Content Manager OnDemand
Version 10 Release 5

Content Manager OnDemand Distribution Facility Installation and Reference

SC19-3358-03

Note

Before using this information and the product it supports, read the information in “Notices” on page 77.

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

- Version 10 Release 5 of IBM® Content Manager OnDemand for Multiplatforms (product number 5724-J33) and IBM Content Manager OnDemand for z/OS (product number 5697-CM1)
- Version 7 Release 3 of IBM Content Manager OnDemand for i (product number 5770-RD1.)

Contents

Chapter 1. ibm.com® and related resources

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

Support and assistance

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

IBM Knowledge Center

See your online product information in IBM Knowledge Center.

- For Content Manager OnDemand for Multiplatforms, see https://www.ibm.com/support/knowledgecenter/SSEPCD.
- For Content Manager OnDemand for z/OS, see https://www.ibm.com/support/knowledgecenter/SSQHWE
- For Content Manager OnDemand for i, see https://www.ibm.com/support/knowledgecenter/SSB2EG

PDF publications

See the following for PDF publications:

- For Content Manager OnDemand for Multiplatforms, see https://www.ibm.com/support/pages/node/1079037.
- For Content Manager OnDemand for z/OS, see https://www.ibm.com/support/pages/node/1079043
- For Content Manager OnDemand for IBM i, see http://www.ibm.com/support/docview.wss?uid=ibm10740829

Contacting IBM

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

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

Chapter 2. Introduction

Content Manager OnDemand Distribution Facility (ODF) is the report distribution feature for IBM Content Manager OnDemand for Multiplatforms, z/OS®, and IBM i. ODF provides an easy way to automatically group reports and portions of reports and distribute the reports to multiple users. ODF distributions can be printed, created as an output file, or emailed as an attachment.

ODF can retrieve reports that are stored in a Content Manager OnDemand server on z/OS, IBM i, or any of the operating systems that are supported by Content Manager OnDemand for Multiplatforms.

ODF architecture

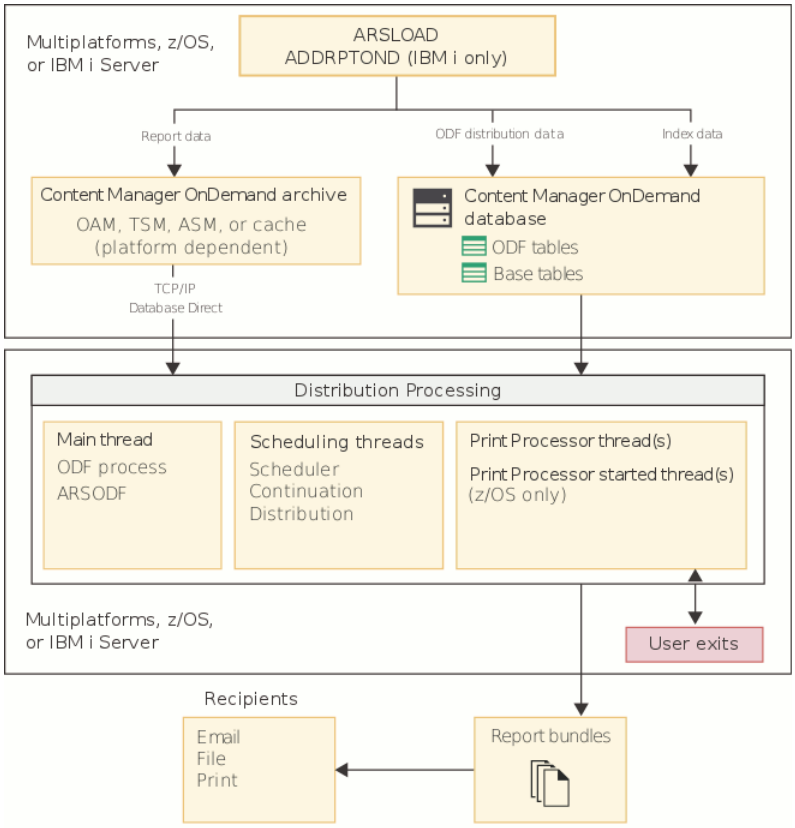


Figure 1. ODF architecture

The architecture of ODF includes core functions of Content Manager OnDemand that trigger ODF processing based on predefined distribution and recipient definitions. The process begins when the ARSLOAD program, the primary data loading program, runs on a z/OS server, IBM i server, or on one of the Multiplatforms servers running AIX®, Linux, or Windows. Report data is loaded into the Content Manager OnDemand archive, and index data and information about the ODF distributions that are defined for that data are added to the Content Manager OnDemand database.

The ARSODF program, the distribution processing program, uses input from the ARSLOAD process to create the distribution output. Distributions are scheduled based on criteria that you define to ODF. Distributions can contain a single report or multiple reports that are grouped together. Some distributions generate email attachments or files. Some distributions generate print output. A number of user exit programs allow for special handling of distributions with unique requirements.

ODF components

ODF provides an easy way to automatically group reports and portions of related reports, organize them, convert the report data into different formats, and send them through email to multiple users, write them to a file, or make them available for printing.

Applications

In Content Manager OnDemand, an *application* is an object that describes the physical attributes of a report or input file, such as the type of data in the input file, the code page, and whether the input data contains carriage control characters. An application also contains instructions that the data indexing and loading programs use to process the input data.

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
- Information about carriage control characters

The logical characteristics of a report include the different ways that you want to present the information that is 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 also create logical views for the application. Logical views represent different ways to display pages of the report. You can configure 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 the following sections:

- General
- View Information
- Indexer Information
- Load Information
- Logical View Fields
- Logical Views
- Miscellaneous Options

Each section contains fields that you can use to enter information about the application.

Application groups

An *application group* is a collection of reports that is stored in Content Manager OnDemand.

The reports in an application group have common indexing and storage management requirements. An application group represents both the input data that is stored in Content Manager OnDemand and the indexing information that is used to search and retrieve documents.

Content Manager OnDemand maintains a table that describes each application group. The application group table contains the characteristics and properties that you assigned when you created (and updated) the application group. For example, when you define the users that can access the reports that are stored in an application group, Content Manager OnDemand stores the information in the application group table.

Content Manager OnDemand also maintains tables of index data in the application group. When you load a report into the application group, Content Manager OnDemand stores the index data for the report into a table in the application group.

When you define an application group, you specify the properties of the application group, such as the database organization, storage characteristics for the reports that are contained in the application group, and the names and characteristics of the database fields that are common to the applications that are assigned to the application group. You can also specify the users and groups that can access the data contained in the application group by using Content Manager OnDemand client programs. You can specify the types of functions users can perform, such as viewing and printing. You can also assign authority to administer the application group to a user. The administrator authority allows a user to update and delete the application group.

Recipients

Each person in your organization that will receive a distribution must be defined as a recipient. The recipient name is used in conjunction with the distribution name to uniquely identify the distribution. A recipient does not need to be a Content Manager OnDemand user.

Recipient lists

An ODF recipient list contains a group of individual recipients who receive the same distributions. You cannot define a recipient list as a member of another recipient list. The ODF recipient list name must be unique and cannot be the same as an individual recipient.

Report IDs

The report ID identifies the Content Manager OnDemand application group and application to the ODF system. It is used to identify the reports that ODF can distribute.

Distributions

An ODF distribution is a collection of reports or portions of reports that can be distributed to a recipient or recipient list. The distribution can include reports from different application groups and applications.

The reports to include in the distribution are identified by one or more report bundles that are sent to one or more recipients. As the reports are stored into Content Manager OnDemand, the identified report bundles are collected so that they can be later distributed based on the distribution criteria that is selected. A distribution can be printed, written to a file, or sent in an email.

A distribution is identified by two components:

- Recipient name
- Distribution name

Report bundles

An ODF report bundle identifies a report or a portion of a report that will be included in a report distribution. The definitions in the report bundle determine the output format and the location of the report bundle. Report bundle output is created from a unique combination of report ID (application group and application), SQL query, and load ID.

Report bundle Wait/Ignore Indicator

The report bundle Wait/Ignore Indicator is used with the distribution scheduling method to control when a distribution is submitted for report bundle processing.

A value of **Wait** indicates that this report bundle is a required report bundle and the distribution should not be processed without it. A value of **Don't Wait** indicates that the distribution can be processed without the report bundle being available.

