaddata -a -g -c -N -A 0 -v
```

- An example of the messages that are generated by the ARSDOC program. New messages are highlighted:

```
03/10/2004 10:32:12: Starting arsdoc. Version: 8.4.1.0
03/10/2004 10:32:14: arsdoc get -u oduser -h odserver -f CRD -q named_query -o loaddata -a -g -c -N -A 0 -v
03/10/2004 10:32:14: Attempting login for userid 'oduser' on server 'odserver' ...
03/10/2004 10:32:20: Login successful
03/10/2004 10:32:20: Searching for folder 'CRD' ...
03/10/2004 10:32:27: Search successful
```

```
03/10/2004 10:32:27: Searching for documents in 'CRD' ...

03/10/2004 10:32:28: Querying database with SQL string 'where account = '000-000-000''

03/10/2004 10:32:56: Search successful

03/10/2004 10:32:56: 1 document(s) have been queried. Retrieving 1 document(s).

03/10/2004 10:34:05: (1): Retrieving document for userid 'oduser' ...

03/10/2004 10:34:05: Document successfully retrieved and stored in file 'loaddata.2.CRD.CRD.out'

03/10/2004 10:34:15: Writing generic indexer file(s).

03/10/2004 10:34:46: '1' annotations were written to file 'loaddata.2.CRD.CRD.ann'

03/10/2004 10:34:46: A total of 1 annotations were written to file 'loaddata.2.CRD.CRD.ann'

03/10/2004 10:34:53: Generic indexer file 'loaddata.2.CRD.CRD.ind' has been successfully created.

03/10/2004 10:34:54: arsdoc completed.
```

18. **-Q** parameter: The following is an example of the messages that are generated by the ARSDOC program when multiple SQL queries are used. New messages are highlighted:

```
03/10/2004 10:32:12: Starting arsdoc. Version: 8.4.1.0

03/10/2004 10:32:14: arsdoc get -u oduser -h odserver -f CRD -Q SQL-filename -o loaddata -a -g -c -N -v

03/10/2004 10:32:14: Attempting login for userid 'oduser' on server 'odserver' ...

03/10/2004 10:32:20: Login successful

03/10/2004 10:32:20: Searching for folder 'CRD' ...

03/10/2004 10:32:27: Search successful

03/10/2004 10:32:27: Searching for documents in 'CRD' ...

03/10/2004 10:32:28: Querying database with SQL string 'where account = '000-000-000''

03/10/2004 10:32:28: 1 document(s) matches the search criteria.'
03/10/2004 10:32:29: Search successful

03/10/2004 10:32:29: Searching for documents in 'CRD' ...

03/10/2004 10:32:30: Querying database with SQL string 'where account = '000-000-001''

03/10/2004 10:32:32: 1 document(s) matches the search criteria.'

03/10/2004 10:32:33: Search successful

03/10/2004 10:32:33: Searching for documents in 'CRD' ...

03/10/2004 10:32:34: Querying database with SQL string 'where account = '000-000-002''

03/10/2004 10:32:38: 1 document(s) matches the search criteria.'

03/10/2004 10:32:38: Search successful

03/10/2004 10:32:56: 3 document(s) have been queried. Retrieving 3 document(s).

03/10/2004 10:33:05: (1): Retrieving document for userid 'oduser' ...

03/10/2004 10:33:05: Document successfully retrieved and stored in file 'loaddata.2.CRD.CRD.out'

03/10/2004 10:33:07: (2): Retrieving document for userid od'user' ...

03/10/2004 10:33:10: Document successfully retrieved and stored in file 'loaddata.2.CRD.CRD.out'

03/10/2004 10:33:12: (3): Retrieving document for userid 'oduser' ...

03/10/2004 10:33:15: Document successfully retrieved and stored in file 'loaddata.2.CRD.CRD.out'

03/10/2004 10:33:24: arsdoc completed.
```

19. **-U** parameter: In this example, if a common user ID *oduser* is shared by three users: John Smith, Mark Jones, and Kathy Brown. A user alias is assigned to identify each of the actual users. In this example, the user alias is the real user name.

```
arsdoc query -u oduser -p odpasswd -h host -i query_string -f folder
-v -U "John Smith"
arsdoc query -u oduser -p odpasswd -h host -i query_string -f folder
-v -U "Mark Jones"
arsdoc query -u oduser -p odpasswd -h host -i query_string -f folder
-v -U "Kathy Brown"
```

In the system log, the user ID column then contains:

```
ODUSER-JOHN SMITH
ODUSER-MARK JONES
ODUSER-KATHY BROWN
```

20. HOLD_ADD function:

```
arsdoc hold_add -u oduser -p odpasswd -h odserver -l hold_audit -f
"Monthly Status Reports" -i "where code='TX' and sdate=14117" -v
```

21. HOLD_RELEASE function:

```
arsdoc hold_release -u oduser -p odpasswd -h odserver -l hold_audit -f
"Monthly Status Reports" -i "where code='TX' and sdate=14117" -v
```

22. CFSOD_FED function:

```
arsdoc cfsod_fed -u oduser -p odpasswd -h odserver -f
"Monthly Status Reports" -i "where code='TX' and sdate=14117" -v
```

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 print the PTF version number and the ARSDOC GET function can print the number of documents that were queried and retrieved and print a status message for each document that is retrieved. To enable the messages, specify the **-v** parameter.

For the purpose of retrieving a document from an OnDemand server (with the intention of reloading or loading the data into another OnDemand serve), ARSDOC GET should be used to retrieve reports, as opposed to the ARSADMIN RETRIEVE and DECOMPRESS.

You can use ARSDOC GET with a load ID by specifying the **-X** parameter, for example

```
arsdoc get ...... -X <loadid>.
```

Use the load ID as it appears in the system log, for example

```
-X  5804-1-0-5FAA-12360-12360
```

You can also specify the **-g** parameter to produce a generic index file. Alternatively, the documents retrieved can be re-indexed.

When running ARSDOC GET **-a** with the **-g** parameter to create generic indexer files, the `.res` resource file is created. See example 16 in "Examples" on page 268.

Files

/usr/lpp/ars/bin/arsdoc
> The AIX executable program.

/opt/ondemand/bin/arsdoc
> The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsdoc
> The Windows executable program.

ARSJESD

Syntax

```
►►──arsjesd──-p──port#──┬──-d──filesystem──┬──────────────────────►◄
                        └◄─────────────────┘
                                          └──-x──command──┘
```

Description

The ARSJESD program is the server component of Download. A licensed feature of PSF for z/OS, Download can be used to automatically transmit the output data sets of application programs from the JES spool to file systems on Content Manager OnDemand servers. Download also transmits values that you specify in JCL parameters and can optionally transmit additional job information. You can use the values and job information to specify the name of the application and application group into which Content Manager OnDemand loads the transmitted data. Download is made up of a pair of cooperating components, a Download writer that transmits the data set and a program that receives and stores the data on the server. See *PSF for z/OS: Download for z/OS* for details about Download, including how to configure and operate Download on z/OS systems, a list of the JCL parameters that are transmitted with a file, and information about a user-written program that may be used to provide additional job information to the ARSJESD program. See "Download exits and processing" on page 193 for more information about using the Download user exit with Content Manager OnDemand.

The ARSJESD program monitors a TCP/IP port for activity from a z/OS system. Multiple instances of the ARSJESD program may be started, each monitoring a specific port for data transmitted from a different Download writer. A Download writer and its corresponding ARSJESD instance must specify the same port number. If you start multiple instances of the ARSJESD program, you must specify different file systems for use by each instance. Using the same file systems for more than one ARSJESD program can adversely affect performance.

The ARSJESD program now receives files larger than 4GB.

The ARSJESD program stores the transmitted data sets into file systems on the server. You can specify up to ten file systems that the ARSJESD program can use to store the transmitted data. The ARSJESD program always stores data beginning with the first file system specified. If there is not enough space available to store the data set being transmitted in the first file system specified, the ARSJESD command switches to the second file system specified. If there is not enough space available to store the data set being transmitted in the last file system specified, the ARSJESD program switches to the first file system specified.

Parameters

-p port#

The TCP/IP port number that the ARSJESD program monitors for data from the z/OS system. The valid range of port numbers is 5001 through

64000. However, you should avoid using port numbers 5001, 8251, and 8253. The Download writer and the ARSJESD program must specify the same port number. See *PSF for z/OS: Download for z/OS* for information about configuring Download on z/OS systems.

-d filesystem

The name of the file system into which the ARSJESD program stores the transmitted data. You can specify up to 10 file system parameters, each separated by the blank character.

-x command

The name of a user-written program that the ARSJESD program invokes to process the downloaded file, JCL parameters, and other job information after receiving and storing the file on the server. You can specify the full path name of the program. If you do not specify the full path name, the operating system searches the current directory for the program. If the file is not found in the current directory, the operating system continues to search for the file using the directories defined in the PATH environment variable.

You typically specify this parameter when you need to process the transmitted data before it can be loaded into the system. See "Download exits and processing" on page 193 for more information.

Examples

1. The following shows how to start the ARSJESD program to monitor TCP/IP port number 6001 and place transmitted data in the /arsacif/acif1 file system:

   ```
   arsjesd -p 6001 -d /arsacif/acif1
   ```

2. The following shows how to start the ARSJESD program to monitor TCP/IP port number 6002 and place transmitted data in the /arsacif/acif1 file system. If there is not enough space available in the /arsacif/acif1 file system to store the data set being transmitted, then the ARSJESD program stores the data set in the /arsacif/acif2 file system.

   ```
   arsjesd -p 6002 -d /arsacif/acif1 -d /arsacif/acif2
   ```

3. The following shows how to start the ARSJESD program to monitor TCP/IP port number 6002 and place transmitted data in the /arsacif/acif1 file system. After transmitting the file to the server, the ARSJESD program invokes the /usr/lpp/ars/exits/download.exe user-written program. A user-written program may perform any required functions, such as determining the name of the application group and application to load, renaming or copying input files, and invoking other programs.

   ```
   arsjesd -p 6002 -d /arsacif/acif1 -x /usr/lpp/ars/exits/download.exe
   ```

   Note: You must supply your own user-written program, although IBM provides a sample program in "Download exits and processing" on page 193.

Notes

A file transmitted by Download uses the following file naming convention:

```
MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD
```

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

- By default, the ARSLOAD program uses the `FORM` part of the file name to identify the application group to load. When you run the ARSLOAD program,

you can use the **-G** parameter to specify a different part of the file name to identify the application group (MVS™, JOBNAME, or DATASET). For example, `arsload -G JOBNAME`.

- If the application group to load contains more than one application (source of data), you must identify the application to load. Otherwise, the load will fail. When you run the ARSLOAD program, you can use the **-A** parameter to specify the part of the file name that identifies the application (MVS, JOBNAME, DATASET, or FORMS). For example, to use the `DATASET` part of the file name to identify the application, run the ARSLOAD program with the **-A** `DATASET` parameter.

Depending on the date format that you specify when you initiate Download on the z/OS system, the date part of the file name may be `YYYYDDD` or `YYDDD`. Apply APAR OW36539 to Download and use the DATE_FORMAT keyword in the routing file to control the format of the date part of the file name. See *PSF for z/OS: Download for z/OS* for details.

Files

/usr/lpp/ars/bin/arsjesd
> The AIX executable program.

/opt/ondemand/bin/arsjesd
> The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for WinNT\bin\arsjesd
> The Windows executable program.

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

```
►►─ arsload ─┬──────────────────┬──┬─ -B─Format ─┬──┬─ -b─Field name ─┬──►
             ├─ -a─application ─┤  └─────────────┘  └──────────────────┘
             └─ -A─applID ──────┘


►─┬─ -c─indexDir ─┬──┬──────────────────┬──┬─ -E ─┬──┬─ -f ─┬──┬─ -F─C─T─X ─┬──►
  └───────────────┘  │  ◄──────────────  │  └──────┘  └──────┘  └────────────┘
                     ├──── -d─dataDir ───┤
                     │  ◄──────────────  │
                     └──── -D─DELIM ─────┘


►─┬─ -g─applGroup ─────┬──┬─ -h─instance ─┬──┬─ -i ─┬──┬─ -I─instance ─┬──►
  ├─ -G─applGroupID ───┤  └────────────────┘  └──────┘  └───────────────┘
  └────────────────────┘


►─┬─ -j─parmFile ─┬──┬─ -J─file_name_delimiter ─┬──┬─ -n ─┬──┬─ -p─password ─┬──►
  └───────────────┘  └──────────────────────────┘  └──────┘  └───────────────┘


►─┬─ -t─seconds ─┬──┬─ -T ─┬──┬─ -u─userid ─┬──┬─ -U─filename ─┬──┬─ -v ─┬──►
  └──────────────┘  └──────┘  └─────────────┘  └───────────────┘  └──────┘