Distribution scheduling methods

The distribution scheduling method gives you control over when the distribution will be scheduled and, in conjunction with the **Wait/Ignore Indicator**, determines when it will be submitted for report bundle processing. Report bundles defined with a **Wait/Ignore Indicator** of **Wait** are considered required reports.

You can use the following distribution scheduling methods:

Loaded
The distribution is scheduled for processing when the first report in the distribution is loaded into Content Manager OnDemand. The distribution is submitted for report bundle processing when all the required reports in the distribution have been loaded. The **Wait/Ignore Indicator** is not overridden with the **Loaded** distribution method. If none of the report bundles in the distribution are defined as required reports, then the distribution will be scheduled and processed each time a report bundle is loaded. If any report bundle is defined as required, then the distribution will not be processed until that report bundle is loaded.

All Ready
This distribution is always scheduled for report bundle processing. The scheduling of the distribution is not dependent on report bundles being loaded. The distribution is submitted for report bundle processing when all the required report bundles in the distribution have been loaded. The **Wait/Ignore Indicator** is not overridden with the **All Ready** distribution method. The distribution will not be processed until all required report bundles are loaded.

Time of Print
This distribution method is processed similar to the **Loaded** method. The distribution is scheduled upon the first load of a report bundle and will be submitted for report bundle processing when all report bundles become available. If the distribution has not been processed by the time specified in the **Time** field, it will be submitted for report bundle processing even if all of the required reports are not available. This method ensures that the distribution will be printed at least once a day with any available report bundles.

Time of Day
The distribution is only scheduled at the specified time of day. The distribution is submitted for report bundle processing when all the required report bundles have been loaded into Content Manager OnDemand. A distribution with this method will only be scheduled once a day. If all required report bundles are not available, then the distribution will not be submitted for report bundle processing.

External
A process outside of ODF schedules the distribution. The distribution is submitted for report bundle processing when all required report bundles have been loaded.

Chapter 3. Enabling ODF

About this task

With Content Manager OnDemand for Multiplatforms, the ODF software is included as part of the Content Manager OnDemand for Multiplatforms product. However, you must enable ODF to run on your system by following the instructions in this section. You can enable the software by using a GUI interface with the commands as shown or you can add `-i console` to the end of the command string to run the enabler in console mode. For example, you can run the AIX enabler in console mode by running: `./ododfaix.bin -i console`

With Content Manager OnDemand for z/OS, the ODF software is included as part of the Content Manager OnDemand for z/OS product. No additional steps are required to enable ODF on z/OS.

With Content Manager OnDemand for i, the ODF software is an optional program product option. For installation instructions, see the *Content Manager OnDemand for i Common Server Planning and Installation Guide*.

Running the enabler program

Follow the instructions in this section to enable ODF.

About this task

Choose one of the following commands to run to enable ODF, depending on your Content Manager OnDemand server platform.

Procedure

AIX On an AIX system, run the following command:

- `./ododfaix.bin`

Linux On a Linux system, run the following command:

- `./ododflinux.bin`

Linux On a Linux 390 system, run the following command:

- `./ododflinux390.bin`

Windows On a Windows system, run the following command:

- `x:\ododf\windows\ododfwin.exe`

z/OS On a z/OS system, ODF is automatically enabled when you install the ODF SMP/E package.

- No command is required to enable ODF.

IBM i On an IBM i system, ODF is automatically enabled when you install the ODF program product option.

- No command is required to enable ODF.

Setting up the OnDemand Configurator (Windows™)

When running ODF on a Windows server, you can setup the OnDemand Configurator to configure and run ARSODF. This is optional.

About this task

Windows To setup the OnDemand Configurator to use with ODF on a Windows server, follow these steps:

Procedure

1. Open the Content Manager OnDemand configurator program.
   To do this, select **Start** > **All Programs** > **IBM OnDemand Server V10.5** > **OnDemand Configurator V10.5**
2. Expand the server and select **Services**.

   You should see the **OnDemand Distribution Facility** service listed. If it is not there, close and restart the configurator program.
3. If the current status of the **OnDemand Distribution Facility** service is Stopped, the service is already installed. If the current status of the service is Not installed, you need to install the service. To install the service, right-click **OnDemand Distribution Facility** service to open the option menu, and select **Install**. Enter the user ID and password, and click **OK**. Specify the same user ID that you used to install other Content Manager OnDemand services when the instance was created.

Chapter 4. Creating ODF table spaces or filegroups

The method to create table spaces or filegroups for ODF is different depending on the database you are using. Use the instructions in this section to create table spaces or filegroups for ODF based on your database environment.

About this task

Creating table spaces is required when running ODF on a Content Manager OnDemand for z/OS server.

Creating table spaces or file groups is optional when running ODF on a Content Manager OnDemand for Multiplatforms server. Tables and indexes can be created in a default table space or user-defined table spaces. If you want to use a default table space, you do not need to perform any additional steps related to table spaces. If you want to use user-defined tables spaces, follow the instructions in this section.

Creating table spaces is not required when running ODF on a Content Manager OnDemand for i server.

Db2®

About this task

Run the following CREATE commands to create tables spaces that are used for the ODF system tables when using DB2®.

Procedure

1. Modify the sample CREATE command to create the table spaces listed. The CREATE command must be run multiple times, once for each of the table spaces names. The example shows the CREATE command for the first table space in the list.
2. Submit the job to run the CREATE commands.
3. Check the completion status to confirm that the table spaces were created successfully.

Example

Linux AIX For Content Manager OnDemand for Multiplatforms, use the following sample to create each of the table spaces listed. For example, to create the table space for table ARSDFBDT:

```
# db2 connect to ODinstance

CREATE TABLESPACE ARSDFBDT;
```

z/OS For Content Manager OnDemand for z/OS, use the following sample to create each of the table spaces listed. For example, to create the table space for table ARSDFBDT:

```
SET CURRENT SQLID='ARCHIVE';

CREATE TABLESPACE ARSDFBDT
       IN         ARSDBASE
       USING      STOGROUP ARSSGRP
       PRIQTY     200
       SECQTY     40
       LOCKSIZE   ANY
       CLOSE      NO
       BUFFERPOOL BP0;
```

For Content Manager OnDemand for z/OS, a sample job for creating table spaces is available in SARSINST (ARSTBODF).

Use appropriate sample for your platform to create these table spaces:

- ARSDFBDT
- ARSDFDCT
- ARSDFDRT
- ARSDFDST
- ARSDFLIS
- ARSDFPPT
- ARSDFUOT
- ARSDFCRT
- ARSDFEML

Linux AIX Verify or replace the following parameter in the CREATE command example for Content Manager OnDemand for Multiplatforms:

ODinstance
The Content Manager OnDemand instance database name to which the ODF tables and table spaces belong.

z/OS Verify or replace the following parameters in the CREATE command example for Content Manager OnDemand for z/OS:

SQLID='*ARCHIVE*'
The SQLID that is specified for the database should be the same as that specified for the **SRVR_INSTANCE_OWNER** parameter in the ARS.INI file. The default value in the ARS.INI file is ARCHIVE. (See “Verify the ARS.INI file” in the *IBM Content Manager OnDemand for z/OS: Configuration Guide*.)

IN *ARSDBASE*
Specifies the name for the Content Manager OnDemand database.

Important: The database name that you specify must match the name that you specified for the **SRVR_INSTANCE** parameter in the ARS.INI file. The default value in the ARS.INI file is ARSDBASE. (See “Verify the ARS.INI file” in the *IBM Content Manager OnDemand for z/OS: Configuration Guide*.)

USING STOGROUP *ARSSGRP*
Specifies the name for the storage group.

Oracle

About this task

Run the following CREATE commands to create tables spaces that are used for the ODF system tables when using Oracle.

Procedure

1. Modify the sample CREATE command to create the table spaces listed. The CREATE command must be run multiple times, once for each of the table spaces names. The example shows the CREATE command for the first table space in the list.
2. Submit the job to run the CREATE commands.
3. Check the completion status to confirm that the table spaces were created successfully.