►─┬─ -X ──────────┬──┬─ -Z─userExit ─┬──┬────────────────────┬──┬─ -1─tracefile ─┬──►◄
  │  ┌─ A ─┐      │  └────────────────┘  │  ◄──────────────   │  └────────────────┘
  └──┤     ├──────┘                      └──── loadFilename ──┘
     └─ G ─┘
```

                                    ►──┬─────────────┬──────────────────────────◄►
                                       └─ -2─*level*─┘
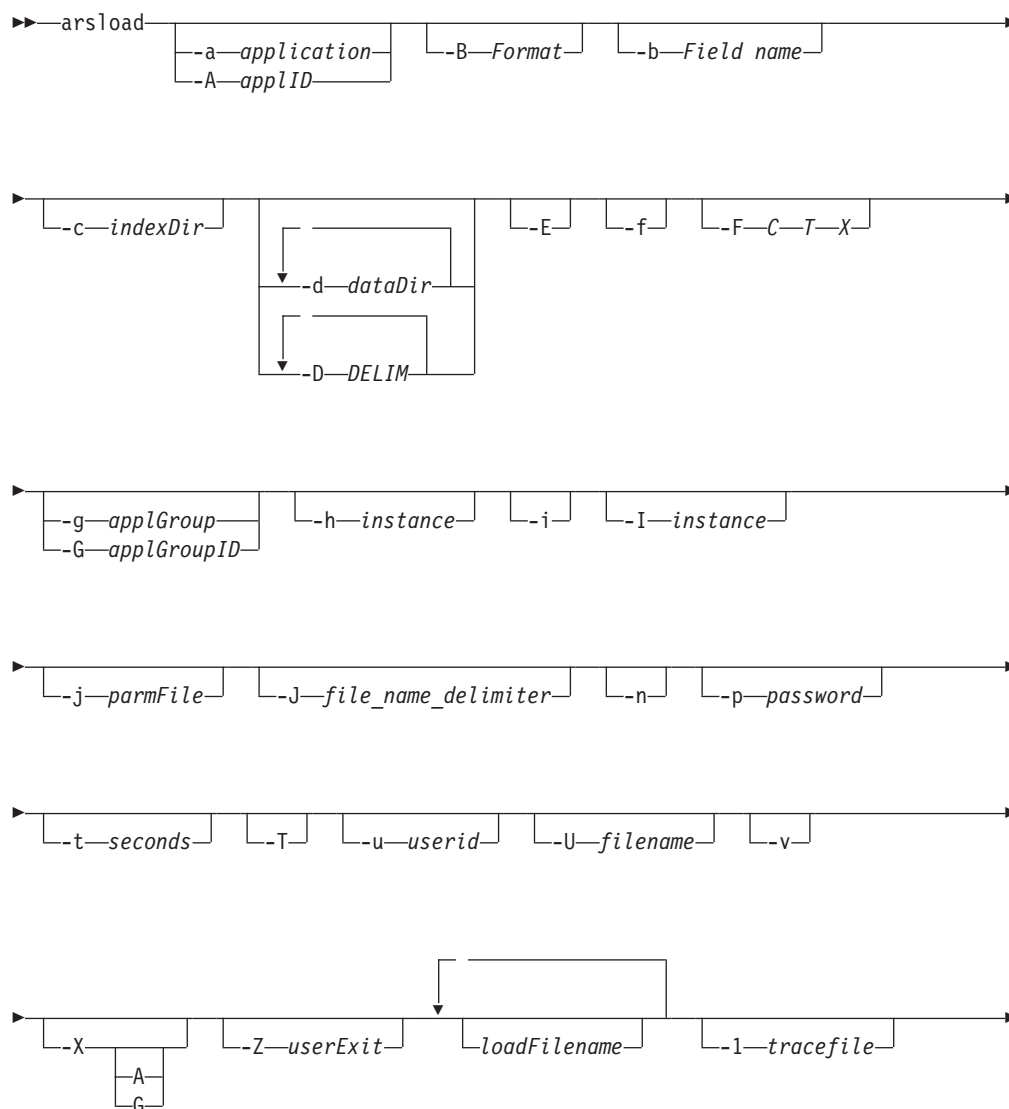
Description

The ARSLOAD program is the main IBM 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 daemon (UNIX
servers) or service (Windows servers), to periodically check specified file systems
for input files to process.

The ARSLOAD program can use the following sources for input files to process:
- one or more file systems specified with one or more **-d** parameters,
- 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 directories that are specified by the **-d**
parameter. If you omit the load file name and do not specify the daemon mode
parameter (**-d**), the ARSLOAD program will issue a usage note and exit.

Important: When running the ARSLOAD program in daemon mode, the `.ARD` and
`.PDF` file name extensions are required to initiate a load process.

When you run the ARSLOAD program, you must provide the user ID and
password of an 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 user ID and password:
- Use the **-u** and **-p** parameters each time that you run the ARSLOAD program.
- Use the **-U** parameter each time that you run the ARSLOAD program to name a
  file that contains a single line with the userid and password of a user that has
  administrator authority for the application group.
- On UNIX servers, set a default system administrator userid and password in the
  `ARSLOAD.CFG` configuration file. The ARSLOAD program uses the values in the
  `ARSLOAD.CFG` file unless you specify otherwise (with the **-u** and **-p** parameters) or
  if you specify a different configuration file with the **-U** parameter.
- On Windows servers, the Load Data service uses the userid and password of the
  instance owner. If you run the ARSLOAD program from the command line, you
  should use Windows unified logon. See the **-u** and **-U** parameters for details.

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 files

into different application groups and applications, then use the **-A** parameter to specify the part of the file name that identifies the name of the application.

-A applID

Determines the part of the 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 specify this parameter when you run the ARSLOAD program as a daemon (UNIX servers) or service (Windows servers) to automate the loading of files into different application groups and applications. For example, you can use Download to transmit files from z/OS systems to the server. A file transmitted by Download uses the following file naming convention:

`MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD`

- Unless you specify otherwise, the ARSLOAD program uses the `FORMS` 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 (MVS, JOBNAME, or DATASET). 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. When you run the ARSLOAD program, you can use the **-A** parameter to specify the part of the file name that identifies the application (MVS, JOBNAME, DATASET, or FORMS). For example, to use the `DATASET` part of the file name to identify the application, run the ARSLOAD program with the **-A DATASET** parameter.

-b <field>

Field identifier for file name index parameter. Specifies the field name in the application group that is set to the value that is specified by the IDX in the **-B** parameter. If an IDX is not identified by the **-B** parameter, this value is ignored.

For example,

`arsload -b "field1" -B "AG.IDXAPP.ARD" -g ApplGroup1 ... ApplGroup1.App1.ARD`

takes the value `App1` from the input file name and insert it into `"field1"` of application group `"ApplGroup1"` during the load. In this example we set `"field1"` to the name of the application that is loaded, `"App1."`

-B Format

Use this parameter to define file name formats for MVS download files and files that are processed by the ARSLOAD daemon. Use the following to create the file name format:

WRI	WRITER
MVS	MVS
JOB	JOBNAME
DAT	DATASET
FOR	FORMS
YY	YYDDD
HH	HHMMM
AG	Application group name

APP   Application name

IDX   Indicates which part of the file name to use as index

IGN   Ignore

.EXT   Extension

The letters in lowercase are optional.

Here is an example of the file name in the default format:
```
MVS.JOBNAME.DATASET.FORMS.YYDDD.HHMMM.ARD
```

The file name format can also be written as:
```
MVS.JOB.DAT.FOR.YY.HH.EXT
```

or
```
MVS.JOB.IDXDAT.FOR.YY.HH.EXT
```

to associate the DAT part of file name as a load index.

Here are a few more examples:
```
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"
```

Here are two examples of using the IDX identifier:

- Simple index input format:
  ```
  arsload —b "account_num" —B "AG.APP.IDX"
  ```

  where the system uses the IDX part of the file name as an index
  parameter for the account_num field.

- Dual purpose index format:
  ```
  arsload —b "report_name" —B "AG.IDXAPP"
  ```

  where the system uses the APP part of the file name as an index
  parameter for the report_name field

You can use the *-B* parameter or the *-g* and *-a* parameters combined but
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 filename.

-c indexDir
> The file system in which IBM Content Manager OnDemand temporarily
> stores data created by the indexing program. The default location is the
> directory from which the ARSLOAD program was invoked.
>
> For better performance, IBM recommends that the **-c** parameter specify a
> different file system than the file system that is specified with the **-d**
> parameter.

-d dataDir
> Specifies a file system that contains input files to process.
>
> Any file with a file type extension of .ARD or .PDF will be processed (.ARD
> files are transmitted to the server by Download; .PDF files are created by
> Acrobat Distiller). The case of the file type extension is not significant.

You can specify this parameter one or more times. The ARSLOAD program will search for input files to load in each of the directories that you specify.

For better performance, IBM recommends that the **-d** parameter specify a different file system than the file system that is specified with the **-c** parameter.

Files transmitted by Download from a z/OS system must conform to the following file naming convention:

`MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD`

Important: The `.ARD` file name extension is required to initiate a load process.
Unless you specify otherwise, the ARSLOAD program uses the `FORMS` part of the file name to identify the application group to load. However, you can use the **-g** and **-A** parameters to specify different parts of the file name (MVS, JOBNAME, DATASET, or FORMS) to identify the application group and application to load.

-E   When specified, the user-written program that is associated with the ARSUUPDT exit point is to be invoked. If the **-E** parameter is not specified, the user-written program is not invoked. The user-written program may be used to modify some of the parameters that are used by Content Manager OnDemand when document data is being 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 Multiplatforms: Installation and 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.

Important: 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 "ARSADMIN" on page 215 for help with removing the data that is created when a file is loaded into the system.

-F Trace output format
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.

-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. However, if you specify the **-d** parameter, unless you specify otherwise, the ARSLOAD program uses the FORMS part of the file name to determine the name of the application group to load. If you plan to automate the loading of files into different application groups and applications, then you should use the **-g** parameter to specify the part of the file name that identifies the application group to load.

-G applGroupID

Determines the part of the file name that the ARSLOAD program uses to identify the name of the application group to load.

You typically specify this parameter when you run the ARSLOAD program as a daemon (UNIX servers) or service (Windows servers) to automate the loading of files into different application groups and applications. For example, you can use Download to transmit files from OS/390 systems to the server. A file transmitted by Download uses the following file naming convention:

MVS.JOBNAME.DATASET.FORMS.YYYYDDD.HHMMSST.ARD

- Unless you specify otherwise, the ARSLOAD program uses the FORMS 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 (MVS, JOBNAME, or DATASET). 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. When you run the ARSLOAD program, you can use the **-A** parameter to specify the part of the file name that identifies the application (MVS, JOBNAME, DATASET, or FORMS). For example, to use the DATASET part of the file name to identify the application, run the ARSLOAD program with the **-A DATASET** parameter.

-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 (UNIX servers; the registry for Windows servers), 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.

Tip: 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 that the system on which the instance that you want to process is running.

  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 Multiplatforms: Installation and Configuration Guide* for information about configuring instances.

-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 (UNIX servers; the registry for Windows servers), 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.

Tip: 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 that the system on which the instance that you want to process is running.

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 Multiplatforms: Installation and Configuration Guide* for information about configuring instances.

-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.

Typical usages of this command include:
- Appending a document name to an index value.
- Populating an index value with something not seen within the data.
- Populating an index value with something that does not change regularly.

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

applications.) 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.

The following shows an example of a file that contains additional indexing parameters to be processed by ACIF.

```
FORMDEF=F1ABBB0
PAGEDEF=P1ABBB0
USERLIB=/usr/acif/resources/sapkuli1
```

-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 "**.**"

-n

Determines whether Content Manager OnDemand deletes the input files when the ARSLOAD program terminates.

- 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 file name as the original input file.

-p password

The password for the user that is specified with the **-u** parameter. If the user is not assigned a password, enter a null password (that is, specify **-p** "").

-t seconds

Determines the polling time in seconds. This is the interval of time in which the ARSLOAD program checks the input directory (specified by the **-d** parameter) for input files to process. The default value is 600 seconds, which means that the ARSLOAD program checks the input directory every ten minutes.

-T Fully qualified trace file name

When a file is specified, trace is activated for ARSLOAD. If the file already exists, it is renamed with the current date and time, and a new file is created.

-u user ID

The user ID of an Content Manager OnDemand user with administrator authority for the application group. The user must have permission to add documents to the application group.

There are several ways that you can provide the user ID and password:

- You can specify the **-u** and **-p** parameters each time that you run the ARSLOAD program.

- For Windows servers, if you run the ARSLOAD program from the command line, use Windows unified logon to protect your Content

Manager OnDemand user ID and password. Unified logon attempts to log on to the Content Manager OnDemand server with the Windows logon account user name.

- For UNIX servers, you can specify the **-U** parameter each time that you run the ARSLOAD program and name a file that contains a single line with the user ID and password of a user with administrator authority for the application group.
- For UNIX servers, you can specify a system administrator user ID and password in the ARSLOAD.CFG configuration file. The ARSLOAD program uses the values in the ARSLOAD.CFG file, unless you specify otherwise (with the **-u** and **-p** parameters or if you specify a different configuration file with the **-U** parameter).
- For Windows servers, the Load Data service uses the user ID and password of the instance owner.

-U filename

For UNIX servers, if you omit the **-u** parameter, determines the file that contains the user ID and password that the ARSLOAD program uses to log on to the system and access the application group. If the ARSLOAD program cannot locate or use the file named with the **-U** parameter, it attempts to log on to the system and access the application group with a user ID of `admin` and no password. If you omit the **-U** parameter (and the **-u** parameter), Content Manager OnDemand attempts to obtain the user ID and password from the `/usr/lpp/ars/config/arsload.cfg` file. If the ARSLOAD program cannot locate or use the `/usr/lpp/ars/config/arsload.cfg`, it attempts to log on to the system and access the application group with a user ID of `admin` and no password. The *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* describes how to configure the `arsload.cfg` file.

Important: After you configure the file named with the **-U** parameter (or the `arsload.cfg` file), remember to change the password any time that you change the user's password in Content Manager OnDemand; otherwise the load will fail. The ARSLOAD program can accept an expired password; however, the ARSLOAD program will fail if you specify an incorrect password.

Restriction: The `ARSLOAD.CFG` file and the **-U** parameter are not used on Windows servers. The Content Manager OnDemand Load Data service logs on to the system and accesses the application group with the user ID and password of the instance owner (typically a Content Manager OnDemand system administrator user). If you run the ARSLOAD program from the command line, Windows unified logon should be used to log on to the system and access the application group. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for help with configuring services and unified logon.

-v   Enables verbose mode, which displays all messages (informational and error). By default, the ARSLOAD program displays error messages.

-X indexer

Allows you to override the indexing program that was specified on the Indexer Information page for the application. The possible values are `A` and `G`:

- Specify `-X A` to use ACIF.

- Specify `-X G` to use the Content Manager OnDemand Generic 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 program generates index data that is in the Generic indexer format. To load the index data into the application, you must specify:

```
arsload -X G . . .
```

-Z userExit

Specifies a user-defined string that is passed to the load user exit program (ARSULOAD). The ARSULOAD program is installed in the `exits` directory.

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 concatenates the following file type extensions to the file name that you specify: `IND`, `OUT`, and `RES`

- The `IND` file contains the index data
- The `OUT` file contains the report data
- The `RES` file contains the resource data

For example, if you specify `arsload -g BILLS po3510`, the ARSLOAD program processes the following files:

- `po3510.ind`
- `po3510.out`
- `po3510.res`

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 omit an input file name, the ARSLOAD program will run in daemon mode and attempt to load input data from the directories that are specified by the **-d** parameter. If you omit an input file name and do not specify the daemon mode parameter (**-d**), the ARSLOAD program will issue a usage note and exit.

-1 Fully qualified trace file name

When a file is specified, trace is activated for ARSLOAD. If the file already exists, it is renamed with the current date and time, and a new file is created.

-2 Trace level number

Optional parameter when the -1 parameter is specified. The values are additive.

1   Errors

2   Warnings

4   Info

Provides informational trace messages for debugging problems.

| **Restriction:** Use only under the supervision of support because it
| might affect performance.

| 8   Flow

| Provides function entry and exit information.

| **Restriction:** Use only under the supervision of support because it
| might affect performance.

Examples

1. The following shows how to run the ARSLOAD program to check the specified
   directory for input files to process. The input files must have a file type
   extension of `.ARD` or `.PDF`. The ARSLOAD program stores temporary work files
   in the location specified by the **-c** parameter. In this example, the ARSLOAD
   program uses the `FORM` part of the file name to determine the application group
   to load and the application group contains only one application; the ARSLOAD
   program logs on to the system and accesses the application group by using the
   use ID and password from the`/usr/lpp/ars/config/arsload.cfg` file.

   ```
   arsload -c /arsacif/acif1 -d /arsacif/acif2
   ```

2. The following shows how to run the ARSLOAD program to check the specified
   directory for input files to process. The input files must have a file type
   extension of `.ARD` or `.PDF`. The ARSLOAD program stores temporary work files
   in the location specified by the **-c** parameter. In this example, the ARSLOAD
   program uses the `JOBNAME` part of the file name to determine the application
   group to load and the `DATASET` part of the file name to determine the
   application to load; the ARSLOAD program logs on to the system and accesses
   the application group by using the use ID and password from
   the`/usr/lpp/ars/config/arsload.cfg` file.

   ```
   arsload -c /arsacif/acif1 -d /arsacif/acif2 -A DATASET -G JOBNAME
   ```

3. The following shows how to run the ARSLOAD program to load the specified
   file into the specified application group. The ARSLOAD program logs on to the
   system and accesses the application group by using the user ID and password
   from the **-u** and **-p** parameters.

   ```
   arsload -g BILLS -u bob -p secret PO3510
   ```

4. The following shows how to run the ARSLOAD program to load several input
   files into the specified application group. The application group name contains
   an embedded blank character, and must be quoted. The ARSLOAD program
   logs on to the system and accesses the application group by using the user ID
   and password from the **-u** and **-p** parameters.

   ```
   arsload -g "ABC Credit" -u bob -p secret RW7505 RW8505
   ```

5. The following shows how to run the ARSLOAD program to load the specified
   file into the specified application group. The TCP/IP address of the IBM
   Content Manager OnDemand library server is specified. This format of the
   command can be used to load input files from an object server and store the
   index data on the library server. The ARSLOAD program logs on to the system
   and accesses the application group by using the user ID and password from the
   file named with the **-U** parameter.

   ```
   arsload -g BILLS -h "9.99.111.222" -U /tmp/userpw.cfg ARSP0MST
   ```

Notes

IBM recommends that you do not run the ARSLOAD program at the same time that you run the ARSMAINT program with the **-r** parameter (or any other program that optimizes or reorganizes the database).

IBM recommends that you do not run the ARSLOAD program at the same time that you run the ARSDB program with the **-m** or **-s** parameters (or any other program that optimizes or reorganizes the database).

If you do not specify an input file name, the ARSLOAD program checks the directory specified with the **-d** parameter(s) for input files to process. Any file with a file type extension of `.ARD` or `.PDF` will be processed (`.ARD` files are transmitted to the server by Download; `.PDF` files are created by Acrobat Distiller). The case of the file type extension is not significant.

The IBM Content Manager OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) must be running, otherwise the ARSLOAD program will fail.

Files

/usr/lpp/ars/bin/arsload
> The AIX executable program.

/opt/ondemand/bin/arsload
> The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsload
> The Windows executable program.

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.

Note to Oracle users: See "Using the ARSMAINT program with Oracle" on page 297.

Syntax

```
▶▶──arsmaint──┬─────┬──┬─────┬──┬─────┬──┬──────────────────┬──┬─────────────────┬──▶
              └─-c──┘  └─-d──┘  └─-e──┘  │   ┌─95────────┐   │  └─-g──applGroup───┘
                                         └─-f─┴─threshold─┘

 ▶──┬─────┬──┬─────────────┬──┬─────┬──┬─────┬──┬─────────────┬──┬──────────────┬──▶
    └─-i──┘  └─-I──instance─┘  └─-L──┘  └─-m──┘  └─-n──minimum──┘  └─-p──password──┘

 ▶──┬─────┬──┬─────┬──┬──────────────────┬──┬────────────┬──┬─────┬──────────────────▶
    └─-r──┘  └─-s──┘  └─-t──internalDate──┘  └─-u──userid──┘  └─-v──┘

 ▶──┬──────────────────┬──────────────────────────────────────────────────────────▶◀
    └─-x──maximum───────┘
```

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 Tivoli Storage Manager or Object Access Method (OAM), 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.

-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. 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 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 Multiplatforms: Installation and 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 stdout (UNIX servers), the console (Windows servers), the system log, and the system log user exit point.

-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 Length of Time to Keep Imported Migrated Indexes (see 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 (UNIX servers; the registry for Windows servers), 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.
- You are running the ARSMAINT program on a system other that 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 workstation, always specify the **-I** 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 Multiplatforms: Installation and Configuration Guide* for information about configuring instances.

-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.

-p password

Specifies the password for the Content Manager OnDemand userid that is identified 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 this parameter, the ARSMAINT program prompts you to enter the password when you run the program. 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. IBM recommends that you schedule the ARSMAINT program to optimize the database every day.

-s   Prints a report that contains status information about the cache storage file systems. The report includes information such as the name of the file systems, and the amount of free space for each file system. The ARSMAINT program saves the report in the system log. IBM recommends that you schedule the ARSMAINT program every day to report the status.

-t internalDate

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 value that you specify must be a valid Content Manager OnDemand internal date value. You can use the ARSDATE program to display the internal date value for a given date string. When you specify this option, you must specify the **-u** and **-p** parameters. 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 an 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.

-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.

Examples

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 -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.

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.

IBM recommends that you do not load data on the system when you run the ARSMAINT program with the **-r** parameter.

It is recommended that you create a backup image of the Content Manager OnDemand database and the Tivoli Storage Manager database before you run the ARSMAINT program.

The Content Manager OnDemand server program (ARSSOCKD or ARSOBJD in UNIX; LibSrvr or ObjSrvr in Windows) must be running, otherwise the ARSMAINT program will fail.

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 program with the **-a1** parameter. For more information, see "ARSTBLSP" on page 299.

If the 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 ls 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.

Using the ARSMAINT program with Oracle

You can use the ARSMAINT program to maintain the tables that contain user-defined application group data. User-defined application groups are the application groups that you define to the system. Customers in a high maintenance operation should run the ARSMAINT program on a regular schedule.

The syntax is:

```
/opt/ondemand/bin/arsmaint <options>
```

The options are:

–d, –i   Expire index data from the database. The **–i** parameter expires index data that has been imported from archive storage. If you do not migrate index data to archive storage, you do not need to specify the **–i** parameter.

–e   Migrate index data from the database to archive storage. If you do not migrate index data to archive storage, you do not need to specify the **–e** parameter.

–r   Collect statistics. **Note:** The ARSMAINT program collects statistics only on the tables that have changed since the last time that statistics were collected. Content Manager OnDemand maintains information about the tables, including the last time that a table was modified and the last time that statistics were collected on a table.

–g applGroup
Process the tables for the specified application group. If you do not specify this parameter and name an application group, the ARSMAINT program processes all of the user-defined application groups.

Files

/usr/lpp/ars/bin/arsmaint
The AIX executable program.

/opt/ondemand/bin/arsmaint
The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsmaint
The Windows executable program.

ARSTBLSP

Purpose

Migrate tables of application group data from the default table space to separate table spaces. The ARSTBLSP program may also be used to change the table that Content Manager OnDemand loads data into.

Note: The ARSTBLSP program is available only for DB2.

Syntax

```
►►──arstblsp──-a─action──┬──────────┬──-g─applGroup──┬──────────────┬──►
                         └─-d─dir──┘                  └─-I─instance─┘

►──┬──────────┬──►◄
   └─-t─table─┘
```

Description

The ARSTBLSP program can be used to migrate existing tables of application group data from the default table space to separate table spaces. IBM recommends that all customers who are upgrading from earlier versions of Content Manager OnDemand use this program to migrate their existing application group data to table spaces.

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, a customer may desire to close a table to loading before the Maximum Rows value is reached. For example, migration processing (by using the ARSMAINT **-e** function) will not process a table that is open for loading, and the customer may desire to migrate the table earlier than initially anticipated.

Parameters

-a action
    The action to perform. The action can be one of the following values:

    0   Migrate one or more tables of application group data to separate table spaces. Specify the application group to migrate with the **-g** parameter. Optionally specify a table to migrate with the **-t** parameter. A table must be closed before you can migrate it to a table space.

    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.

4　　　　Create a new table if there are no existing open tables. When you use this value, you must specify the **-g** parameter.

-d dir　Forces the ARSTBLSP program to export the existing application group data to disk. If you specify this parameter, then you must name a directory or file system that contains sufficient free space to hold a copy of the existing application group table. Depending on the size of the table that you are migrating, this can be a significant amount of disk space. For example, a table that contains ten million rows, where each row is one hundred bytes, may require up to 2 GB of disk space to complete the migration.

Note: Unless you specify otherwise, on UNIX servers, Content Manager OnDemand uses named pipes to export the existing application data, which provides the best performance with no file system overhead. IBM recommends that you use the named pipes method. On Windows servers, the system uses a temporary directory to export data; you can specify the temporary directory by using the Content Manager OnDemand configurator program.

-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 (UNIX servers; the registry for Windows servers), 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 that the system on which the instance that you want to process is running.

Note: If running multiple instances of Content Manager OnDemand on the same system, 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 Content Manager OnDemand. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for information about configuring instances.

-t table
When used with the **-a1** parameter, identifies the table to close. When used

with the **-a0** parameter, optionally identifies the table of application group data to migrate. If you do not specify this parameter when you use the **-a0** option, the ARSTBLSP program migrates all tables for the specified application group to separate table spaces.

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 specified table for a specific application group:

   ```
   arstblsp -a 1 -g credit -t AAA26
   ```

5. The following shows how to migrate all tables for a specific application group. The ARSTBLSP program migrates each table in the application group to a separate table space:

   ```
   arstblsp -a 0 -g credit
   ```

6. The following shows how to migrate a specific table of application group data:

   ```
   arstblsp -a 0 -g credit -t AAA26
   ```

Notes

When migrating application group data to table spaces, you should do so at a time when there is little or no other activity on the system.

Do not load data into the system while you are migrating tables with the ARSTBLSP program.

Before you can migrate application group data to table spaces, all of the tables in the application group must be closed. You can use the ARSTBLSP program command with the **-a1** parameter to close a table.

Before you begin to migrate application group data to table spaces, IBM recommends that you create a backup image of the Content Manager OnDemand database.

If you plan to use Tivoli Storage Manager to maintain DB2 archived log files and backup image files, then you must define the Tivoli Storage Manager storage objects before you begin migrating application group data to table spaces. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for assistance with configuring Tivoli Storage Manager.

Files

/usr/lpp/ars/bin/arstblsp
> The AIX executable program.

/opt/ondemand/bin/arstblsp
>> The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arstblsp
>> The Windows executable program.

ARSVIEW

Purpose

The ARSVIEW program processes an AFP document and creates a line data file suitable for viewing on a 3270 display or other line-mode terminal.

Syntax

```
►►──arsview─────┬───────────┬──────┬──────────────┬──┬──────────────┬───────►
                │    ┌─18─┐  │      └─-f──fontDir─┘  │     ┌─11.0──┐  │
                └─-c─┴─cpi┴──┘                       └─-h──┴─height┴──┘

►──────┬───────────┬──┬────────┬──┬──────────────────┬──┬─────────────┬────►
       │   ┌─12─┐  │  │  ┌─0──┐│  └─-r──resourceFile──┘  └─-t──traceFile┘
       └─-l─┴─lpi┴──┘  └─-o─┼─90─┤
                           ├─180─┤
                           └─270─┘

►──────┬─────────────┬──┬────┬──┬───────────┬──────inputFile────────────────►◄
       │    ┌─8.5──┐  │  └─-x─┘  └─-y──csra──┘
       └─-w─┴─width─┴──┘
```

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 ARSVIEW 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 ARSVIEW program can also be used as a general purpose AFP to line data batch conversion utility.

You can invoke the ARSVIEW program 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 ARSVIEW 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 ARSVIEW 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 ARSVIEW 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 ARSVIEW program can process information contained in the form definition, such as the offset, and include text from an overlay in the output file.

-t traceFile

> Specifies the name of the file in which the ARSVIEW program saves debugging information. You can use this option to obtain detailed information about how the ARSVIEW program converts the input 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 csra   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

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:

   ```
   arsview -c 12 -l 8 -o 90 37271600.AFP > 37271600.LIN
   ```

2. The following shows how to process an input file using the default options for characters per inch, lines per inch, width, height, and orientation. The output is redirected to stdout:

```
arsview 96089900.AFP
```

Files

/usr/lpp/ars/bin/arsview
   The AIX executable program.

/opt/ondemand/bin/arsview
   The HP-UX and Sun Solaris executable program.

\Program Files\IBM\OnDemand for Windows\bin\arsview
   The Windows executable program.

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

Syntax for adding, updating, and deleting administrative objects in an Content Manager OnDemand system

```
►►──arsxml──┬─add────┬──┬──────────────┬──────────────────────────►
            ├─update─┤  └─-d──┬───────────┐
            └─delete─┘        └─directory─┘

►──┬────────────────────────────────────┬──-h──hostname──-i──input_xml_file──►
   └─-e──┬─────────────────┬──┬────┐
         └─error_handling──┤  ├─a─┤
                           │  ├─c─┤
                           │  └─u─┘

►──┬───────────────┬──┬──────────────┬──┬────┬──┬────┬──┬──────────────────┬──►
   └─-p──┬──────────┐  └─-u──userid──┘  └─-v─┘  └─-x─┘  └─-1──trace_file──┘
         └─password─┘

►──┬─────────────────┬──────────────────────────────────────────────────────►◄
   └─-2──level──┘
```
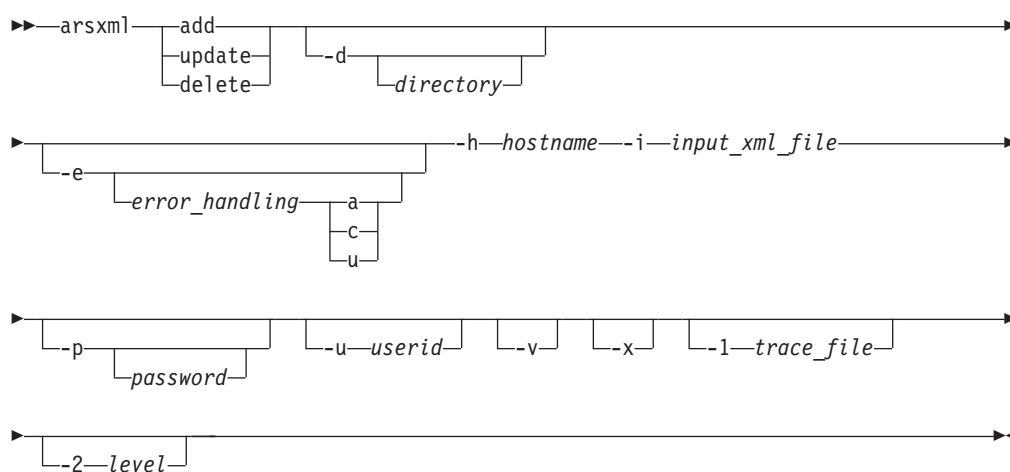
Syntax for exporting existing Content Manager OnDemand objects into a Content Manager OnDemand XML file

```
►►──arsxml──export──┬──────────────────┬──┬──────────────────────┬──────────►
                    └─-d──┬───────────┐  └─-e──┬─────────────────┐
                          └─directory─┘        └─error_handling──┘

►──-h──hostname──┬──────────────────────┬──┬────────────────────────┬──────►
                 └─-i──┬─────────────────┐  └─-o──┬─────────────────┐
                       └─input_xml_file──┘        └─output_xml_file─┘

►──┬───────────────┬──┬──────────┬──┬──────────────┬──┬────┬──────────────────►
   └─-p──┬──────────┐  └─-r──┬───────┐  └─-u──userid──┘  └─-v─┘
         └─password─┘        └─range─┘

►──┬─────────────────┬──┬────┬──┬──────────────┬──┬──────────────────┬──────►
   └─-w──┬──────────┐  └─-x─┘  └─-y──┬───────────┐  └─-1──trace_file──┘
         └─encoding─┘               └─directory─┘

►──┬─────────────────┬──────────────────────────────────────────────────────►◄
   └─-2──level──┘
```
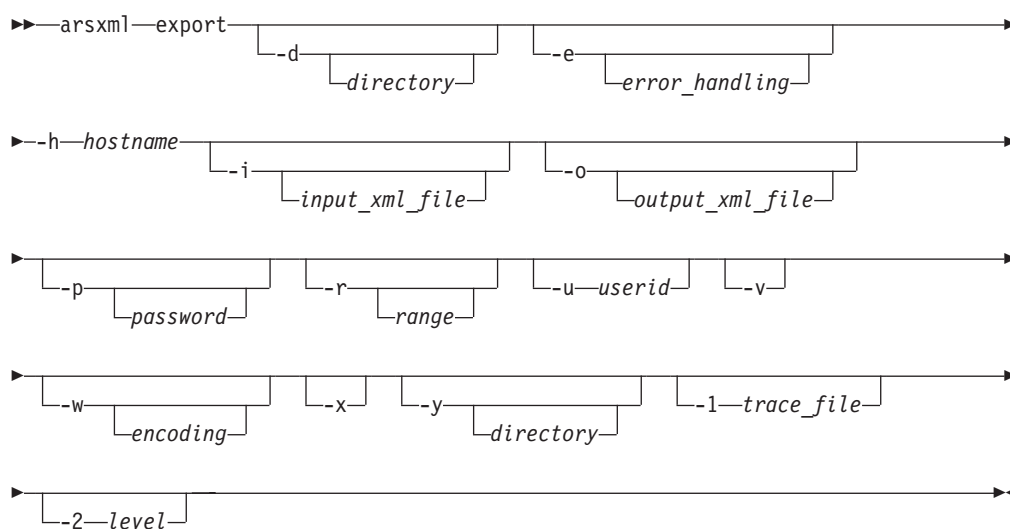
Syntax for validating the schema of the input XML file

```
►►──arsxml──validate──-i──file──────────────────────────────────►◄
                            └─-d─┬──────────┬─┘
                                 └─directory─┘
```

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.

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 OnDemand XML schema file that is shipped with the product. Objects must appear in this order:
  1. Users
  2. Groups
  3. Printers
  4. Storage sets
  5. Application groups
  6. Applications
  7. Folders
  8. Cabinets,
  9. Holds
- 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 "Objects and data model used in the OnDemand XML file" on page 351.
- The ARSXML validate function validates the input XML file schema only. 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>
...
```

For more information on the use of the "task" attribute during an update see "Creating an XML file" on page 198.

Restriction: Do not use the update action to synchronize objects between servers. For example, exporting an object definition from one server followed by an update action of the object definition on another server is not supported.

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 an 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.