Example

Use the following sample to create each of the table spaces listed. For example, to create the table space for table ARSDFBDT:

```
# sqlplus "userid/password@db_name"

CREATE TABLESPACE ARSDFBDT DATAFILE 'ARSDFBDT_TS.dbf' SIZE 1M
      AUTOEXTEND ON MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
```

Use the sample to create these table spaces:

- ARSDFBDT
- ARSDFDCT
- ARSDFDRT
- ARSDFDST
- ARSDFLIS
- ARSDFPPT
- ARSDFUOT
- ARSDFCRT
- ARSDFEML

Verify or replace the following parameters in the CREATE command example:

userid
The Content Manager OnDemand instance owner.

password
The Oracle database password of the instance owner.

db_name
The Content Manager OnDemand Oracle database name.

SQL server

About this task

Run the following ALTER DATABASE commands to create filegroups that are used for the ODF system tables when using SQL Server.

Procedure

1. Modify the sample ALTER DATABASE command to create the filegroups listed. The ALTER DATABASE command must be run multiple times, once for each of the filegroup names. The example shows the ALTER DATABASE command for the first filegroup in the list.
2. Submit the job to run the ALTER DATABASE commands.
3. Check the completion status to confirm that the filegroups were created successfully.

Example

Use the following sample to create the filegroups listed. For example, to create the filegroup for table ARSDFBDT:

```
# sqlcmd -d ODinstance

ALTER DATABASE CMOD_DB ADD FILEGROUP "ARSDFBDT";
```

Use the sample to create these filegroups:

- ARSDFBDT
- ARSDFDCT
- ARSDFDRT
- ARSDFDST
- ARSDFLIS
- ARSDFPPT
- ARSDFUOT
- ARSDFCRT
- ARSDFEML

Verify or replace the following parameter in the ALTER DATABASE command example:

ODinstance
The Content Manager OnDemand instance database name to which the ODF tables and filegroups belong.

Chapter 5. Creating ODF tables

After you successfully create any required table spaces or filegroups for ODF, use the instructions in this section to create the ODF tables.

About this task

Note: IBM i On IBM i, you do not need to create the ODF tables because they are created when ODF is installed.

To create the ODF tables on your Content Manager OnDemand server, follow these steps:

Procedure

1. Set up the environment:
   a) Windows For Windows: Open the OnDemand Command Window V10.5.
   b) z/OS For z/OS: Log in to OMVS. Switch to the super user (su). Export the environment variable by entering this command:
   `export DSNAOINI="/etc/ars/cli.ini"`
   c) z/OS Linux AIX For AIX, Linux, and z/OS: Change to the Content Manager OnDemand executable directory by entering this command:
   `cd /executabledir` where *executabledir* is the directory that contains your Content Manager OnDemand executable programs.
2. Run the ARSDB program by entering this command:
   `arsdb -I instance -O -rstv` where *instance* is your Content Manager OnDemand instance name.

Results

The ARSDB program generates a series of messages similar to the following:

```
ARS4026I Creating table ARCHIVE.arsdfbdt
ARS4034I Creating index arsdfbdt_idx1
ARS4034I Creating index arsdfbdt_idx2
ARS4034I Creating index arsdfbdt_idx3
ARS4034I Creating index arsdfbdt_idx4
ARS4034I Creating index arsdfbdt_idx5
ARS4026I Creating table ARCHIVE.arsdfcrt
ARS4034I Creating index arsdfcrt_idx1
ARS4034I Creating index arsdfcrt_idx2
ARS4026I Creating table ARCHIVE.arsdfdct
...
ARS4026I Inserting default values into table ARCHIVE.arsdfeml
ARS4034I Creating index arsdfeml_idx1
ARS4026I Creating table ARCHIVE.arsdflis
ARS4034I Creating index arsdflis_idx1
ARS4034I Creating index arsdflis_idx2
```

Chapter 6. Configuring ODF

About this task

Configure ODF for use with Content Manager OnDemand by following the instructions in this section.

Verifying the ARS.INI file

About this task

z/OS Windows Linux AIX The ODF DB2 table spaces and tables reside in the same database as the Content Manager OnDemand table spaces and tables. Verify that the contents in the ARS.INI file are correct for both Content Manager OnDemand and ODF.

IBM i On IBM i, you must add an entry to the ARS.INI file for each instance that uses ODF. The entry specifies the SRVR_ODF_CFG parameter, which defines the IFS path that Content Manager OnDemand uses to locate the ODF configuration file named arsodf.xml. The entry should look similar to this:

```
SRVR_ODF_CFG=/QIBM/UserData/OnDemand/instance/arsodf.xml
```

where *instance* is the name of your Content Manager OnDemand instance. You must end and restart the instance server after you update the ARS.INI file.

See the Content Manager OnDemand installation or configuration guide for your platform for more details about the content of the ARS.INI file.

Verifying the ARS.CFG file

The ARS.CFG file (or Windows registry if you are running ODF on a Windows server) contains parameters that are related to customizing ODF for your environment. The ARS.CFG file or Windows registry might contain parameters other than the parameters that are described in the following topics. However, the parameters that are described in the following topics are the only parameters that apply to ODF.

About this task

The following is an example of the ODF contents in the ARS.CFG file on Multiplatforms servers:

```
ARSODF_CONTSLEEP=3600
ARSODF_DISTSLEEP=60
ARSODF_SCHDSLEEP=180
ARSODF_PRINTTASKS=2
ARSODF_FILE_LOCATION=/odf/distribution/out
```

The following is an example of the ODF contents in the ARS.CFG file on z/OS:

```
ARSODF_CONTSLEEP=3600
ARSODF_DISTSLEEP=60
ARSODF_SCHDSLEEP=180
ARSODF_PRINTTASKS=2
ARSODF_PROC=ARSRPRNB
ARSODF_MAXJOBS=5
ARSODF_JOBSLEEP=300
ARSODF_JOBSPACE=0
ARSODF_DATASET_HLQ=ODFTEST
```

The following is an example of the ODF contents in the ARS.CFG file on IBM i:

```
ARSODF_CONTSLEEP=3600
ARSODF_DISTSLEEP=60
ARSODF_SCHDSLEEP=180
ARSODF_PRINTTASKS=2
ARSODF_FILE_LOCATION=/odf/distribution/out
```

Configuring ARS.CFG parameters

You can use various ARS.CFG file parameters to customize ODF for your installation.

For ODF running on all platforms except Windows, you modify these parameters by editing the ARS.CFG file. For ODF running on a Windows server, these parameters are updated by using the OnDemand Configurator.

Table 1. Supported parameter case in ARS.CFG

Operating system	Supported parameter case in ARS.CFG
Multiplatforms servers	Parameters must be specified in uppercase.
z/OS servers	Parameters can be specified in uppercase, lowercase, or mixed case.
IBM i servers	Parameters must be specified in uppercase.

If you are upgrading from a previous version of ODF, review the list of ARS.CFG file parameters that are no longer supported.

Task management

ARS_JAVA_CMD
Specifies the installed Java™ executable program on your Content Manager OnDemand server. For example: `ARS_JAVA_CMD=/usr/java7_64/jre/bin/java`

ARS_USER_EXITS_DIR
Specifies the directory where your compiled user exit programs are located.

z/OS **ARSODF_BANNER_ALLOC_PARMS**
For z/OS, specifying a value for the ARSODF_BANNER_ALLOC_PARMS parameter causes all banner data to be created in a sysout data set that is separate from the report. The CLASS, DEST, FORMS, and WRITER parameter values are inherited from the distribution, or from the report bundle definitions if specified. If the ARSODF_BANNER_ALLOC_PARMS parameter is not specified, the banner is created in the same sysout as the report.

z/OS **ARSODF_DATASET_HLQ**
For z/OS, specifies the high-level qualifier to use for non-GDG data set allocation if a **File name location** is not specified in the distribution or report bundle definition. The default value is ARSODF.

z/OS **ARSODF_DEFAULT_VARIABLE_LRECL**
For z/OS, specifies the logical record length (LRECL) value passed to the open routine when JES opens the sysout data set. The default value that is passed to the open routine is 32756. The LRECL that JES actually allocates might be different than this value.

ARSODF_DISTRIBUTION_THRESHOLD
Specifies the maximum number of distributions that ODF can process at one time. The default value is 0, which indicates that there is no maximum.

ARSODF_EXPIRE_DAYS
ODF routine maintenance removes all entries from the ARSDFPPT, ARSDFDRT, and ARSDFDST ODF operational tables that are older than the number of days specified. If the ARSODF_EXPIRE_DAYS parameter is not specified, then the ARSMAINT program uses 30 days as the default value.

IBM i Windows Linux AIX **ARSODF_FILE_LOCATION**
For any Multiplatforms or IBM i server, specifies a default path name for a distribution with a Location specified as **File** if a **File name location** is not specified in the distribution or report bundle definition.

If the ARSODF_FILE_LOCATION parameter is not specified, then the path that is specified by the ARS_TMP parameter in the ARS.CFG file is used.

z/OS **ARSODF_JESS**
For z/OS, JESS must be a string that identifies the JES subsystem name (4 characters). This parameter is used with the Address Space Create dynamic JOB START facility. Valid options are:

- JES2
- JES3
- No entry

An invalid entry is indicated by an error display message, and the default is applied.

z/OS **ARSODF_JOB_TRACEDIR**
For z/OS, specifies the directory name that contains the trace file if trace is active in ARSODF and when **Job Name** is specified in the distribution definition.

z/OS **ARSODF_JOBSPACE**
For z/OS, indicates how the newest address spaces are classified. A value of 1 specifies that address spaces are assigned the normal job space attribute. A value of 0 specifies that address spaces are assigned the system address space attribute. The default is 0.

This parameter allows address spaces to be classified as normal job spaces rather than as system address spaces. The difference is that system address spaces are not displayed in response to the D A,L operator command. Normal job spaces are displayed.

z/OS **ARSODF_MANIFEST_ALLOC_PARMS**
For z/OS, specifies allocation parameters to use when a manifest is created in a sysout data set that is separate from the report. The CLASS, DEST, FORMS, and WRITER parameter values are inherited from the distribution, or from the report bundle definitions if specified. If the ARSODF_MANIFEST_ALLOC_PARMS parameter is not specified, the manifest is created with the same allocation parameters as the report.

z/OS **ARSODF_MAXJOBS**
For z/OS, enter a number from 1 through 99. Specifies the number of concurrent print processes that are allowed to be started as submitted jobs. A conservative setting for this parameter is 5. An aggressive setting for this parameter is between 10 and 20.

z/OS **ARSODF_ODFMSG_ALLOC_PARMS**
For z/OS, specifies the output parameters used to allocate the JES spool file data set for the ODF Message file. The default value is 'CLA(A)'

ARSODF_PRINTTASKS
Specifies the number of parallel report bundle processor subtasks that are attached. The value can be 1 - 15 numeric digits. For most installations, this value should never exceed 40.

z/OS **ARSODF_PROC**
For z/OS, this parameter names the procedure to be invoked whenever the distribution requirement calls for a print process to be executed as a submitted job. The procedure can be stored in SYS1.PROCLIB, or it can be stored in some other proclib that is known to the Master Scheduler. You can customize the ARSRPSUB procedure that is provided in ARS.V10R5M0.SARSINST.

ARSODF_TMP
Specifies the directory ODF should use for temporary files that are created during the document retrieval process. The default value is `/tmp`.

Timing controls

ARSODF_CONTSLEEP
Must be followed by 1 - 15 numeric digits to specify the time, in seconds, that the continuation thread waits between processing.

The default for this keyword is 600 seconds, which is 10 minutes. The continuation sleep setting requires experimental adjustment based on the nature of distribution scheduling methods that are

established in the distribution create process. A high volume of **Time of Print** distribution scheduling methods can dictate a more frequent need for processing.

ARSODF_DISTSLEEP
Must be followed by 1 - 15 numeric digits to specify the time, in seconds, that the distribution thread waits between one pass of submitting distributions for report bundle processing and the next.

The default for this keyword is 300 seconds, which is 5 minutes. This keyword controls the downtime between scheduling distributions for report bundle processing. All scheduled distributions are analyzed for available reports to submit for report bundle processing, so it is necessary to process distributions in a timely manner.

Tip: Three hundred seconds is a good place to start. A high volume of distributions might dictate a much faster setting, perhaps less than one minute. Despite a faster distribution setting, when there is no work to be done the cycle becomes inactive quickly.

z/OS **ARSODF_JOBSLEEP**
For z/OS, must be followed by 1 - 15 numeric digits to specify the time, in seconds, that the address space task is inactive when there is no outstanding work for it to perform.

The default for this keyword is 300 seconds, which is 5 minutes.

ARSODF_SCHDSLEEP
Must be followed by 1 - 15 numeric digits to specify the time, in seconds, that the scheduler thread is inactive when there is no outstanding work for it to perform.

The default is 300 seconds, which is 5 minutes. This keyword determines the frequency of the distribution scheduling process. The scheduler thread is responsible for scheduling distributions with a scheduling method of **All Ready**, **Time of Day**, and **Time of Print**. If most of the distributions are of the **Loaded** distribution scheduling method type, then the scheduler can run infrequently, for example, hourly or even every two hours. The **All Ready**, **Time of Day**, and **Time of Print** distribution scheduling methods require more frequent scheduling.

OAM parameters

z/OS **ARSODF_OAM_PLAN**
For z/OS, specifies the name of the OAM DB2 application plan to be used. The operand string is required to be a non-null character string with no embedded blanks, less than or equal to 8 characters in length. The default is CBRIDBS.

z/OS **ARSODF_OAM_SLOW_TASKS**
For z/OS, determines the number of task control blocks (TCBs) started to handle connections to OAM for retrievals from objects with a slow retrieval time as defined by the ARSODF_OAM_SLOW_THRESHOLD parameter. If the value specified for this parameter is 0, no TCBs are dedicated for slow retrievals and all retrievals are processed by the TCBs that are associated with the ARSODF_OAM_TASKS parameter.

The default is 0. The ARSODF_OAM_SLOW_TASKS TCBs are in addition to the ARSODF_OAM_TASKS TCBs and use additional DB2 connections.

z/OS **ARSODF_OAM_SLOW_THRESHOLD**
For z/OS, specifies the threshold at which OAM retrievals are processed by the TCBs that are associated with the ARSODF_OAM_SLOW_TASKS parameter. If the estimated retrieval time for an object (as indicated by QELQERRT) is greater than or equal to the value of the ARSODF_OAM_SLOW_THRESHOLD parameter, the OSREQ RETRIEVE is processed by an ARSODF_OAM_SLOW_TASKS TCB.

The default value is 0. See the *Object Access Method Application Programmer's Reference* for other valid QELQERRT values. An ARSODF_OAM_SLOW_THRESHOLD value of 0 along with a nonzero ARSODF_OAM_SLOW_TASKS value causes all OAM retrieve requests to be processed by the ARSODF_OAM_TASKS TCBs.

z/OS **ARSODF_OAM_TASKS**
For z/OS, specifies the number of OAM service threads that need to be established. The operand should be a non-negative decimal integer, less than or equal to 30. A value of 0 implies that OAM services initialization should not be performed. The default is 0.

No longer supported

The following parameters are no longer required or supported. If these parameters are specified, they are ignored.

ARSODF_DOC_RETRIEVAL_TRACE
ARSODF_NUM_OAMSRVR
ARSODF_OAMSSID
ARSODF_ODFMSG_MANAGER
ARSODF_PDF_DIRECTORY
ARSODF_PRINT_PROCESSING_TRACE
ARSODF_REMOTE_DB
ARSODF_SECURITY_EXIT_FLAG
ARSODF_SECURITY_EXIT_PROGRAM
ARSODF_TEST
ARSODF_TRACE_REPORTID

Verifying the ARSODF.XML file

Review the following tasks to determine if you need to update the ARSODF.XML file.

About this task

The ARSODF.XML file is required if you send any emails as part of your ODF processing, or if you need to set any of the following ODF configuration details. The ARSODF.XML file is used as input to the ODFProcessDist.java program.

- Customize the details and format of the outgoing emails that contain distributions when the distribution **Location** value is set to **E-mail** or for any **Location** value with the **Notify by e-mail** check box selected. For each distribution location type, you can customize the email content and the maximum size for email attachments within a single email.
- Specify your SMTP server name.
- Specify whether or not to enable the secure sockets layer (SSL) when using the SMTP server to send mail.
- Specify trace parameters.
- IBM i Windows Linux AIX On Content Manager OnDemand for Multiplatforms or IBM i servers, specify the name of the command and parameters to use to submit ODF print requests and the name of the printer queue to use.

z/OS Windows Linux AIX The ARSODF.XML file is located in the `config` directory on your Content Manager OnDemand server.