-h hostname
Specifies the host name of the Content Manager OnDemand system

-i   Specifies the input XML file.

-p password

The password is optional. If the **-u** parameter is used but the **-p** parameter is omitted, you are prompted for the password before the command is processed. If both the **-u** and the **-p** parameters are omitted, 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 it is omitted, you are prompted for it before the command is processed. If both the **-u** and the **-p** parameters are omitted, the system 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 process. Any other response terminates the process. If you use this option, you do not get the prompt during an operation.

-1 trace_file

Specify the fully-qualified trace file name.

-2 level

Specify the trace level number. These values are additive. The default value is 3.

1   Errors

2   Warnings

4   Info

8   Flow

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.

-h hostname
Specifies the host name of the Content Manager OnDemand system

-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.

-p password
The password is optional. If the **-u** parameter is used but the **-p** parameter is omitted, you are prompted for the password before the command is processed.

-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 no range option is specified, then only the specified object or objects are exported.

-u userid
The user ID is optional. If it is omitted, you are prompted for it before the command is processed.

-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.

-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 the fully-qualified trace file name.

-2 level
Specify the trace level number. These values are additive. The default value is 3.

    1   Errors

    2   Warnings

    4   Info

    8   Flow

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"/>
<user name="Brie" phone="(212) 555-0301"/>
</onDemand>
```

Example 3: 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 4: 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 5: 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 6: Validating input XML file schema

A user wants 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"/>
<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
```

Appendix. Appendixes to the Administration Guide

Mastering CDs from the OnDemand client

About this task

You can extract data from an OnDemand server, transfer the documents to a staging driver, and then use a CD-ROM authoring software to burn the staging drive to a CDR.

When you use the CD-ROM mastering option, selected data is transferred from the server to the client, and then to a staging drive on the client PC. The data transfer is repeated for additional folders if necessary.

Prerequisite: You need to activate the client data distribution for the CD-ROM mastering option to be available in the OnDemand client.

Mastering CDs from the OnDemand client is a two-step process:

Procedure

1. Transfer documents from an OnDemand server to a staging drive.
2. Burn the CD image to the CD-ROM

Results

"Activating the client data distribution for the CD-ROM mastering option"

"Transferring documents from an OnDemand server to a staging drive" on page 318

"Burning the CD image to the CD-ROM" on page 319

Activating the client data distribution for the CD-ROM mastering option
About this task

To activate the client data distribution, copy `arsmstr.ini` from the OnDemand server or from the temporary installation directory that is created by `arssetup`. You can download `arsmstr.ini` from `/usr/lpp/ars/client/windows/win32/ars32` or copy from `0dwin32.zip\ars32`, and copy the file to `x:\program files\ibm\ondemand32\arsmstr.ini`.DLL and DFN files are handled by `arssetup`. After all the files are in their correct locations, the CD-ROM mastering option is

available in the OnDemand client on the File menu.

Transferring documents from an OnDemand server to a staging drive

About this task

Follow these steps to use the CD-ROM mastering option:

Procedure

1. Launch the OnDemand client, and select **File —> Set CD-ROM Mastering Options**. The Set CD-ROM Mastering Options window opens.
2. Under CD-ROM User, enter your user ID and password to the CD-ROM, which is an OnDemand server. The default user ID and password are both cdrom. **Optional:** select the staging drive. By default, the staging drive is d:.

   Important:
   - If the **Exclude Notes**® check box is checked, public exportable annotations are not copied even if they are present.
   - Only local hard drives are available as staging drives.
   - You cannot select more than 1 copy.
3. Click **OK**. The CD-ROM Mastering option becomes active on the File menu.
4. After you run a search and get a list of documents, select **File —> CD-ROM Mastering...**. The CD-ROM Mastering window opens.

   Important:
   - The CD-ROM Mastering option is only available when a search results list is displayed. If a document in the list is being viewed, the CD-ROM Mastering option is grayed out.
5. In the CD-ROM Mastering window, select the folder that you want to add to CD-ROM. Your current folder is automatically selected, and you can select alternate folders from the drop-down list. Provide a description in the CD-ROM folder description field. This description is required, and will be saved in the drop-down list for user in other mastering sessions. You can update the CD-ROM mastering options by choosing the **Set Options** button.

   Important:
   - To transfer the documents to a staging drive, you must keep the search results list on the screen. Only one folder can be staged at a time.
   - All items in the search results list will be placed on the CD-ROM.
6. Click **OK** after you set the options. The CD-ROM mastering process starts. You should be able to see the a window with five options in it:

   Clean   Removes all files on the stage drive

   Setup   Creates the necessary directory structures on the staging drive

   Fetch   Retrieves the data and resources for the items in the hit list

   Index   Re-indexes the retrieved data for the CD-ROM

   Stage   Copies the CD-ROM installation files and the OnDemand GUI (along with any installed languages) to the staging drive
7. After the CD-ROM mastering process finishes, you receive this message:
   `Do you wish to add another folder?`

Click **No** to finish the process. Click **Yes** to return to the previous screen to select another folder.

Results

After the CD-ROM image is completed, OnDemand writes an entry into the system log indicating the contents of the CD-ROM image. This log entry contains the user ID, password, and a complete listing of all the hits that are included in the image. The following is an example system log:

```
CD-ROM Volume AOD00000094

Produced on Saturday, August 22, 1998 at 14:50:41 MDT by CJLEWIS

COPIES      0

    USER   cdrom
PASSWORD   cdrom

FOLDER    Customer Information - Testing...

Account             Customer Name           Document              Date
--------------------------------------------------------------------------------
000-000-152   A & T PIANO                    Customer Letters      11/15/94
000-000-152   A & T PIANO                    Credit Card Statement 10/20/94
000-000-152   A & T PIANO                    Customer Reports      11/30/94
000-000-152   A & T PIANO                    Company Letters       11/21/94
000-000-152   A & T PIANO                    Company Letters - PDF 11/15/94
```

Burning the CD image to the CD-ROM
About this task

Use a CD-ROM authoring software to burn the staging drive to a CDR. The staging drive is as specified in the `arsmstr32.ini`. Popular CD-ROM authoring software includes Roxio Ez-CD Creator, Nero, and Stomp.

Important: Any issue from burning the staging drive to CD is not supported by IBM.

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 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.

Fully accessible alternatives for the line data viewer and the AFP plugin

To perform functions of the line data viewer and the AFP plugin, use the client or
administrative client. At this time, functions of the line data viewer and the AFP
plugin are not accessible through the Web administrator.

Accessible documentation

Documentation for the 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.

ARSSOCKD stop command

IBM provides a command to stop the server process on UNIX servers. The syntax is:

```
arssockd stop instance
```

Where *instance* is the name of the OnDemand instance to stop. Instance names are listed in the ARS.INI file. The default instance name is `archive`.

Note:

1. When you stop the server process, all users who are connected over the network to the OnDemand system are disconnected. Therefore, it is a good idea to warn connected users before stopping the server process.
2. The ARSSOCKD STOP command is for UNIX servers only. For Windows servers, the server process runs as a service. The service can be stopped by using the Services administrative tool in Windows or the OnDemand configurator client.
3. In the temp directory of your system, there is a file named `.instance_name.pid`, for example, `.ARCHIVE.pid`. This file contains the process ID of the service that needs to be stopped. If this file is deleted, the `arssockd stop` command will not succeed. The `.instance_name.pid` is hidden, but sometimes might be deleted by temp directory cleanup tools.

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 -cp .\arssupport.jar arssupport parameters
```

Prerequisites

* Ensure that you have Java runtime version 1.4.2 or higher to run this program.
* Ensure that you are logged on to the operating system using an ID that has the administrator authority on Windows or root authority on UNIX.
* On Windows systems, run ARSSUPPORT from the Content Manager OnDemand 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

```
►►──arssupport──┬──────┬──┬──────────────────────┬──┬──────┬──┬─────────────┬──►
                └──-h──┘  └──-I──instance name──┘  └──-l──┘  └──-m──minutes──┘

►──┬───────────────────────┬──┬──────────────────┬──┬────────────────┬──┬──────┬──►◄
   └──-o──outputpath──┘       └──-p──password──┘     └──-u──userid──┘    └──-v──┘
```

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 and -p parameters.

-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. If the -l parameter is specified, the -p parameter is required.

-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 arssupport -h

java arssupport -o c:\pmr11256 -l -u admin -p password

java arssupport -o . -I instance -l -m 28 -u admin -p password
```

Monitoring servers

System logging facility

Content Manager OnDemand provides a logging facility to help administrators track Content Manager OnDemand activity and monitor the system. When you enable logging for system events, user events, and application group events, Content Manager OnDemand stores the messages that are generated by the various Content Manager OnDemand programs in the system log. You can use one of the Content Manager OnDemand client programs to search for and filter messages by time stamp, severity, message number, and user name.

Note: Before you start Content Manager OnDemand for the first time, you must initialize the system logging facility. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for information about initializing the system logging facility.

Searching for and viewing messages

To search for and view the messages that are stored in the system logging facility, log on to Content Manager OnDemand with an Content Manager OnDemand client program and open the System Log folder. Enter search criteria in one or more of the search fields. When you choose the **Search** command, Content Manager OnDemand retrieves the messages from the database that match the search criteria that you specified.

You can specify a date and time value to search for and retrieve messages. You can also specify other search criteria, such as:

Log Id

Each time that a client logs on to the server, Content Manager OnDemand assigns a transaction number to that instance of the client program. All messages that are generated by that instance of the client program include the same Log Id.

Userid

The Content Manager OnDemand userid

Severity

Content Manager OnDemand assigns a severity to each message: Alert, Error, Warning, Info, and Debug

View Depending on the type of message in the system log, you may be able to view other information that is related to or associated with the message. For example:

- You can display the message log that was generated during a load process by selecting an ARSLOAD message and then choosing the **View all Selected** command.

  While most processes do not generate other information that can be stored in the system log, you could provide a user-written program to process the messages and generate your own information about the events. For example, you could provide a user-written program to generate a report that lists the number of users that are logged on to the system in thirty minute increments. Content Manager OnDemand provides a system log user exit point so that you can process any message that is stored in the system log and take the action that you require. See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about the system log user exit.

- Other messages in the system log do not provide additional records or other data associated with an event. For example, the Logon and Logoff events each generate a single message, with no additional information that you can view.

Msg Num

The message number that is assigned by Content Manager OnDemand

Message

The text of the message that Content Manager OnDemand uses to restrict a search. For example, if you type **Login**, Content Manager OnDemand searches for and displays the messages issued by the Logon to a Server command.

System log user exit

When you enable logging for system, user, and application group events, Content Manager OnDemand sends a copy of each message that is generated by the system to the system log user exit program. The system log user exit program on UNIX servers is named `arslog` and resides in the Content Manager OnDemand executable directory (`bin`). The system log user exit program on Windows servers is named ARSLOG.BAT.

The system log user exit program that is provided by IBM does not perform any functions. However, you can replace the program that is provided by IBM with a user-written program that does user-defined processing. For example, you could provide a user-written program to check for certain message numbers or severity, and take whatever action you deem appropriate.

You configure the system to send messages to the system log user exit by selecting User Exit Logging options from the System Parameters dialog box. See the online help for the administrative client for more information about the User Exit Logging options on the System Parameters dialog box.

See *IBM Content Manager OnDemand for Multiplatforms: Installation and Configuration Guide* for more information about the system log user exit.

Sample ARSLOG user exit script for UNIX

Figure 47 shows the sample ARSLOG user exit script for UNIX that is provided by IBM.

```
# $1 - OnDemand Instance Name
# $2 - Time Stamp
# $3 - Log Identifier
# $4 - Userid
# $5 - Account
# $6 - Severity
# $7 - Message Number
# $8 - Message Text
# $9 - Document File
#

echo "$@" >> ${ARS_TMP}/syslog.log

if [ -n "$9" ];then
   if [ -f "$9" ];then
      print "Copy log doc $9\n" >> ${ARS_TMP}/syslog.log
      cp $9 /tmp/syslogdocs/$(basename $9).doc 2>> ${ARS_TMP}/syslog.log
   else
      print "$9 does not exist\n" >> ${ARS_TMP}/syslog.log
   fi
fi

exit 0
```

Figure 47. Sample ARSLOG user exit script for UNIX

Sample ARSLOG user exit batch file for Windows

Figure 48 on page 325 shows the sample ARSLOG user exit batch file for Windows that is provided by IBM.

```
REM %1 - OnDemand Instance Name
REM %2 - Time Stamp
REM %3 - Log Identifier
REM %4 - Userid
REM %5 - Account
REM %6 - Severity
REM %7 - Message Number
REM %8 - Message Text
REM %9 - Document File
REM

ECHO %1 %2 %3 %4 %5 %6 %7 %8 %9>%ARS_TMP%\System.log

REM make sure the %ARS_TMP%\Syslog directory exists
IF EXIST %9 COPY %9 %ARS_TMP%\Syslog

EXIT
```

Figure 48. Sample ARSLOG user exit batch file for Windows

Monitoring users

The Content Manager OnDemand server generates system log messages to help you track the number of users that are logged on to the server:

- Content Manager OnDemand stores message number 201 in the system log every thirty minutes. This message contains the current number of users that are logged on to the server.
- Content Manager OnDemand stores message number 202 in the system log every time that the number of concurrent users exceeds the previous maximum number of concurrent users. The number of concurrent users is reset each time that you restart the Content Manager OnDemand server processes.

Reviewing a CD-ROM creation manifest

You can open the System Log folder to review a manifest created by the Content Manager OnDemand ad-hoc CD mastering software. A CD creation manifest contains a cover page with control information about the CD and one or more pages that list the items retrieved from the database and written on the CD.

To review a CD creation manifest:

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

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

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

System load logging facility

Content Manager OnDemand provides a logging facility to help you track loading activity. After you enable system load logging, Content Manager OnDemand stores the messages that are generated by the load programs in the system load log. You can use one of the Content Manager OnDemand client programs to search for and filter messages by criteria such as load date, application group name, application name, load ID, and input file name.

Before you start Content Manager OnDemand for the first time, make sure that you initialize the system loading logging facility. See *Installation and Configuration Guide* for information about initializing the system logging facility.

Monitoring performance

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

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

You can also monitor applications, such as DB2 and Tivoli Storage Manager.

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

As part of the initial setup of your system, IBM recommends that you create a measurement baseline of information for use in analyzing system performance. This is important because to determine if a resource is being overused, under used, or is at maximum capacity and performance, you need to identify what is normal. You can also use this information to set expectations of system performance for users.

Automating the collection of server data is essential to control the amount of data collected and to facilitate its analysis. You can use tools provided with the

operating system, such as Performance Monitor and Network Monitor on Windows servers, to automatically capture server data during specified periods and then store the data in a database.

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

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

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

For more information about Performance Monitor, see your Windows operating system documentation.

Configuration changes for Content Manager OnDemand client applications in an FDCC environment

This section provides guidance on how to configure the Content Manager OnDemand client applications on Windows XP and Windows Vista in a Federal Desktop Core Configuration (FDCC) environment.

Federal Desktop Core Configuration (FDCC) on Windows XP and Windows Vista is a mandate issued by United States Federal government. The U.S. Executive Office of the President issued policy memorandum M-07-18 and policy memorandum M-07-11 requesting that all government agencies set up Windows XP and Windows Vista with the security settings that are defined in the FDCC. All agencies must comply with the new standard by February 1, 2008. The U.S. government pays high attention to the security and vulnerability of its IT infrastructure through the enforcement of the FDCC requirement.

Ensure that you have the administrator privileges on the FDCC system before you proceed with the instructions in this section.

Windows client, Windows administration client

1. Before you use the Windows client or the Windows administration client, add the library server to the FDCC system.
2. If the Windows client and the Windows administration client are installed in the Program Files directory (the default), then you need only read access to the Program Files directory.

3. Unless you specify otherwise, temporary user data are written to the `C:\Documents and Settings\<userid>\Application Data\IBM\OnDemand32` directory.

AFP Web Viewer, Image Web Viewer

1. In the Group Policy Editor of your system, make the following changes for **Local Computer Policy** → **Computer Configuration** → **Administrative Templates** → **Windows Components** → **Internet Explorer**:
   - For **Security Zones: Do not allow users to change policies**, change from **Enabled** to **Disabled**.
   - For **Security Zones: Do not allow users to add/delete sites**, change from **Enabled** to **Disabled**.

   On Windows Vista only, make the following change for **Local Computer Policy** → **Computer Configuration** → **Administrative Templates** → **Internet Explorer**:
   - For **Security Zones: Use only machine settings**, change from **Enabled** to **Disabled**. To ensure that users can add a site to the trusted sites, after this change, ask each user on the operating system to add all the sites of your Content Manager OnDemand servers into the trusted sites.

   Make the following changes for **Local Computer Policy** → **Computer Configuration** → **Administrative Templates** → **Windows Components** → **Internet Explorer** → **Internet Control Panel** → **Security Page** → **Trusted Sites Zone**:
   - For **Run ActiveX controls and plugins**, change from **Disabled** to **Prompt**.
   - Restart your system for these changes to take effect.
   - Add the Content Manager OnDemand server URL to the trusted sites of your Internet Explorer browser.

   Make the following changes for **Local Computer Policy** → **Computer Configuration** → **Administrative Templates** → **Windows Components** → **Internet Explorer**:
   - For **Security Zones: Do not allow users to change policies**, change from **Disabled** to **Enabled**.
   - For **Security Zones: Do not allow users to add/delete sites**, change from **Disabled** to **Enabled**.

2. If you use Internet Explorer Version 7, then enable the Federal Information Processing Standard 140 (FIPS-140) on the WebSphere® Application Server where your Content Manager OnDemand server is deployed into the server operating system.

Java Line Data Viewer Version 2 applet

1. Before you access the Java Line Data Viewer in a browser, install the Java Runtime Environment (JRE) plug-in.

2. If you are on Windows Vista, in the Group Policy Editor of your system, make the following change for **Local Computer Policy** → **Computer Configuration** → **Administrative Templates** → **Windows Components** → **Internet Explorer**:
   - For **Security Zones: Use only machine settings**, change from **Enabled** to **Disabled**. To ensure that users can add a site to the trusted sites, after this change, each user on the operating system must add all the sites of your Content Manager OnDemand servers into the trusted sites.

   Make the following changes for **Local Computer Policy** → **Computer Configuration** → **Administrative Templates** → **Windows Components** → **Internet Explorer** → **Internet Control Panel** → **Security Page** → **Trusted Sites Zone**:
   - For **Java permissions**, change from **Disabled** to **High safety**.

- For **Run ActiveX controls and plugins**, change from **Disabled** to **Prompt**.
- Restart your system for these changes to take effect.
- Add the Content Manager OnDemand server URL to the trusted sites of your Internet Explorer browser.

3. If you use Internet Explorer Version 7, enable the Federal Information Processing Standard 140 (FIPS-140) on the WebSphere Application Server where your Content Manager OnDemand server is deployed into the server operating system.

After you complete these changes, a limited Windows desktop client user can access all the functionality of the ODWEK web viewers. One exception is the Java Line Data Viewer Version 2 applet. Currently, Internet Explorer settings does not allow the Java Runtime Environment to load the applet. Other available browsers do not have this restriction and allow the applet to operate correctly.

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)

This table maintains a catalog of application groups.

Table 14. 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	Segment type (Database Organization)
ann_type	CHAR(1) FOR BIT DATA	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
expire_type	CHAR(1) FOR BIT DATA	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 (DB2, SQL Server, Oracle)	Date and time application group was last updated
last_doc_date	BIGINT (DB2, SQL Server, Oracle)	Migrate last document date
migr_srvr_str	VARCHAR(254)	Migrate server string

Table 14. Application group table (ARSAG) (continued)

Column Name	Data Type	Description
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

Table 15. Indexes for application group table (ARSAG)

Index name	Columns in index	Type of index
arsag_name_idx	name	unique index
arsag_agid_idx	agid	unique cluster 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 16. 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 17. 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

Application group field table (ARSAGFLD)

This table maintains a catalog of application group fields.

Table 18. 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	Application group field type

Table 18. Application group field table (ARSAGFLD) (continued)

Column Name	Data Type	Description
qual	CHAR(1) FOR BIT DATA	Application group field type qualifier
mask	INTEGER	Application group field mask
string_type	CHAR(1) FOR BIT DATA	Type of string field
string_len	SMALLINT	Length of string field
string_case	CHAR(1) FOR BIT DATA	Case of string field
leading	VARCHAR(8)	Leading characters to trim from string field
strip	VARCHAR(8)	Any characters to trim from string field
trailing	VARCHAR(8)	Trailing characters to trim from string field

Table 19. Indexes for application group field table (ARSAGFLD)

Index name	Columns in index	Type of index
arsagfld_idx	agid, field	unique cluster index

Application group field alias table (ARSAGFLDALIAS)

This table maintains a catalog of application group field aliases (Database values and Displayed values).

Table 20. 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 21. Indexes for application group field alias table (ARSAGFLDALIAS)

Index name	Columns in index	Type of index
arsagfldalias_idx	agid, field	cluster index

Application group composite index table (ARSAGINDEX)

This table maintains application group composite indexes on application group fields.

Table 22. 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

Table 22. Application group composite index table (ARSAGINDEX) (continued)

Column Name	Data Type	Description
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	Specifies that index entries should be kept in ascending or descending order of the column values

Table 23. Indexes for application group composite index table (ARSAGINDEX)

Index name	Columns in index	Type of index
arsagindex_idx	agid	cluster index

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 24. 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	LONG VARCHAR	Query restriction; user-defined and validated SQL string

Table 25. Indexes for application group permissions table (ARSAGPERMS)

Index name	Columns in index	Type of index
arsagperms_idx	agid, id	unique cluster index

Annotation table (ARSANN)

This table maintains a catalog of annotations.

Table 26. Annotation table (ARSANN)

Column Name	Data Type	Description
id	INTEGER	User identifier
userid	VARCHAR(128)	User name
agid	INTEGER	Application group identifier

Table 26. Annotation table (ARSANN) (continued)

Column Name	Data Type	Description
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 (DB2, SQL Server, Oracle)	Time annotation was created
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	LONG VARCHAR	The text of the annotation
image_buf	BLOB(1048576)	Any images that are attached to the annotation
table_name	VARCHAR(18)	Segment in which annotation exists
doc_exp_date	BIGINT (DB2, SQL Server, Oracle)	Expiration date of document

Table 27. 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

Application table (ARSAPP)

This table maintains a catalog of applications.

Table 28. 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	Document type
doc_comp_type	CHAR(1) FOR BIT DATA	Default document compression type
res_comp_type	CHAR(1) FOR BIT DATA	Default resource compression type
idx_type	CHAR(1) FOR BIT DATA	Indexer
comp_obj_size	INTEGER	Compressed object size

Table 28. Application table (ARSAPP) (continued)

Column Name	Data Type	Description
pid	INTEGER	Default printer identifier
alias	VARCHAR(254)	Application identifier within application group
fixed_view	LONG VARCHAR	Default and fixed logical view information
indexer	LONG VARCHAR	Indexing parameters
preprocessor	LONG VARCHAR	Parameters used to preprocess rows before loading
upd_userid	VARCHAR(128)	Userid of last user to update application
upd_date	BIGINT (DB2, SQL Server, Oracle)	Date and time application was last updated
db_exp_date	INTEGER	Expiration date
appl_type	INTEGER	Application type

Table 29. Indexes for application table (ARSAPP)

Index name	Columns in index	Type of index
arsapp_idx	agid, aid, name	Unique cluster index

User logical views table (ARSAPPUSR)

This table maintains a catalog of logical views.

Table 30. User logical views table (ARSAPPUSR)

Column Name	Data Type	Description
agid	INTEGER	Application group identifier
aid	INTEGER	Application identifier
id	INTEGER	User identifier
user_view	LONG VARCHAR	Logical view information

Table 31. Indexes for user logical views table (ARSAPPUSR)

Index name	Columns in index	Type of index
arsappusr_idx	agid, aid, id	Unique cluster index

Cabinet table (ARSCAB)

This table maintains a catalog of cabinet definitions.

Table 32. Cabinet table (ARSCAB)

Column Name	Data Type	Description
name	VARCHAR(60)	Cabinet name
description	VARCHAR(120)	Cabinet description
cid	INTEGER	Cabinet identifier

Table 32. Cabinet table (ARSCAB) (continued)

Column Name	Data Type	Description
upd_userid	VARCHAR(128)	Userid of last user to update the cabinet
upd_date	BIGINT	Date and time cabinet was last updated

Table 33. 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

Cabinet to folder mapping table (ARSCAB2FOL)

This table maintains a catalog of mappings between cabinets and folders.

Table 34. Cabinet to folder mapping table (ARSCAB2FOL)

Column Name	Data Type	Description
cid	INTEGER	Cabinet identifier
fid	INTEGER	Folder identifier

Table 35. Indexes for cabinet to folder mapping table (ARSCAB2FOL)

Index name	Columns in index	Type of index
arscab2fol_idx	cid, fid	Unique cluster index

Cabinet permissions table (ARSCABPERMS)

This table maintains permissions specifications for cabinets.

Table 36. Cabinet permissions table (ARSCABPERMS)

Column Name	Data Type	Description
id	INTEGER	User identifier
cid	INTEGER	Cabinet identifier
perms	INTEGER	User permissions for the cabinet

Table 37. Indexes for cabinet permissions table (ARSCABPERMS)

Index name	Columns in index	Type of index
arscabperms_idx	cid, id	Unique cluster index

CFS-OD work table (ARSCFSODWORK)

This table is a catalog of work between Content Manager OnDemand and CFS-OD.

Table 38. CFS-OD 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	Timestamp of work
misc_comp_type	CHAR(1) FOR BIT DATA	Compression type of misc
misc_info	BLOB	Work information

Table 39. Indexes for CFS-OD work table

Index name	Columns in index	Type of index
arscfsodwork_idx	agid, name	Cluster index

Folder table (ARSFOL)

This table maintains a catalog of folders.

Table 40. 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	Annotation display
upd_userid	VARCHAR(128)	Userid of last user to update folder
upd_date	BIGINT (DB2, SQL Server, Oracle)	Date and time folder was last updated

Table 41. 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

Folder field table (ARSFOLFLD)

This table maintains a catalog of folder fields.

Table 42. Folder field table (ARSFOLFLD)

Column Name	Data Type	Description
fid	INTEGER	Folder identifier
field	SMALLINT	Folder field identifier

Table 42. Folder field table (ARSFOLFLD) (continued)

Column Name	Data Type	Description
name	VARCHAR(60)	Folder field name
description	VARCHAR(120)	Folder field description
type	CHAR(1) FOR BIT DATA	Folder field type
qual	CHAR(1) FOR BIT DATA	Folder field type qualifier
map_type	SMALLINT	Type of field mapping

Table 43. Indexes for folder field table (ARSFOLFLD)

Index name	Columns in index	Type of index
arsfolfid_idx	fid, field	Unique cluster index

Folder user fields table (ARSFOLFLDUSR)

This table maintains a catalog of folder fields for users that have specific field
information for a folder.

Table 44. 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	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 (DB2, SQL Server, Oracle)	Smallest integer allowed
max_int	BIGINT (DB2, SQL Server, Oracle)	Largest integer allowed
datefmt	VARCHAR(60)	Format of date strings in search criteria
datedef_type	CHAR(1) FOR BIT DATA	Default date units for interval
datedef_arg	INTEGER	Default date value for interval
min_float	FLOAT	Smallest decimal allowed
max_float	FLOAT	Largest decimal allowed

Table 44. Folder user fields table (ARSFOLFLDUSR) (continued)

Column Name	Data Type	Description
decimal_pt_pos	SMALLINT	Precision (right of decimal point)
string_wild	CHAR(1) FOR BIT DATA	Wildcard
dspl_datefmt	VARCHAR(60)	Format of date strings in document list

Table 45. Indexes for folder user fields table (ARSFOLFLDUSR)

Index name	Columns in index	Type of index
arsfolfldusr_idx	fid, id, field	Unique cluster index

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 46. 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 47. Indexes for folder permissions table (ARSFOLPERMS)

Index name	Columns in index	Type of index
arsfolperms_idx	fid, id	Unique cluster index

Group table (ARSGROUP)

This table maintains a catalog of groups.

Table 48. 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 (DB2, SQL Server, Oracle)	Date and time group was last updated

Table 49. Indexes for group table (ARSGROUP)

Index name	Columns in index	Type of index
arsgroup_gid_idx	gid	Unique cluster index
arsgroup_name_idx	name	Unique index

Hold table (ARSHOLD)

This table is a catalog of holds.

Table 50. 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	Timestamp when the hold was created

Table 51. 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

Hold mapping table (ARSHOLDMAP)

This table maintains a catalog of documents to holds.

Table 52. 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	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

Table 53. Indexes for hold mapping table (ARSHOLDMAP)

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

Hold permissions table (ARSHOLDPERMS)

This table maintains a catalog of hold permissions.

Table 54. Hold permissions table (ARSHOLDPERMS)

Column Name	Data Type	Description
id	INTEGER	User/Group identifier
hid	INTEGER	Hold identifier
perms	INTEGER	Permission

Table 55. Indexes for hold permissions table (ARSHOLDPERMS)

Index name	Columns in index	Type of index
arsholdperms_idx	hid, id	Unique cluster index

Hold work table (ARSHOLDWORK)

This table maintains a catalog of hold work.

Table 56. 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	Timestamp of work

Table 57. Indexes for hold work table (ARSHOLDWORK)

Index name	Columns in index	Type of index
arsholdwork_idx	agid, aid, name	Unique cluster index

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 58. 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

Table 58. Load table (ARSLOAD) (continued)

Column Name	Data Type	Description
start	BIGINT	Start date in segment
stop	BIGINT	Stop date in segment
exp_date	BIGINT	Expiration date
aid	INTEGER	Application ID
total_doc	INTEGER	Number of documents in load

Table 59. Indexes for load table (ARSLOAD)

Index name	Columns in index	Type of index
arsload_idx	agid, exp_date	Cluster index
arsload_name_idx	agid, name	Index

Named query table (ARSNAMEQ)

This table maintains a catalog of named queries.

Table 60. 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	LONG VARCHAR	Named query

Table 61. Indexes for named query table (ARSNAMEQ)

Index name	Columns in index	Type of index
arsnameq_idx	fid, id, name	Unique cluster index

Node table (ARSNODE)

This table maintains a catalog of storage nodes.

Table 62. 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

Table 62. Node table (ARSNODE) (continued)

Column Name	Data Type	Description
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 (DB2, SQL Server, Oracle)	Date and time storage node was last updated

Table 63. Indexes for node table (ARSNODE)

Index name	Columns in index	Type of index
arsnode_idx	sid, nid	Unique cluster index

Printer table (ARSPRT)

This table maintains a catalog of printers.

Table 64. 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	Type of printer
upd_userid	VARCHAR(128)	Userid of last user to update printer
upd_date	BIGINT (DB2, SQL Server, Oracle)	Date and time printer was last updated

Table 65. Indexes for printer table (ARSPRT)

Index name	Columns in index	Type of index
arsprt_name_idx	name	Unique index
arsprt_pid_idx	pid	Unique cluster index

Printer options table (ARSPRTOPTS)

This table maintains a catalog of printing options (parameters) for printers.

Table 66. 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	Type of printer
buf	LONG VARCHAR	Printing options

Table 67. Indexes for printer options table (ARSPRTOPTS)

Index name	Columns in index	Type of index
arsprtopts_idx	id, name	Unique cluster index

Printer user table (ARSPRTUSR)

This table maintains a catalog of user permissions for printers.

Table 68. Printer user table (ARSPRTUSR)

Column Name	Data Type	Description
pid	INTEGER	Printer identifier
id	INTEGER	User or group identifier

Table 69. Indexes for printer user table (ARSPRTUSR)

Index name	Columns in index	Type of index
arsprtusr_idx	pid, id	Unique cluster index

Resources table (ARSRES)

This table maintains a catalog of AFP resources.