IBM i For new instances that are created after Content Manager OnDemand Version 7.3 is installed, the ARSODF.XML file is located in the `/QIBM/UserData/OnDemand/[instance]` directory on your Content Manager OnDemand server, where *[instance]* is the name of your Content Manager OnDemand instance. For instances that were created prior to version 7.3, the ARSODF.XML file must be copied from the `/QIBM/ProdData/OnDemand/config` directory into the `/QIBM/UserData/OnDemand/[instance]` directory, where *[instance]* is the name of your Content Manager OnDemand instance.

For all IBM i servers, or to specify a different location for Multiplatforms or z/OS servers, add the **SRVR_ODF_CFG** parameter to the stanza for your ODF instance in the ARS.INI file. For example, on an IBM i server, add the following to your ARS.INI file:

```
SRVR_ODF_CFG=/QIBM/USERDATA/ONDEMAND/[instance]/ARSODF.XML
```

where [instance] is the name of your Content Manager OnDemand instance.

Configuring ARSODF.XML elements

You can use various XML elements to customize ODF for your installation.

General ODF settings

<TraceLevel>
Overrides any trace level setting specified on the command line. Specify this when you are having problems with the process distribution command. Possible values are **SEVERE**, **WARNING**, **INFO**, **FINE**, **FINER**, **FINEST**, and **OFF**.

<SMTPServer>
Server address for outgoing email. Enter the IP address or hostname of a system running SMTP, that is, the system that is actually able to send outgoing email. For example, mailserver.yourcompany.com. The SMTP server address is usually defined on the Content Manager OnDemand library server.

<SSL>
Whether or not to enable the secure sockets layer (SSL) when using the SMTP server to send mail. Possible values are **true** and **false**.

Report bundle print processing

IBM i Windows Linux AIX **<Printer name>**

Name of the printer to use for distribution location type of **Print**. In many cases, you should leave this value set to DEFAULT for Multiplatforms servers, and DEFAULT_I for IBM i servers. The sample version of the ODFProcessDist.java program that is shipped with Content Manager OnDemand will look for a printer name of DEFAULT or DEFAULT_I, depending on the platform, in the ARSODF.XML file. If you need to specify more than one printer name and associated parameters for your **Print** distributions, you can customize the ODFProcessDist.java program and ARSODF.XML file to support this. For example, you could specify the printer name in the Customer Variables or Exit Information fields in the distribution definition and modify the ODFProcessDist.java program to use that name when retrieving the printer definition from the ARSODF.XML file.

For more information on customizing the ODFProcessDist.java program, see “ODFProcessDist.java - Processed distribution exit” on page 61.

IBM i Windows Linux AIX **<Command>**

Name of the command to use to submit the print job for distribution location type of **Print**.

IBM i For IBM i, specify ARSPRT400 for this element in the DEFAULT_I section of the ARSODF.XML file.

IBM i Windows Linux AIX **<Parameter>**

One or more arguments to be passed to the print command named by the <Command> element. If you need to pass an argument without a value, use an empty <Parameter> element. For example, to pass the –x argument without a value, you would specify:

```
<Parameter arg="-x"></Parameter>
```

instead of this example, which passes –x 133 to the print command.

```
<Parameter arg="-x">133</Parameter>
```

IBM i For IBM i, update the -o parameter argument for the OUTPUTQUEUE parameter name in the DEFAULT_I section of the ARSODF.XML file as shown:

```
<Parameter name="OUTPUTQUEUE" arg="-o">QGPL/QPRINT</Parameter>
```

where QGPL is the library name and QPRINT is the output queue name to receive distributions with location type of **Print**.

Important: IBM i In the DEFAULT_I printer definition and any others you might create for IBM i, there are <DocParameter> and <Type> elements that must not be changed. If you require only one output queue for the destination for your ODF distributions, you can use the DEFAULT_I printer name as shipped with ODF. The only detail in the ARSODF.XML file that should be changed is the output queue name specified by the OUTPUTQUEUE argument. If you need to create multiple ODF print destinations to specify different output queues, you must copy the entire DEFAULT_I stanza to duplicate it. Then, change the printer name element to specify a unique name that you will reference in your customized `ODFProcessDist.java` program, and change the OUTPUTQUEUE parameter name argument to a unique output queue name.

Email customization

<Threshold>
Maximum number of bytes for email attachments within a single email. Sample program ships with **500000**.

<From>
This is the From line used for all email notifications. Sample program ships with **noreply@yourcompany.com**.

<Subject>
This is the Subject line used for all email notifications. You can override this value to make it unique for each type of distribution if needed. See the ARSODF.XML sample file and the customizable parameters in the following table.

<Content>
This is the content of the body of the email. You can make it unique for each type of distribution if needed. See the ARSODF.XML sample file and the customizable parameters in the following table.

Each distribution type (**Location** value set to **Email**, **Print**, **File**, or **None** with **Notify by email** check box selected) can have a unique subject line and customized email content containing any of the values shown in the following table.

Table 2. Email customization values

Field name	Explanation
For 'Subject' line, the following parameters can be used:	
%email_count%	When multiple attachments are being sent for a distribution and their sizes are greater than the Threshold value, they are split across multiple emails. This value indicates what email number this is in relation to the overall number of emails sent.
%email_total%	When multiple attachments are being sent for a distribution and their sizes are greater than the Threshold value, they are split up across multiple emails. This value indicates the total number of emails sent for this distribution.
%distribution_name%	This is the name of the distribution.

Table 2. Email customization values (continued)	
Field name	**Explanation**
%recipient%	This is the name of the recipient for the distribution.
For 'Content' of email, the following parameters can be used:	
%distribution_name%	This is the name of the distribution.
%recipient%	This is the name of the recipient for the distribution.
%attachment_begin%	This parameter separates the HTML body header from the attachment text. This header text will only be displayed once at the beginning of the email.
%attachment_end%	This parameter separates the attachment text from the HTML footer. This footer text will only be displayed once at the end of the email.
%attachment_number%	This parameter must be placed within the attachment text and will be replaced with the attachment number.
%attachment_size%	This parameter must be placed within the attachment text and will be replaced with the attachment size.
%report_begin%	This parameter allows for multiple report IDs to be listed. This parameter must be placed within the attachment text.
%report_end%	This parameter allows for multiple report IDs to be listed. This parameter must be placed within the attachment text.
%report_id%	This parameter must be placed within the report text and will be replaced with the report ID for the bundle. There could be more than one bundle for each file.
%report_sequence%	This parameter must be placed within the report text and will be replaced with the report sequence number for the bundle. There could be more than one bundle for each file.
%file_name%	This parameter must be placed within the attachment text and will be replaced with the name of the file.

Example ARSODF.XML files

The following example ARSODF.XML files show some of the XML elements available.

Defining default settings for email, print, and notifications

See “Configuring ARSODF.XML elements” on page 20 for the parameterized text values available to include in the email subject line or email body content.

This portion of the ARSODF.XML is an example of some of the default settings that might be defined on a Multiplatforms server.

```
<?xml version="1.0" encoding="utf-8"?>
<ODFSettings>
  ...
  <TraceLevel>OFF</TraceLevel>
  <Printers>
    <Printer name="DEFAULT">
      <Command>lpr</Command>
      <Parameters>
        <Parameter arg="-P">ODFPrintQueue</Parameter>
        <Parameter arg="-S">your.print.server.company.com</Parameter>
      </Parameters>
    </Printer>
  </Printers>
  <SMTPServer>your.smtp.server.company.com</SMTPServer>
  <SSL>false</SSL>
  <From>noreply@yourcompany.com</From>
  <Threshold>500000</Threshold>
  ...
</ODFSettings>
```

This example is a portion of the DEFAULT_I stanza in the ARSODF.XML file that shows some of the default settings that might be defined on an IBM i server. On IBM i, ARSPRT400 should always be used for the <Command> value, and the OUTPUTQUEUE parameter name with the –o argument should always be used to specify a library name (such as QGPL in the example) and an output queue name (such as QPRINT in the example) for distributions with location type of **Print**. If you only need to specify one output queue for your distributions with location type of **Print**, no other lines should be changed in the ARSODF.XML file. If you need to customize the ODFProcessDist.java program to use multiple printer definitions, you must make one additional change. You must add multiple <Printer> elements, specifying a unique printer name for each element.

```
<?xml version="1.0" encoding="utf-8"?>
<ODFSettings>
  ...
  <TraceLevel>OFF</TraceLevel>
  <Printers>
    <Printer name="DEFAULT_I">
      <Command>ARSPRT400</Command>
      <Parameters>
  ...
        <Parameter name="OUTPUTQUEUE" arg="-o">QGPL/QPRINT</Parameter>
  ...
      </Parameters>
    </Printer>
  </Printers>
  <SMTPServer>your.smtp.server.company.com</SMTPServer>
  <SSL>false</SSL>
  <From>noreply@yourcompany.com</From>
  <Threshold>500000</Threshold>
  ...
</ODFSettings>
```