Table 70. Resources table (ARSRES)

Column Name	Data Type	Description
rid	INTEGER	Resource identifier
agid	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	Method of compression
add_date	BIGINT	Date that the resources was loaded into the system
comp_size	INTEGER	Compressed size
decomp_size	INTEGER	Decompressed size
exp_date	BIGINT	Expiration date
hold	SMALLINT	Whether or not on hold

Table 71. Indexes for resources table (ARSRES)

Index name	Columns in index	Type of index
arsres_idx	agid, rid	Unique cluster index

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 72. 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 (DB2, SQL Server, Oracle)	Segment start date
stop_date	BIGINT (DB2, SQL Server, Oracle)	Segment stop date
post_date	BIGINT (DB2, SQL Server, Oracle)	Date table was created
closed_date	BIGINT (DB2, SQL Server, Oracle)	Date table was closed
reimported_date	BIGINT (DB2, SQL Server, Oracle)	Date table was imported to the database
last_update	BIGINT (DB2, SQL Server, Oracle)	Date table was last updated
last_backup	BIGINT (DB2, SQL Server, Oracle)	Date of last table backup
last_stats	BIGINT (DB2, SQL Server, Oracle)	Date statistics last collected
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

Table 73. Indexes for segment table (ARSSEG)

Index name	Columns in index	Type of index
arsseg_name_idx	agid, table_name	Unique cluster index
arsseg_idx	agid, start_date, stop_date	Index
arsseg_cdate_idx	agid, closed_date	Index

Storage set table (ARSSET)

This table maintains a catalog of storage sets.

Table 74. 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	Type of load
upd_userid	VARCHAR(128)	Userid of last user to update storage set
upd_date	BIGINT (DB2, SQL Server, Oracle)	Date and time storage set was last updated

Table 75. 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

System parameters table (ARSSYS)

This table contains a single row that specifies the current system parameters.

Table 76. 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	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	LONG VARCHAR	reserved
upd_userid	VARCHAR(128)	Userid of last user to update system parameters
upd_date	BIGINT	Date and time system parameters were last updated

Table 76. System parameters table (ARSSYS) (continued)

Column Name	Data Type	Description
sys_mask2	INTEGER	Types of annotations
min_gid	INTEGER	Group ID starting value

Indexes: The system parameters table is not indexed

User table (ARSUSER)

This table maintains a catalog of users.

Table 77. User table (ARSUSER)

Column Name	Data Type	Description
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	Date password last set
pid	INTEGER	Default printer identifier
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	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	Date and time user was last updated
passwd_exp	INTEGER	Password expiration
uclass	VARCHAR(1)	Printer class value
banner	VARCHAR(1)	Printer banner

Table 77. User table (ARSUSER) (continued)

Column Name	Data Type	Description
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

Table 78. Indexes for user table (ARSUSER)

Index name	Columns in index	Type of index
arsuser_id_idx	uid	Unique cluster index
arsuser_name_idx	userid	Unique index

Users in groups table (ARSUSRGRP)

This table maintains the association of groups with individual users.

Table 79. Users in groups table (ARSUSRGRP)

Column Name	Data Type	Description
gid	INTEGER	Group identifier
uid	INTEGER	User identifier

Table 80. Indexes for users in groups table (ARSUSRGRP)

Index name	Columns in index	Type of index
arsusrgrp_idx	gid, uid	Unique cluster index

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 81. 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 82. Indexes for user group ID table (ARSUSRGRPID)

Index name	Columns in index	Type of index
arsusrgrpid_idx	id, owner_id	Unique cluster index

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 one to 32 user-defined fields followed by approximately 40 bytes of Content Manager OnDemand control information (fields).

Table 83. 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	Annotation flag
comp_type	CHAR(1) FOR BIT DATA	Compression type
resource	INTEGER	Resource identifier
pri_nid	SMALLINT	Primary storage node identifier
sec_nid	SMALLINT	Secondary storage node identifier

Table 84. 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.

Objects and data model used in the OnDemand XML file

XML objects in the OnDemand XML file

You can use an XML interface to import data into an 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 from scratch. This section helps you to understand the objects in the XML file.

The following diagram illustrates the overall layout of the OnDemand XML file.

CM OnDemand
→ user → permission
→ group → user
→ printer → permission
→ storageSet → node
→ applicationGroup → permission
  → field → mapping
  → index → field
→ application → lineData → lvHeader
  → lvField
  → logView
  → afpData → logView
  → imageData → logView
  → pdfData → logView
  → SCSData → lvHeader
  → udData → lvField
  → preprocessParm → logView
  → imageOverlay
  → indexParm
  → SCSExData
→ folder → applicationGroup → application
  → permission
  → field → mapping
  → fieldInfo
→ cabinet → folder
  → permission
→ hold → permission

Figure 49. Overall layout of the OnDemand XML file

The following tables provide detailed descriptions about the objects in the OnDemand XML file. The objects include:

- user
- group
- printer
- storageSet
- applicationGroup
- application
- folder
- cabinet
- hold

If you create an input XML file, please ensure that you include the objects following this order:

1. user
2. group
3. printer
4. storageSet
5. applicationGroup
6. application
7. folder
8. cabinet
9. hold

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.

DataType

The XML data type of the attribute. Types in italics indicates an XML data type that is defined in the 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 children, 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.

User object

Table 85. Attributes for the user object

Name	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the user	*nameString*	Yes			Yes
uid	User ID value	Integer	No	0	• Integer value 1–1,080,000 • 0 to indicate system generated values	No
description	Description of the user	*descString*	No	""		Yes
password	User's password	*passwordString*	No	""		Yes
userType	Type of user	*userTypeString*	No	"User"	• "User" • "User Admin" • "AG/Folder/ Cabinet Admin" • "System Admin" • "Hold Admin"	Yes
timeOut	Time out length in minutes	*timeOutType*	No	"Use System Value"	• "Use System Value" • "No Limit" • Any positive integer value	Yes
fullName	User's full name	*userMiscString*	No	""		Yes
acctInfo	Accounting for the user	*userMiscString*	No	""		Yes
company	Name of the company	*userMiscString*	No	""		Yes
title	User's title of job responsibility	*userMiscString*	No	""		Yes
addr1	User's address	*userMiscString*	No	""		Yes
addr2	Address, line number 2	*userMiscString*	No	""		Yes

Table 85. Attributes for the user object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
addr3	Address, line number 3	*userMiscString*	No	""		Yes
addr4	Address, line number 4	*userMiscString*	No	""		Yes
dept	User's department name or number	*userMiscString*	No	""		Yes
building	User's building name or number	*userMiscString*	No	""		Yes
room	User's office name or number	*userMiscString*	No	""		Yes
phone	User's phone number	*phoneString*	No	""		Yes
fax	User's fax number	*phoneString*	No	""		Yes
email	User's email address	*emailString*	No	""		Yes
coverPage	Default Fax cover page for the user	*userMiscString*	No	""		Yes
printer	User's default printer	*nameString*	No	"NONE"		Yes
createCabinetsAuth	Whether this user or group has authority to create cabinets	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
createFoldersAuth	Whether this user have authority to create folders	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
createHoldsAuth	Whether this user has authority to create holds	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
createUsersAuth	Whether this user has authority to create users	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
createGroupsAuth	Whether this user have authority to create group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 85. Attributes for the user object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
createAppGroupsAuth	Whether this user have authority to create application groups	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
disableUser	Specifies whether this user should be disabled	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
passwordExp	Specifies when the user's password expires	*pwExpValue*	No	"Use System Value"	• "Use System Value" • "Never Expires" • An integer between 1 and 365	Yes
printerClass	The printer class	*char*	No	""		Yes
printerBanner	Specifies whether there should be a printer banner	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
printerDest	The printer destination	*valString*	No	""		Yes
printerNode	The printer node	*valString*	No	""		Yes
printerWriter	The printer writer	*valString*	No	""		Yes
printerPagedef	The printer pagedef	*valString*	No	""		Yes
printerFormdef	The printer formdef	*valString*	No	""		Yes
printerForms	The printer forms	*valString*	No	""		Yes
printerRouting	The printer routing	*routingString*	No	""		Yes

Table 86. Child object under the user object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes

Table 87. 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	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
user	The name of a user that can view and maintain this user	*nameString*	Yes	""	• User name • "_ALL"	No
group	The name of a group that can view and maintain this user	*nameString*	Yes	""	• Group name • "_ALL"	No
adminAuthority	Whether this user or group has authority to update and delete the user	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Note: In the permission child, you must specify either a user or a group object, however, you cannot specify both.

Group object

Table 88. Attributes for the group object

Name	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the group	*nameString*	Yes			Yes
gid	Group ID value	Integer	No	0	• Integer value greater than or equal to 1080001 • 0 to indicate a system value	No
description	Description of this group	*descString*	No	""		Yes
ownerUser	Name of the user that is the owner of this group	*nameString*	No	"NONE"		Yes
ownerGroup	Name of the group that is the owner of this group	*nameString*	No	"NONE"		Yes

Note: You can specify an ownerUser or an ownerGroup attribute, however, you cannot specify both.

Table 89. Child object under the group object

Name	Minimum number	Maximum number	Deleteable	Updateable
user	0	Unbounded	Yes	No

Table 90. 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	*taskString*	No	"add"	• "add" • "delete"	N/A
name	The name of a user that is a member of this group	*nameString*	Yes			No

Printer object

Table 91. Attributes for the printer object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the printer	*nameString*	Yes			Yes
description	A description of the printer	*descString*	No	""		Yes
queue	The printer queue	*queueString*	Yes			Yes
printerType	The type of printer	*printerTypeString*	No	"Printer"	• "Printer" • "Fax" • "Printer With Information"	Yes

Table 92. Child objects under the printer object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	No

Table 93. Attributes for the permission object under the printer object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete"	N/A
user	The user to give permission to	*nameString*	Yes	""	• User name • "_ALL"	No
group	The group to give permission to	*nameString*	Yes	""	• Group name • "*PUBLIC" • "_ALL"	No

Storage set object

Table 94. Attributes for the storageSet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the storage set	*nameString*	Yes			Yes
description	A description of the storage set	*descString*	No	""		Yes
storageType	The load type of the storage set	*storageTypeString*	No	"Fixed"	• "Fixed" • "Local"	Yes

Table 95. 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 96. Attributes for the node object under the storageSet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
name	Name of the storage set node	*nameString*	Yes			No
description	A description of the storage set node	*descString*	No	""		Yes
server	The server where the node resides	*string*	No	"*ONDEMAND"		Yes
logon	Client node name	*nameString*	Dependent	""		Yes
password	Client node password	*passwordString*	Dependent	""		Yes
loadData	Whether this is a load data node	*yesNoString*	No	"No"	• "Yes" • "No"	Yes
accessMethod	Determines the type of archive storage for the primary node	*accessString*	No	"TSM"	• "OAM" • "VSAM" • "TSM" • "Cache"	Yes
reloadHoldData	Whether this node should be used to reload hold data	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Note: The logon and password attributes might be required based on the type of node.

Application group object

Table 97. Attributes for the applicationGroup object

Name	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the application group	*nameString*	Yes			Yes
database	Name of the database	*nameString*	No	""		Yes
description	Description of the application group	*descString*	No	""		Yes
storageSet	Name of the storage set to use for this application group	*nameString*	No	""		Yes
cacheDataLen	How many days to cache the data for	*cacheDataLenValue*	No	90	• "No Cache" • "Search Cache" • A positive integer	Yes
CFSOD	Use CFSOD	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
declareCFSOD	Use to declare CFSOD	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
expirationType	Determines how data is deleted from the application group	*expTypeString*	No	"Load"	• "Load" • "Segment" • "Document" • **z/OS only option:** Storage Manager	No
expirationDate	The number of days that OnDemand keeps documents, resources, and index data in the application group	*expDateValue*	No	90	• "Never Expire" • A positive integer	Yes
federateCFSOD	Federate CFSOD	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
logRetrieval	Whether document retrieval messages should be logged	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes

Table 97. Attributes for the applicationGroup object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
logBulkRetrieval	Whether document bulk retrieval messages should be logged	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logDBQueriesAfter	Whether database queries (after the query) should be logged	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
logDBQueriesBefore	Whether document database queries should be logged	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
logServerPrinting	Whether document server printing messages should be documented	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logIndexAdd	Whether document index add messages should be documented	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logIndexUpdate	Whether document index update messages should be documented	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logIndexDelete	Whether document index delete messages should be documented	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logObjectStore	Whether object store messages should be documented	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logObjectRetrieve	Whether object retrieve messages should be logged	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logAnnotAdd	Whether annotation add messages should be logged	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
logAnnotUpdate	Whether annotation update messages should be logged	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes

Appendix. Appendixes to the Administration Guide   **375**

Table 97. Attributes for the applicationGroup object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
logAnnotDelete	Whether annotation delete messages should be logged	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
maxDBRows	The maximum number of database rows for this application group	Integer	No	10000000	• Positive integer • "Single table for all loads"	Yes
loadsPerTable	The amount of loads per table	*loadsString*	No	"Multiple"	• "Multiple"	No
dbTablespaceType	The type of database table space to use for this application group	*dbtString*	No	"SMS"	• "None" • "SMS" • "Use Tablespace"	Yes
annot	Annotation flags in the document database tables	*yesnoString*	No	"No"	• "Yes" • "No"	No
enhancedRetManagement	Use enhanced retention management	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
impliedHold	Use implied hold	*yesnoString*	No	"No"	• "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"	• "Load" • "Never" • Migration • A positive integer	Yes
migrateIndex	When to migrate indexes from cache	*migrateIndString*	No	"Never"	• "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"	• "Yes" • "No"	Yes

Table 97. Attributes for the applicationGroup object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
useFilegroups	Specifies whether this application group uses file groups	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
updateExistingTables	Specifies whether tables should be updated if an index changes	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

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.

Table 98. Child objects for the applicationGroup object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes
field	1	32	No	Yes
index	0	Unbounded	Yes	No

Table 99. Attributes for the index object under the applicationGroup object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete"	N/A
name	The name of the index	*indexName*	Yes		A positive integer	No
cluster	Whether this is a cluster index	*yesnoString*	No	"No"	• "Yes" • "No"	No

Table 100. Child objects under the index object

Name	Minimum number	Maximum number	Deleteable	Updateable
field	2	16	Yes	No

Table 101. Attributes for the field object under the index object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	The name of the field	*string*	Yes			No
order	The field order	*orderString*	No	"Ascending"	• "Ascending" • "Descending"	No

Table 102. Attributes for the permission object under the applicationGroup object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
user	Provide permissions to the application group to this user	*nameString*	Yes	""	• User name • "_ALL" • "*PUBLIC"	No
group	Provide permissions to the application group to this group	*nameString*	Yes	""	• Group name • "*PUBLIC" • "_ALL"	No
docAddPerm	Whether this user or group should be allowed to add documents to this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
docDeletePerm	Whether this user or group should be allowed to remove documents from this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
docUpdatePerm	Whether this user or group should be allowed to update documents in this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
docPrintPerm	Whether this user or group should be allowed to print documents from this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
docCopyPerm	Whether this user or group should be allowed to copy documents from this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
docCFSODPerm	Whether this user or group should be allowed to add CFS-OD documents to this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 102. Attributes for the permission object under the applicationGroup object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
docHoldPerm	Whether this user or group should be allowed to add holds to documents for this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
annotViewPerm	Whether this user or group be allowed to view annotation in this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
annotAddPerm	Whether this user or group be allowed to add annotation to this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
annotDeletePerm	Whether this user or group be allowed to delete annotation from this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
annotUpdatePerm	Whether this user or group be allowed to update annotation in this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
annotPrintPerm	Whether this user or group be allowed to print annotation from this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
annotCopyPerm	Whether this user or group be allowed to copy annotation from this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
authority	Specifies the authority given	*authagString*	No	"Access"	• "Access" • "Administrator" • "Logical Views"	Yes
queryRes	A query restriction for the user or group	*queryString*	No	""		Yes

Important: Either a user or a group must be specified, but not both.