Defining default email templates for attachments

This example shows a portion of the ARSODF.XML file that defines a default email template for sending distributions as email attachments.

```
<?xml version="1.0" encoding="utf-8"?>
<ODFSettings>
  ...
<EmailAttachmentNotify>
    <Subject>OnDemand Distribution Facility - %email_count% of %email_total%</Subject>
    <Content><![CDATA[
      <h2>Your documents are attached for distribution: %distribution_name%</h2>
      <br/>
      %attachment_begin%
      <h3>Attachment #%attachment_number% of %attachment_size% bytes includes:</h3>
      %report_begin%
      <h3>Report Id=%report_id%, Sequence=%report_sequence%</h3>
      %report_end%
      %attachment_end%
```

```
      <br/>
      ]]>
    </Content>
  </EmailAttachmentNotify>
  ...
</ODFSettings>
```

Verifying the ARSXFORM.XML file

Review the following information to determine if you need to update the ARSXFORM.XML file.

About this task

The ARSXFORM.XML file is used to enable data transforms for your ODF distributions. This file is required only if you have selected one of the **Transform** values in a distribution or report bundle definition in the Content Manager OnDemand Administrator client or specified one of the **transform** values in your XML for your distributions or report bundles.

The XML file contains the configuration information for different transform styles provided by third party transforms. Most of the third party transforms need the same set of input parameters that are used during transforming data from one format to another. However, the parameters or options of the program are not always the same. The purpose of the Content Manager OnDemand transform interface is to provide a mapping between the parameters or options required by the third party transform program and the available values that are stored in the Content Manager OnDemand application definition related to viewing of the data.

z/OS Windows Linux AIX The ARSXFORM.XML file is located in the `config` directory on your Content Manager OnDemand server.

IBM i The ARSXFORM.XML file must be copied from the `/QIBM/ProdData/OnDemand/config` directory into the `/QIBM/UserData/OnDemand/config` directory.

Configuring ARSXFORM.XML elements

You can use the following XML elements to enable data transforms for your ODF distributions.

Parent element

Each transform is uniquely defined by the <TransformName> element. Each transform definition used by ODF contains elements where the text value is not editable. There are also elements where the user provides the element text value required by the transform. Element names are case sensitive. Some elements might be found in the sample that are not used by ODF. ODF uses only the following elements:

<transform>
This is an empty element that contains the definition and characteristic of a transform.

Child elements

<TransformName>
This is a text element. The value is the transform name which identifies a transform program. The naming convention follows this format: OD_*input type*2*output type*, where *input type* is the data type of the document to be transformed, and *output type* is the data type into which the data is transformed. Some transform names include **OD_LINE2PDF**, **OD_LINE2UTF8**, **OD_AFP2PDF**, and **OD_AFP2UTF8**. For example:

- ODF looks for this specific line in the ARSXFORM.XML file whenever a transform is needed to transform report output from LineData to PDF:

  ```
  <TransformName>OD_LINE2PDF</TransformName>
  ```

- ODF looks for this specific line in the ARSXFORM.XML file whenever a transform is needed to transform report output from AFP to PDF:

```
<TransformName>OD_AFP2PDF</TransformName>
```

<TransformDescription>
This is a text element. The value is the description of the transform and can be changed.

<CmdParms>
This is an empty element with children text elements. The children text elements can be used to map transform options to the Content Manager OnDemand document specifications.

The following are the text elements. The value is the transform program parameter that Content Manager OnDemand uses to pass the specified information to the transform program command line. See the OnDemand Administrator help text for the **Field Information** page in the application definition for more information on many of these fields.

- <AG_NAME/>

  Specify which transform program parameter to use to pass the application group name to the transform program.

- <APPL_NAME/>

  Specify which transform program parameter to use to pass the application name to the transform program.

- <RECORDFORMATFIXED/>

  Specify which transform program parameter to pass to the transform program if **RECFM** is set to **Fixed** in the Content Manager OnDemand application definition.

- <RECORDFORMATVARIABLE/>

  Specify which transform program parameter to pass to the transform program if **RECFM** is set to **Variable**.

- <RECORDFORMATSTREAM/>

  Specify which transform program parameter to pass to the transform program if **RECFM** is set to **Stream**.

- <RECORDLENGTH/>

  Specify which transform program parameter to use to pass the record length of the document to the transform program.

- <CARRIAGECONTROL/>

  Specify which transform program parameter to use to pass the carriage control of the document to the transform program.

- <DOCROTATION/>

  Specify which transform program parameter to use to pass the rotation of the document to the transform program.

- <CODEPAGE/>

  Specify which transform program parameter to use to pass the code page of the document to the transform program.

- <LINEDELIMITER/>

  Specify which transform program parameter to use to pass the line delimiter of the document to the transform program.

- <LINEPERPAGE/>

  Specify which transform program parameter to use to pass the lines per page of the document to the transform program.

- <TRCYES/>

Specify which transform program parameter to pass to the transform program if **TRC** is set to **Yes** in the Content Manager OnDemand application definition.

- <TRCNO/>

  Specify which transform program parameter to pass to the transform program if **TRC** is set to **No**.

- <PRMODENONE/>

  Specify which transform program parameter to pass to the transform program if **PRMODE** is set to **None** in the Content Manager OnDemand application definition.

- <PRMODESOSI1/>

  Specify which transform program parameter to pass to the transform program if **PRMODE** is set to **SOSI1**.

- <PRMODESOSI2/>

  Specify which transform program parameter to pass to the transform program if **PRMODE** is set to **SOSI2**.

- <PRMODESOSI3/>

  Specify which transform program parameter to pass to the transform program if **PRMODE** is set to **SOSI3**.

- <CCANSI/>

  Specify which transform program parameter to pass to the transform program if **CC** is set to **Yes** and **CC Type** is set to **ANSI** in the Content Manager OnDemand application definition.

- <CCMACHINE/>

  Specify which transform program parameter to pass to the transform program if **CC** is set to **Yes** and **CC Type** is set to **Machine**.

- <CCNONE/>

  Specify which transform program parameter to pass to the transform program if **CC** is set to **No**.

- <INPUTFILE/>

  Specify which transform program parameter to use to pass the input file name to the transform program. This is a required child element of the <CmdParms> element. If your transform program does not require a parameter (such as -i) to be specified along with the input file name, leave this element value blank. This element might need to be specified as the last element, depending on your transform program requirements.

- <OUTPUTFILE/>

  Specify which transform program parameter to use to pass the output file name to the transform program. This is a required child element of the <CmdParms> element.

<CmdLineExe>

This is a text element. The value is the full qualified path name of an executable transform program. The user is responsible for setting the text value.

For example, the <CmdLineExe/> entry for a transform program called line2pdf might look similar to the following. The actual path depends on the install path you choose when you install the transform.

- Linux AIX On an AIX or Linux server:

```
<CmdLineExe>/opt/line2pdf/line2pdf</CmdLineExe>
```

- Windows On a Windows server:

```
<CmdLineExe>c:\\opt\\line2pdf\\line2pdf.exe</CmdLineExe>
```

- z/OS On a z/OS server:

```
<CmdLineExe>/opt/line2pdf/line2pdf</CmdLineExe>
```

- IBM i On an IBM i server:

```
<CmdLineExe>/transforms/line2pdf/line2pdf</CmdLineExe>
```

<Passthru>

This is an empty element with a child text element, <cmdlineparm/>. These values are passed directly to the transform program. You can specify command parameters and associated option values if the options cannot be matched with the Content Manager OnDemand values.

ODF will not parse or validate the text value of this <cmdlineparm/> element. The text value will be passed directly to the third party transform program.

Example transform setup

Use the following information as an example of how to setup a transform for ODF to use. The example uses a transform from line data to PDF.