Table 103. Attributes for the field object under the applicationGroup object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "update"	N/A
name	Database field name	*nameString*	Yes			No
type	Index type	*fieldTypeString*	No	• "Filter" • Index	• "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	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
dataType	The data type of the field	*dataTypeString*	No	"String"	• "Big Int" • "Date" • "Date/Time" • "Date/Time(TZ)" • "Decimal" • "Integer" • "Small Int" • "String" • "Time"	No
segment	If the expiration type is segment, value of this field is used by OnDemand to determine when to delete data from the application group	*yesnoString*	No	"No"	• "Yes" • "No"	No
expireDate	Whether this field is used for the expiration date	*yesnoString*	No	"No"	• "Yes" • "No"	No
lockdown	Whether this is a lockdown field	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
log	Whether to use the system log	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 103. Attributes for the field object under the applicationGroup object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
userExit	Whether to send data to the user exit	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
partition	Used to partition the index data across the multiple nodes	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
cluster	Determines the clustering index for the application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
stringCase	The case to use if the data type is string	*caseString*	No	"Upper"	• "Upper" • "Lower" • "Mixed"	No
stringType	The type of string	*stringType*	No	"Fixed"	• "Fixed" • "Variable"	No
stringLength	The length of the string	Integer	No	0	• **For fixed strings:** 1–254 • **For variable length strings:** 1–2000	No
stringStrip	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	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
task	The task to perform on the object	*taskString*	No	Same as the task attribute of the parent	• "add" • "Update"	N/A
pageCount	Specifies whether this is a page count field	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 103. Attributes for the field object under the applicationGroup object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
documentSize	Specifies whether this is a document size field	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
updateable	Whether this field can be updated	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes

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 104. Child objects under the field object

Name	Minimum number	Maximum number	Deleteable	Updateable
mapping	0	Unbounded	Yes	Yes

Table 105. Attributes for the mapping object under field

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "update" • "delete"	N/A
dbValue	The database value	*dbString2*	Yes			No
displayedValue	The displayed value	*dvString*	Yes			Yes

Application object

Table 106. Attributes for the application object

Name	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the application	*nameString*	Yes			Yes
description	Description of the application	*descString*	No	""		Yes
appGroup	The application group that this application belongs to	*nameString*	Yes			No
identifier	The application group identifier	*dbString2*	Yes	""	Determined by the application group	Yes
dataType	The data type of this application	*appTypeString*	No	• "AFP" • "SCS"	• "AFP" • "Line" • "BMP" • "GIF" • "PCX" • "TIFF" • "PDF" • "JPEG" • "PNG" • "Metacode" • "User Defined" • "OD Defined" • "E-Mail" • "None" • "SCS" • SCS-Extended • Global DJDE	No

Table 106. Attributes for the application object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
indexer	The name of the indexer program to use with this application	*indexerString*	No	"None"	• "None" • "ACIF" • "Generic" • "PDF" • "Xenos" • OS/390 • OS/400®	Yes
dataCompression	The compression technique that is used for the data	*compressionString*	No	"OD77"	• "OD77" • "LZW12" • "LZW16" • "Disable" • "None"	Yes
resCompression	The compression technique that is used for the resources	*compressionString*	No	"OD77"	• "OD77" • "LZW12" • "LZW16" • "Disable" • "None"	Yes
resourceComparison	The number of resource comparisons	*resCompValue*	No	50	0–9999	Yes
largeObject	Whether this application a Large Object	*yesnoString*	No	"No"	• "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	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 106. Attributes for the application object (continued)

Name	Description	DataType	Required	Default value	Possible values	Updateable
postProcesserString	The post processor string	*postProcString*	No	""		Yes
printAmount	The default print option to specify which page or pages to print	*printAmString*	No	"All Pages"	• "All Pages" • "Current Pages"	Yes
defaultPrinter	Default printer	*nameString*	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	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
previewParms	Whether the preview parameters should be used	string	No	""		Yes
expirationDate	Specifies the life of the data and indexes	expDateString	No	"Use AppGroup Value"	• "Use AppGroup Value" • Positive integer	Yes

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 107. 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
udlData	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 108. Attributes for the SCSData 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	*prmodeString*	No	"None"	• "None" • "SOSI1" • "SOSI2" • "SOSI3"	Yes

Table 109. Child objects under the SCSData 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 115 on page 408, Table 116 on page 409, and Table 117 on page 410 for descriptions of the lvHeader, lvField, and logView objects.

Table 110. 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	*orientString*	No	0	 • 0 • 90 • 180 • 270	Yes
paperSize	The output paper size	*paperString*	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 111. Child object under the SCSExData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	20	Yes	Yes

Table 112. 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	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
name	Name of this logical view	*nameString*	Yes			Yes
user	User for this logical view	*nameString*	Yes		• User name • "*PUBLIC"	No
defaultView	Whether this is the default view	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
zoom	Zoom Factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected Area Color	*saColorString*	No	"Yellow"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom"	Yes
bgColor	Background	*bgColorString*	No	"White"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Green Bar" • "Custom"	Yes

Table 113. 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"	• "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"	• "None" • "SOSI1" • "SOSI2" • "SOSI3"	Yes
trc	Whether the data contains table reference characters	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
CC	Whether the data contains carriage control characters	*yesnoString*	No	"Yes"	• "Yes" • "No"	Yes
CCType	The type of the carriage control character	*yesnoString*	No	"ANSI"	• "ANSI" • "Machine"	Yes

Table 114. 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 115. Attributes for the lvHeader object

Attributes	Description	DataType	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 116. Attributes for the lvField object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "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
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 117. Attributes for the logView object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
name	Name of this logical view	*nameString*	Yes			Yes
user	user for this logical (or PUBLIC) view	*nameString*	Yes		• User name • "*PUBLIC"	No
defaultView	Whether this is the default view	*yesNoString*	No	"No"	• Yes • No	Yes
zoom	Zoom factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected area color	*saColorString*	No	"Yellow"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom"	Yes
bgColor	Background color	*bgColorString*	No	"White"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Green Bar" • "Custom"	Yes

Table 117. Attributes for the logView object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
headingColor	Heading color	*hdgColorString*	No	"BG Color"	• "BG Color" • "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom"	Yes
textFidelity	Text fidelity	*fidelityString*	No	"Line"	• "Line" • "Word" • "Char" • "Draft" • "240-pel"	Yes
overstrike	Text overstrike option	*overstrikeString*	No	"Separate Lines"	• "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	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
lockField	Number of fields that you want to lock in place	Integer	No	0	Nonnegative integer	Yes

Table 118. 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	*orientString*	No	0	"0" "90" "180" "270"	Yes
paperSize	The output paper size	*paperString*	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 119. Child objects under the afpData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	20	Yes	Yes

Table 120. 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	taskString	No	"add"	• "add" • delete • "update"	N/A
name	Name of this logical view	nameString	Yes			Yes
user	User for this logical view (or PUBLIC)	nameString	Yes		• User name • "*PUBLIC"	No
defaultView	Whether this is the default view	yesnoString	No	"No"	• Yes • No	Yes
zoom	Zoom factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected area color	saColorString	No	"Yellow"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom"	Yes
bgColor	Background color	bgColorString	No	"White"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Green Bar" • "Custom"	Yes

Table 120. Attributes for the logView object under the afpData object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
imageColor	Image color	*imgColorString*	No	"Black"	• "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Magenta" • "Cyan" • "Custom"	Yes
textFidelity	Text fidelity	*fidelityString*	No	"Line"	• "Line" • "Word" • "Char" • "Draft" • "240-pel"	Yes
imageIntensity	Image intensity	*intensityString*	No	"Normal"	• "Normal" • "Light" • "None"	Yes
copyGroup	The copy group for the document	Integer	No	1	Nonnegative integer	Yes

Table 121. Attributes for the pdfData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
orientation	The orientation or rotation angle, in degrees, of the document	*orientString*	No	0	• 0 • 90 • 180 • 270	Yes

Table 122. Child objects under the pdfData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	Unbounded	Yes	Yes

Table 123. Attributes for the logView object under pdfData

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • delete • "update"	N/A
name	Name of this logical view	*nameString*	Yes			Yes
user	User for this logical view (or PUBLIC)	*nameString*	Yes		• User name • "*PUBLIC"	No
defaultView	Whether this is the default view	*yesnoString*	No	"No"	• Yes • No	Yes
zoom	Zoom factor	Integer	No	100	Positive integer	Yes
selAreaColor	Selected area color	*saColorString*	No	"Yellow"	• "White" • "Black" • "Red" • "Blue" • "Green" • "Yellow" • "Grey" • "Custom"	Yes

Table 124. Attributes for the imageData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
orientation	The orientation or rotation angle, in degrees, of the document	*orientString*	No	0	• 0 • 90 • 180 • 270	Yes

Table 125. Child objects under the imageData object

Name	Minimum number	Maximum number	Deleteable	Updateable
logView	0	20	Yes	Yes

Table 126. Attributes for the logView object under imageData

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • delete • "update"	N/A
name	Name of this logical view	*nameString*	Yes			Yes
user	User for this logical view (or PUBLIC)	*nameString*	Yes		• User name • "*PUBLIC"	No
defaultView	Whether this is the default view	*yesnoString*	No	"No"	• "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
brightness	Image brightness value	Integer	No	128	Integer value 0–255	Yes
greyScale	Whether the image should use grey scales	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 127. Attributes for the udData object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
fileExt	File extension	*fileExtString*	Yes			Yes

Table 128. 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"	• "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 129. 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 130. Attributes for the imageOverlay object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "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

Folder object

Table 131. Attributes for the folder object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the folder	*nameString*	Yes			Yes
description	A description of the folder	*descString*	No	""		Yes
searchType	The note search type	*searchTypeString*	No	"Retrieve"	• "Retrieve" • "Hit List" • "Note"	Yes
displayDocHold	Whether the client displays the hold status for each document	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
displayDocLocation	Whether the client should show the storage location of each document in the document list	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
queryUnmappedFields	Whether this user or group should be allowed to add CFS-OD documents to this application group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 132. 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	32	No	Yes

Table 133. Attributes for the permission object under the folder object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
user	Provide permission to the folder of this user	*nameString*	Yes	""	• User name • "_ALL"	No
group	Provide permission to the folder of this group	*nameString*	Yes	""	• Group name • "*PUBLIC" • "_ALL"	No
adminAuthority	Whether the user or group should have administrator authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
accessAuthority	Whether the user or group should have access authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
fieldsAuthority	Whether the user or group should have authority to change fields	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
publicNQAuthority	Whether the user or group should have public named query authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
privateNQAuthority	Whether the user or group should have private named query authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
viewNQAuthority	Whether the user or group should have named query view authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
maxHits	The maximum number of hits allowed	*maxHitsType*	No	"No Limit"	• "No Limit" • "None" • A positive integer value	Yes
secondaryFolder	Whether this is a secondary folder	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Table 133. Attributes for the permission object under the folder object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
userGroupFields	Whether user/group fields should be created for this user/group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
fullReportBrowse	Whether full report browse should be allowed for this user/group	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
serverBasedSorting	Whether sorting should be done on the server	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Important: You must specify either a user or a group object, however, you cannot specify both.

Table 134. Attributes for the applicationGroup object under the folder object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete"	N/A
name	Name of this object	*nameString*	Yes		Application group name	No

Table 135. Child objects under the applicationGroup object under the folder object

Name	Minimum number	Maximum number	Deleteable	Updateable
application	0	Unbounded	Yes	No

Table 136. Attributes for the application object under the applicationGroup object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete"	N/A
name	Name of this object	*nameString*	Yes		Application name	No

Table 137. Attributes for the field object under the folder object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
name	Name of the folder field	*nameString*	Yes			Yes
description	A description of the folder field	*descString*	No	""		Yes
fieldType	Field type of the folder object	*fieldString*	No	"String"	• "Ann Color Search" • "Ann Text Search" • "Application Group" • "Big Int" • "Date" • "Date/Time" • "Date/Time(TZ)" • "DecFloat (16)" • "DecFloat (34)" • "Document Type" • "Hold" • "Integer" • "Segment" • "Small Int" • "String" • "Text Search" • "Time"	No
mappingType	Mapping type of the folder field	*mappingString*	No	"Single"	• "Single" • "Range" • "Operator Or"	No
applicationGroup	The Application Group to use for a segment field	*nameString*	No	""	""	Yes

Table 138. 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 139. Attributes for the mapping object under the field object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete"	N/A
dbName	The database field name	*nameString*	Yes			No
appGroup	The application group name	*nameString*	Yes			No

Table 140. Attributes for the fieldInfo object under the field object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
task	The task to perform on this object	*taskString*	No	"add"	"add" "delete" "update"	N/A
user	The user that is associated with this field definition	*nameString*	Yes		User name	Yes
group	The group that is associated with this field definition	*nameString*	Yes		- 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	*sortTypeString*	No	"Ascending"	"Ascending" "Descending"	Yes
equal	Whether the equal operator should be available	*compString*	No	"Default"	"Yes" "No" "Default"	Yes
notEqual	Whether the not equal operator should be available	*compString*	No	"No"	"Yes" "No" "Default"	Yes
lessThan	Whether the less than operator should be available	*compString*	No	"No"	"Yes" "No" "Default"	Yes

Table 140. Attributes for the fieldInfo object under the field object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
ltOrEqual	Whether the less than or equal operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
greaterThan	Whether the greater than operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
gtOrEqual	Whether the greater than or equal operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
in	Whether the in operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
notIn	Whether the not in operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
between	Whether the between operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
notBetween	Whether the not between operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
like	Whether the like operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes
notLike	Whether the not like operator should be available	*compString*	No	"No"	• "Yes" • "No" • "Default"	Yes

Table 140. Attributes for the fieldInfo object under the field object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
default	Determines whether the field contains a default value when the user opens the folder	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
fixed	Determines whether the value that you type in the default fields can be changed by the user	*compString*	No	"No"	• "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"	• "Yes" • "No"	Yes
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	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
internal	Determines whether OnDemand lists the displayed values or the database values for the folder search field	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
wildCard	Determines whether OnDemand uses wildcards with field search values	*wildString*	No	"Append"	• "Prepend" • "Append" • "Both" • "None"	Yes
min	For integer, decimal, date, and time fields, determines the minimum value that the user can type in the search field	Integer	No	Based on datatype		Yes

Table 140. Attributes for the fieldInfo object under the field object (continued)

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
max	For integer, 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	*intervalString*	No	"Last"	• "Last" • "Next"	Yes
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"	• "Days" • "Months" • "Years" • "Hours" • "Minutes" • "Seconds"	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.

Cabinet object

Table 141. Attributes for the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the cabinet	nameString	Yes			Yes
description	A description of the cabinet	descString	No	""		Yes

Table 142. Child objects under the cabinet object

Name	Minimum number	Maximum number	Deleteable	Updateable
folder	0	Unbounded	Yes	No
permission	0	Unbounded	Yes	Yes

Table 143. Attributes for the folder object under the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the folder	*nameString*	Yes			No
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete"	N/A

Table 144. Attributes for the permission object under the cabinet object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
user	Name of the user	*nameString*	No		• The user name • "_ALL"	No
group	Name of the group	*nameString*	No		• The group name • "_ALL" • "*PUBLIC"	No
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	N/A
accessAuthority	Whether this user or group has access authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
adminAuthority	Whether this user or group has administrative authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Hold object

Table 145. Attributes for the hold object

Attributes	Description	DataType	Required	Default value	Possible values	Updateable
name	Name of the hold	*nameString*	Yes			Yes
description	A description of the hold	*descString*	No	""		Yes

Table 146. Child objects under the hold object

Name	Minimum number	Maximum number	Deleteable	Updateable
permission	0	Unbounded	Yes	Yes

Table 147. 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	*nameString*	Yes	""	• The user name • "_ALL"	No
group	Name of the group that can view and maintain this hold	*nameString*	Yes	""	• The group name • "_ALL"	No
task	The task to perform on this object	*taskString*	No	"add"	• "add" • "delete" • "update"	Yes
adminAuthority	Whether this user or group has administrative authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes
accessAuthority	Whether this user or group has access authority	*yesnoString*	No	"No"	• "Yes" • "No"	Yes

Data field limitations

Table 148. Data field limitations

Datatype	Minimum length	Maximum length
dbString	1	18
dbString2	1	254
defString	0	254
descString	0	120
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

Logging

Content Manager OnDemand provides the ability to log many different items, errors, and actions for application groups, users, and server events. Every application group has some message logging turned on by default, and the number of default items that are logged depends on the level of the administrative client in use. To reduce the amount of unwanted logging information, for any new application group, verify that only those items needing to be logged are checked on the Message Logging tab.