Transform from line data to PDF

A sample transform that converts line data into a PDF file might look like this:

```
/opt/line2pdf>./line2pdf
Converts line data into a PDF file.
Usage:    line2pdf [options] {input.txt} output-file < input-file
Options:  -r       replace existing output file
          -n nnn   set the lines per page to nnn
          -lm nnn  set the maximum line length
          -x  nnn  choose the type of carriage control
          -l  nnn  choose a fixed line length
          -a  nnn  choose a code page
          -ag      application group name
          -an      application name
          -af      full path to the configuration file
```

You should test your transform program outside of ODF to learn the behavior and requirements of the program. For example, your transform program might require certain options to be specified in a certain sequence, or require that the input file and output file be specified as the last two parameters of the command.

How to use the <CmdParms> element

Sample mappings between the Content Manager OnDemand available values and the sample transform command line parameters:

```
<CmdParms>
      <AG_NAME>-ag</AG_NAME>
      <APPL_NAME>-an</APPL_NAME>
      <CARRIAGECONTROL>-x</CARRIAGECONTROL>
      <RECORDLENGTH>-lm</RECORDLENGTH>
      <CODEPAGE>-a</CODEPAGE>
      <OUTPUTFILE>-o</OUTPUTFILE>
      <INPUTFILE> </INPUTFILE>
    </CmdParms>
```

Before the transform process begins, ODF extracts the requested information from the Content Manager OnDemand application definition, and then maps the fields and generates a command line to send to the transform process. See the View Information page of the application definition for the type of information that can be requested. The transform process will match up the mappings:

- The application group name with option -ag as: -ag <ag_name>
- The application name with option –an as: -an <app_name>
- The carriage control CCType with option –x as: -x 0

- and so on

The transform process will ignore the mapping if a value is not available in the Content Manager OnDemand application view info.

How to use the Passthru <cmdlineparm> element

Passthru is used for mismatched options or options that are not available in the Content Manager OnDemand repository.

For example, you can use Passthru if a configuration file is required on the transform program command line. Passthru can also be used to set various command line flags that are required by the transform program at run time.

Example: Use a configuration file (-af) and indicate that the transform output file should be overwritten (-r)

You can use the -af option to specify a configuration file for additional transform options, and the -r option to indicate that the transform output file should be overwritten. For example:

```
<Passthru>
        <cmdlineparm>-r -af /opt/odtxf/line2pdf.cfg</cmdlineparm>
</Passthru>
```

Refer to your transform documentation for details about creating the contents of the configuration file, such as the line2pdf.cfg file named in the example.

Example ARSXFORM.XML files

Examples of an ARSXFORM.XML file show the usage of some of the available XML elements.

ODF transform for Line data to PDF

In this example, a transform named OD_LINE2PDF is defined, which will call the transform program called od_arsxline_cmd defined by the <CmdLineExe> tag. In addition to those values, the od_arsxline_cmd program requires some additional information to be passed along on the command line. The -x P specified on the <cmdlineparm> tag has no meaning to Content Manager OnDemand, so it is just passed through and included on the command line call to the od_arsxline_cmd program.

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

  <transform>
    <TransformName>OD_LINE2PDF</TransformName>
    <TransformDescription>ARSXLINE Line Data Transform for OnDemand</TransformDescription>
    <OutputMimeType>application/pdf</OutputMimeType>
    <OutputExtension>pdf</OutputExtension>

    <CmdParms>
      <CCANSI>-a A</CCANSI>
      <CCMACHINE>-a M</CCMACHINE>
      <CCNONE>-a N</CCNONE>
      <CODEPAGE>-c</CODEPAGE>
      <LINEDELIMITER>-d</LINEDELIMITER>
      <RECORDFORMATFIXED>-f F</RECORDFORMATFIXED>
      <RECORDFORMATVARIABLE>-f V</RECORDFORMATVARIABLE>
      <RECORDFORMATSTREAM>-f S</RECORDFORMATSTREAM>
      <FONTNAME>-F</FONTNAME>
      <PAGEHEIGHTPIXELS>-h</PAGEHEIGHTPIXELS>
      <RECORDLENGTH>-l</RECORDLENGTH>
      <MARGIN>-m</MARGIN>
      <FONTSIZE>-n</FONTSIZE>
      <PRMODENONE>-p N</PRMODENONE>
      <PRMODESOSI1>-p 1</PRMODESOSI1>
      <PRMODESOSI2>-p 2</PRMODESOSI2>
      <PRMODESOSI3>-p 3</PRMODESOSI3>
      <LINEPERPAGE>-q</LINEPERPAGE>
      <LINEPERPRINTPAGE>-Q</LINEPERPRINTPAGE>
      <LINESPACINGFACTORDIV100>-s</LINESPACINGFACTORDIV100>
      <TRCNO>-t 0</TRCNO>
      <TRCYES>-t 1</TRCYES>
```

```
      <PAGEWIDTHPIXELS>-w</PAGEWIDTHPIXELS>
      <INPUTFILE>-i</INPUTFILE>
      <OUTPUTFILE>-o</OUTPUTFILE>
    </CmdParms>

    <CmdLineExe>od_arsxline_cmd</CmdLineExe>
    <OperationName>Line to PDF Exe</OperationName>
    <Passthru>
        <cmdlineparm>-x P</cmdlineparm>
    </Passthru>
  </transform>

</Transforms>
```

ODF transform for AFP data to PDF

In this example, a transform named OD_AFP2PDF is defined, which will call the transform program called afp2pdf defined by the <CmdLineExe> tag. From this configuration, the transform interface will recognize that the transform program requires OUTPUTFILE and INPUTFILE information from Content Manager OnDemand, and will set those values on the command line using the options specified in each related XML tag. In addition to those values, the afp2pdf program requires some additional information to be passed along on the command line. The -l specified on the <cmdlineparm> tag has no meaning to Content Manager OnDemand, so it is just passed through and included on the command line call to the afp2pdf program.

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

<transform>
    <TransformName>OD_AFP2PDF</TransformName>
    <TransformDescription>Ricoh AFP2PDF Transform</TransformDescription>
    <OutputMimeType>application/pdf</OutputMimeType>
    <OutputExtension>pdf</OutputExtension>

    <CmdParms>
      <RESOURCEFILE>-r</RESOURCEFILE>
      <OUTPUTFILE>-o</OUTPUTFILE>
    </CmdParms>

    <CmdLineExe>/opt/afp2pdf/afp2pdf</CmdLineExe>
    <OperationName>AFP2PDF Exe</OperationName>
    <Passthru>
        <cmdlineparm>-l</cmdlineparm>
    </Passthru>
  </transform>

</Transforms>
```

Chapter 7. Running ODF

About this task

On an AIX or Linux server, ODF runs as a daemon. On a Windows server, ODF runs as a service. On a z/OS server, ODF runs as a long running program. On an IBM i server, ODF can be run interactively or in batch, but in most cases it should be submitted to run as a batch job.

Using the ARSODF program

The ARSODF program periodically checks for ODF distributions to process.

About this task

By using the ARSODF program, you can start or terminate the main ODF processing program or perform specialized ODF functions such as running the external scheduling process or externally submitting a distribution for report bundle processing. See Chapter 12, “ARSODF,” on page 57 for more information about the syntax and parameters, and for example program calls for the ARSODF program.

The external scheduling process allows you to schedule distributions defined with a distribution scheduling method of **External**. The ODF main thread does not schedule distributions defined with a distribution scheduling method of **External**; you must initiate the scheduling request by using the ARSODF program or the Content Manager OnDemand Monitor. In previous versions of ODF, the external scheduling process was performed by running the ARSRDFGO program.

Externally submitting distributions for report bundle processing allows you to control both the scheduling of the distribution and the submission for report bundle processing. In previous versions of ODF, external processing was performed by running the ARSBDIST program. See “Externally submitting distributions” on page 52 for more information.

Procedure

1. To start the ARSODF program, run the following command:
   `arsodf -I instance -S`
   where *instance* is your Content Manager OnDemand instance name.
2. To stop the ARSODF program, run the following command:
   `arsodf -I instance -T`
   where *instance* is your Content Manager OnDemand instance name.
3. To run the ODF external scheduling process, run the following command:
   `arsodf -I instance -d distribname -r recipname`
   where *instance* is your Content Manager OnDemand instance name, *distribname* is the ODF distribution name, and *recipname* is the name of the ODF recipient or recipient list for the distribution.
4. To externally submit distributions for report bundle processing, run the following command:
   `arsodf -I instance -e -d distribname -r recipname`
   where *instance* is your Content Manager OnDemand instance name, *distribname* is the ODF distribution name, and *recipname* is the name of the ODF recipient for the distribution.