The System Log or System Logging Facility is the main message and error logging interface and repository. It is a standard Content Manager OnDemand application group with a folder that can be viewed from a Content Manager OnDemand client like any other data. Therefore, if you try to log the actions of the System Log itself, you are essentially logging your logging.

For example, if you log the Index Add command, which generates message #103, you see a vast multiplication of messages as the initial action is logged, and then the logging of the action is logged, then the logging of the logging of the action is logged, and so on, into an infinite loop.

Under normal circumstances, no message logging should be turned on for the System Log application group. If you need to turn on message logging for the System Log application group for troubleshooting purposes, you should turn it off again as soon as possible.

Identifying a load ID from a user folder query

To determine what load a document is from, given a particular hit list, you can use the ARSDOC query command.

Use the –D parameter to obtain the document handle information. This gives you some of the following information to query the system log to find the exact load ID:

- Primary node ID
- Secondary node ID
- Part of the doc_name

To get the exact load ID for a specific index, use the ARSDOC query command with the option -I. For the QUERY command function, option -I appends the load ID to the end of each output record. The load ID is separated from the database field values by a delimiter, that is, the comma character. You can specify a different delimiter with the -e parameter. To use the -I parameter, you must have permission to access the system log application group and folder. If the load ID is not found in the system log, the string Load ID could not be found is appended to the end of the output record.

Restriction: You cannot specify the -H, -n, or -N parameters when you specify the -I parameter.

Important: Searching the system log can be 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.

Troubleshooting scenario 1: Cannot find data when using a mm/dd/yy date format in search

Symptoms

You defined a new application or application group that contains date fields formatted as %d/%m/%Y, however, I am only able to find the data when I use a date search format of mm/dd/yy.

Causes

The display format of the field in your folder definitions needs to be changed.

Resolving the problem

Change the display format to: %d/%m/%Y.

Troubleshooting scenario 2: Error occurred during the CD-ROM mastering process

Symptoms

You received this error message during the CD-ROM mastering process:

Unable to open original file for staging.

Causes

It is possible that the Content Manager OnDemand client was updated on your PC, but the new master **INI** file was not transferred.

Resolving the problem

Copy the new master **INI** file over to your PC.

Troubleshooting scenario 3: ARSLOAD daemon cannot load an application group with multiple applications

Symptoms

You created a Content Manager OnDemand load daemon to automate the load process for an application group that contains multiple applications. However, the arsload daemon cannot load an application group.

Causes

The Application Name for your Content Manager OnDemand Load Data services is set to None.

Resolving the problem

Verify your load information in the Content Manager OnDemand Configurator for Windows:

1. Open the Content Manager OnDemand Configurator.
2. Click **Services**.
3. Right click **OnDemand Load Data** (ARCHIVE). Assuming your OnDemand instance is ARCHIVE.
4. Open its Properties and select **Load Information**.
5. Deselect the None option in the Application Name field. You must deselect the None option in the Application Name field to allow the Content Manager OnDemand load daemon to process an application group with multiple applications.

Troubleshooting scenario 4: The load process is slowed down by a .lst file

Symptoms

The loading of Adobe Acrobat PDF files are slowed down due to a .lst file. The file name should be similar to AdobeFnt05.lst.

Causes

It is possible that you have not installed the Adobe fonts on your server. The .lst file that you see is a mapping of fonts from the documents. With each load, this file grows and maps the fonts within the new load object. If you install the Adobe fonts on the server, the .lst file maps to those server fonts and does not continue to grow. The only exception is when it adds those fonts that it could not map to.

Resolving the problem

You can either delete the .lst font file regularly, or install the Adobe fonts on the server so that they are only mapped once.

Troubleshooting scenario 5: You received the error that connection cannot be established while using ARSLOAD

Symptoms

While running the ARSLOAD command, you received this error:

connection cannot be established

Causes

A possible cause is that when you specified the –h parameter for ARSLOAD, you did not set the instance name as ARCHIVE and set the port number to 1445 or 0.

Resolving the problem

If the only parameter that you specify for ARSLOAD is the –h parameter, make sure that you set the instance name as ARCHIVE and set the port number to 1445 or 0.

Troubleshooting scenario 6: ARSLOAD cannot find the TIFF images when using generic index

Symptoms

While running ARSLOAD to automatically load data into OnDemand, each load fails.

Causes

It is possible that you did not specify full file names. The generic index file format requires that you specify absolute file names, that is, full file names.

Resolving the problem

Make sure that you specify absolute file names. If you cannot make this happen, try to run separate ARSLOAD command daemons for each download directory, and start them up by making sure that they are invoked from the download directory in which they monitor.

Troubleshooting scenario 7: Buffer overrun problem while exporting application groups to a local server or another Content Manager OnDemand server

Symptoms

The Content Manager OnDemand administrative client encounters a buffer overrun problem while exporting application groups to a local server or another Content Manager OnDemand server if the application identifier field contains large numbers of Displayed Value/Database Value mappings, for example, 2619 pairs.

You might receive one or both of the following error messages:

```
Buffer overrun detected!
```

or

```
Unable to add the application group application group name.
A return code of 'Exists' was returned from the server.
```

Causes

The current buffer is not large enough to contain so many internal and external mappings. This is a limitation in the export functionality.

Resolving the problem

You can use one of the following workarounds:
- Limit the number of the application group field mappings.
- Use the XML batch administration program ARSXML to export the application groups. For instructions, see "Importing and exporting administrative objects through an XML interface" on page 195.

Troubleshooting scenario 8: The ARSLOAD program with a generic indexer does not load my AFP documents with inline resources correctly

Symptoms

You used ARSLOAD with a generic indexer to load a data file with more than one AFP document and a resource group, confirmed that the structure of your data file was correct, and received a confirmation on your console that the documents were

successfully loaded into your Content Manager OnDemand server. However, when you retrieve the documents from the Content Manager OnDemand server, you received a 1303 error message indicating that some resource was missing.

Causes

The primary cause of this problem is that the resources must be in a separate file when you use ARSLOAD with a generic indexer to load the AFP documents into your DB2 Content Manager OnDemand server. Using ARSLOAD with a generic indexer does not allow resources to be defined inline with the documents in the data file.

Resolving the problem

Change the extension of your data file to `*.out` if your data file has another extension, because ARSLOAD searches for a data file with the `*.out` extension.

Ensure that you have a lowercase extension (for example, `*.out`, `*.ind`, `*.res`) for all the files that ARSLOAD needs to load on AIX, because AIX is case-sensitive. Also ensure that the `GROUP_FILENAME` in `*.ind` is identical to the name of your data file.

If you use ARSLOAD with a generic indexer, you can load several AFP documents (multiple BDT/EDT pairs) into your OnDemand server with just one data file. To do so, set the length and offset in your `*.ind` file. However, if the ACIF indexer is used, you cannot do so, because ACIF indexer allows only one data file to contain one document (one BDT/EDT pair).

Troubleshooting scenario 9: Why Content Manager OnDemand does not support query result modifier SQL clauses

Symptoms

Several commands and APIs are available that allow a user to submit raw SQL for querying the OnDemand application group data tables. However, because the submitted SQL is not parsed or validated, certain query result modifiers are not supported. These query result modifiers include but are not limited to:
- ORDER BY
- READ ONLY
- FETCH FIRST
- OPTIMIZE FOR
- SORT ASC/DESC
- GROUP BY

Causes

This happened because of the overall design of the Content Manager OnDemand application group. OnDemand uses database segments, multiple tables, and even multiple queries in order to increase the performance of queries and lessen the workload of the database. SQL queries against the application group data tables are first limited to just the tables that meet the database segment constraints. This is accomplished by using a start and stop date for each data table. That list of tables is used to generate a full-select with `UNION ALL` SQL statement. This generated SQL string is limited by some restraints, such as the maximum length of the generated SQL statement and the number of tables that are searched in a single

query. These generated SQL statements create some highly complex queries that are repeated in multiple select statements while searching over a large range of data. Adding search result modifiers, such as ORDER BY, compounds the workload on the database.

Resolving the problem

In an attempt to reduce the work to the clients, the Content Manager OnDemand folder definitions support actions such as sorting, grouping, and limiting the number of rows. This approach frees the database from having to perform complex query modifications that require a large amount of time to optimize and large amounts of memory to perform.

The following is an example using the arsdoc command, where you submit an SQL query like ACCTNO LIKE '1001% and specify a segment date like 12/01/2004 to 12/31/2004 to OnDemand:

```
arsdoc query –u user –p whatever –f "Credit Card Statements"
–i "ACCTNO LIKE '1001" –s "12/01/2004,12/31/2004"
```

The Content Manager OnDemand server uses the database segment values to determine what tables need to be searched. This might yield only a single table to search and the generated SQL looks like:

```
SELECT * FROM root.BAA20 WHERE ACCTNO LIKE '1001
```

In this case, you can append a modifier such as ORDER BY to your submitted SQL. The generated SQL is simply appended to the end of a single SELECT/WHERE statement.

In another example, extend the database segment date range to 01/01/2004-12/31/2004:

```
arsdoc query –u user –p whatever –f "Credit Card Statements"
 –i "ACCTNO LIKE '1001" –s "01/01/2003,12/31/2004"
```

The number of tables being searched increase, possibly by two tables, and the generated SQL looks like:

```
SELECT * FROM BAA18 WHERE ACCTNO LIKE '1001 UNION ALL
SELECT root.BAA19.*, 'BAA19' FROM root.BAA19
WHERE ACCTNO LIKE '1001 UNION ALL
SELECT root.BAA20.*,'BAA20'
FROM root.BAA20
WHERE ACCTNO LIKE '1001
```

This example indicates that using a modifier like ORDER BY cannot be appended directly onto each WHERE clause. It results in invalid SQL syntax because it is included as part of each WHERE clause. Valid SQL appends it to the end of the entire full-select statement. Because Content Manager OnDemand does not parse the query SQL that is submitted, it is not able to append the ORDER BY modifier to the end of the full-select statement.

To take this example a step further, change the database segment range to 01/01/1990-12/31/2004:

```
arsdoc query –u user –p whatever –f "Credit Card Statements" –i
"ACCTNO LIKE '1001" –s "01/01/1990,12/31/2004"
```

The number of tables might increase to a hundred or more, and the resulting SQL is too long or span too many tables. In this case, OnDemand generates multiple

full-select queries to accomplish the task, and cannot generate valid SQL across the queries and have it perform an `ORDER BY` for the entire result set.

In this case, Content Manager OnDemand depends on the client to modify the results. While using APIs or commands like ARSDOC to submit SQL queries directly to Content Manager OnDemand, use just the fields being queried and always supply a segment date range. Without a segment date range supplied, all tables in the application group are searched. After the results are returned, you can modify them accordingly.

Troubleshooting scenario 10: AFP file loading failed with invalid user ID or password

Symptoms

I tried to load an AFP file, but the loading failed with the following error:

```
"Userid or password is invalid
Unable to retrieve the object >2<
arsload: 11/14/05 14:58:36 Loading failed
arsload: Processing failed for file
>SYSD.LWMB88FC.FCMNTHYD.FCOCTDEV.2005312.13100688809.ARD<"
```

Causes

The reason that the loading failed is that this specific user does not have the view permission although the user has add permission.

Resolving the problem

Grant the view permission to this user ID for loading.

Troubleshooting scenario 11: The text search failed with error that module arsusdoc cannot be loaded

Symptoms

I tried to do a text search, and received an error similar to the following in my system log:

```
"19/10/2005 15:06:34      ADMIN        8617 Error No      160
Unable to load module >/opt/ondemand/bin/exits/arsusdoc<.
rc=12  Srvr->wd00<-
19/10/2005 15:06:34      ADMIN        8617 Error No      176
BulkTextSearch Failed  Srvr->wd00<- "
```

Causes

This error occurred because the library dependencies were missing although arsusdoc existed.

Resolving the problem

Run the command

```
"ldd  /opt/ondemand/bin/exits/arsusdoc"
```

to output the shared library dependencies for arsusdoc.

The following output of "ldd arsusdoc" shows that shared library 'libOPP.sl' is missing:

```
"/usr/lib/libm.2 =>       /usr/lib/libm.2
 /usr/lib/libcl.2 =>      /usr/lib/libcl.2
 /usr/lib/libisamstub.1 =>        /usr/lib/libisamstub.1
 /usr/lib/libdld.2 =>     /usr/lib/libdld.2
 /usr/lib/libc.2 =>       /usr/lib/libc.2
 /usr/lib/libdld.2 =>     /usr/lib/libdld.2
 /usr/lib/libdld.2 =>     /usr/lib/libdld.2
 /usr/lib/libc.2 =>       /usr/lib/libc.2
 /usr/lib/libstream.2 => /usr/lib/libstream.2
 /usr/lib/libstd.2 =>     /usr/lib/libstd.2
 /usr/lib/libCsup.2 =>   /usr/lib/libCsup.2
 /usr/lib/dld.sl: Can't open shared library: /opt/ondemand/lib/libOPP.sl"
```

Make sure that all the dependencies of arsusdoc are in the Content Manager OnDemand library subdirectory.

Hints and tips about using an XML interface to import and export administrative objects

This section includes some hints and tips about using an XML interface to import and export administrative objects.

"Tip 1: Child objects must be created under parent objects"

"Tip 2: Application group and folder authorities and permissions are not linked"

"Tip 3: Parsing error while running ARSXML" on page 462

Tip 1: Child objects must be created under parent objects

In the XML interface, it is only possible to have child objects under parent objects, not the other way around. For example, the following XML file example is valid:

```
<group name="MyGroup">
 <user name="tom" />
 <user name="chuck" />
</group>
```

However, this example is not valid:

```
<user name="tom">
 <group name="MyGroup">
 </group>
</user>
```

The same is true for permissions for users and groups. They are child objects under the application group and folder objects. For example, to add a folder that contains permission for the user Bill, the XML file might look like this:

```
<folder name="MyFolder"...>
 <permission user="Bill" accessAuthority="Yes"
  viewNQAuthority="Yes" maxHits="No Limit" />
 ...
</folder>
```

Tip 2: Application group and folder authorities and permissions are not linked

In the XML interface, the application group and folder authorities and permissions are not linked. For example, if you specify adminAuthority="Yes", you do not get the view, add, print, fax, and copy document permissions, or the view, add, and copy annotations permissions. You need to specify each item individually.

Tip 3: Parsing error while running ARSXML

If you receive this error message while running ARSXML:

```
A parsing error occurred in file fileName, Line nnn, Column nnn :
cvc-elt.1: Cannot find the declaration of element 'onDemand'.
```

It usually indicates that the OnDemand schema file (`ondemand.xsd`) cannot be found. The OnDemand schema file should be placed in the directory where the OnDemand XML file resides, otherwise, the fully qualified path name should be placed in the `noNamespaceSchemaLocation` element of the OnDemand XML file.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY   10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this 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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All 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 the names and addresses used by an actual business enterprise 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.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at △Copyright and trademark information△ at: www.ibm.com/legal/copytrade.shtml

The following terms are trademarks or registered trademarks of other companies:

Adobe, Acrobat, Portable Document Format (PDF), PostScript®, and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel®, Intel logo, Intel Inside®, Intel Inside logo, Intel Centrino®, Intel Centrino logo, Celeron®, Intel Xeon®, Intel SpeedStep®, Itanium®, and Pentium® are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Index

Numerics

IBM ®

Program Number: 5724-J33