Example

AIX On an AIX server:

1. Change to the executable directory by running the following command:
   `cd /opt/IBM/ondemand/V10.5/bin/`
2. Run the ARSODF program.

Linux On a Linux server:

1. Change to the executable directory by running the following command:
   `cd /opt/ibm/ondemand/V10.5/bin/`
2. Run the ARSODF program.

Windows On a Windows server, choose one of the following options:

- Open a Content Manager OnDemand Command Prompt window and run the ARSODF program.
- Use the Content Manager OnDemand Configurator to run the ARSODF program:
  1. Select **Start** > **All Programs** > **IBM OnDemand Server V10.5** > **OnDemand Configurator V10.5**.
  2. Expand the server and select **Services**.
  3. Click the Content Manager **OnDemand Distribution Facility** service, then press the Start button (or toggle to stop).

z/OS On a z/OS server:

1. Make the appropriate JCL changes to member ARSODF in ARS.V10R5M0.SARSINST as required by your installation.
2. Copy ARSODF to SYS1.PROCLIB or some other procedure library that is known to the master scheduler.
3. Start the ARSODF program by issuing the command `/s ARSODF` or stop the ARSODF program by issuing the command `/p ARSODF` which runs the ARSODF program with the `-S` or `-T` as needed.

IBM i On an IBM i server:

- Submit the ARSODF program to batch within the QSHELL environment by running the following command, where the values specified by the `QSH CMD` parameter are the ARSODF parameters required to start, stop, run the external scheduling process, or externally submit distributions. The following example shows how to start ODF with the `-S` parameter:

  ```
  SBMJOB CMD(QSH CMD('ARSODF -I instancename -S')) JOBD(QRDARS/QOND400) JOBQ(*JOBD)
  ```

  where *instancename* is the name of your Content Manager OnDemand instance. You can optionally add the USER(*instancename*) parameter if you want ODF to be run by the instance user profile, which ensures that all the needed authority requirements are met.

Chapter 8. Planning ODF distributions

The following scenario and example data suggest possible ways to organize real-world data by focusing on business requirements and operations of the ExampleCo. Enterprises company. The information that is used to create the ExampleCo. Enterprises scenario is fabricated and created only to help explain key planning distribution functions.

To plan and organize your data, you need to complete the following tasks:

- Analyze your distribution requirements and determine the distribution scheduling methods and policies that best suit your needs.
- Identify the reports that will be distributed.
- Identify the selection criteria for each distribution.
- Identify recipients for each distribution.

When you identify and organize the data that you have, you can use the Content Manager OnDemand Administrator client or the ARSXML batch administration command to create distributions.

ExampleCo. Enterprises overview

ExampleCo. Enterprises is a multinational corporation that sells widgets to small and medium businesses. ExampleCo. Enterprises has 900 sales offices in 6 countries and 11,000 employees. One of the sales offices is located in San Francisco and covers the entire San Francisco Bay Area. Andy Smith is the regional manager of this sales office.

Every sales representative in ExampleCo. Enterprises submits a sales invoice after each sale is made. These invoices are stored into a single application group in Content Manager OnDemand.

San Francisco sales office business requirements

Five sales representatives who work in the San Francisco office have different roles, responsibilities, and business requirements that affect the way they plan for and define their distributions.

The following table describes the roles and responsibilities, and business requirement of each employee:

Table 3. ExampleCo. Enterprises's San Francisco office members, responsibilities, and business requirements

Employee	Responsibilities	Business requirement
Andy Smith	Regional manager. Andy manages four sales representatives.	Every morning at 8 a.m., Andy needs to receive a hardcopy of all the sales invoices for his team from the day before.
Thiago Ruiz	Sales person who covers accounts in north San Francisco.	Thiago wants to have the sales reports delivered to her own inbox as email attachments (one email attachment for each account) as soon as the reports arrive to prevent a lot of work coming in at once.
Nadina Aleksandra	Sales person who covers accounts in west San Francisco.	Nadina wants to receive hardcopy reports every day at 5 p.m.

Table 3. ExampleCo. Enterprises's San Francisco office members, responsibilities, and business requirements (continued)

Employee	Responsibilities	Business requirement
Sandeep Kapoor	Sales person who covers accounts in the central and Twin Peaks area.	Sandeep prefers to keep his sales invoices in a data set on his z/OS system. He wants to have a new data set generated every day. He wants to use SANDEEP as the high-level qualifier for his data sets.
Ellen Wu	Sales person who covers accounts in east San Francisco.	Ellen prefers to receive hardcopy reports and wants to be notified when a sales invoice becomes available.

Collecting information for the ODF system setup

In order to create the ODF objects to satisfy the business requirements, Andy gathers the details of the information that the team members need to be included in each of their ODF distributions.

Andy Smith collects the following information from his team and provides the information to the ODF administrator:

Report information
Report ID, application group, and application.

Recipient information
Information about the recipients including their recipient names, email addresses, and the region that is covered by each recipient.

Output requirements
The output that is delivered to each recipient based on the individual's business need.

The following tables summarize the report name, recipients, and output requirements that Andy has gathered.

Table 4. Report information

Report ID	Application group	Application
`Daily sales invoices`	`Daily sales invoices`	`Daily sales invoices`

Table 5. Recipient information

Employee name	Recipient name	Department number	Email address	Region
Andy Smith	`ASMITH`	`Dept. 777`	`asmith@exampleco.com`	`sf`
Thiago Ruiz	`TRUIZ`	`Dept. 777`	`truiz@exampleco.com`	`nsf` (North San Francisco)
Nadina Aleksandra	`NALEKSAN`	`Dept. 777`	`naleksan@exampleco.com`	`wsf` (West San Francisco)
Sandeep Kapoor	`SKAPOOR`	`Dept. 777`	`skapoor@exampleco.com`	`csf` (Central and Twin Peaks)
Ellen Wu	`ELLENWU`	`Dept. 777`	`ewu@exampleco.com`	`esf` (East San Francisco)

Table 6. Output requirements

Recipient	Report	Selection criteria	Type of output	Timing of output
Andy Smith	`Daily sales invoices`	Reports for Andy Smith, Thiago Ruiz, Nadina Aleksandra, Sandeep Kapoor, and Ellen Wu	Hardcopy	Every morning at 8 a.m.
Thiago Ruiz	`Daily sales invoices`	North San Francisco region	Email attachments	As soon as the reports arrive
Nadina Aleksandra	`Daily sales invoices`	West San Francisco region	Hardcopy	Every day at 5 p.m.
Sandeep Kapoor	`Daily sales invoices`	Central and Twin Peaks region	Files (with the value SANDEEP as the high-level qualifier of his z/OS data sets)	When both reports are ready, otherwise print whatever is ready at 5:30 p.m. every day
Ellen Wu	`Daily sales invoices`	East San Francisco region	Hardcopy	Ellen wants to be notified when a sales invoice becomes available

Creating the ODF objects

The Content Manager OnDemand administrator creates the report ID definitions and the recipient definitions according to the requirements provided by Andy.

The ODF administrator completes the following tasks by using the Content Manager OnDemand Administrator client or the ARSXML batch administration command to create the ODF objects based on the business requirements from Andy's team:

- Create a report ID by using the following values:

  Report ID
  `Daily sales invoices`

  Application group
  `Daily sales invoices`

  Application
  `Daily sales invoices`
- Add recipients for Andy and his team members by using the recipient information that Andy provided.

Distributions

Distributions provide an easy way to automatically group reports and portions of reports and distribute the reports to multiple users.

Based on the business requirements from Andy and his team, the ODF administrator creates the following distributions for Andy and his team members.

Table 7. Distribution values for Andy and his team members

Recipient name	Distribution name	Report ID	Query includes	Distribution scheduling method	Location
ASMITH	Daily sales invoices	Daily sales invoices	WHERE Sales_Person = 'TRUIZ' OR Sales_Person = 'NALEKSAN' OR Sales_Person = 'SKAPOOR' OR Sales_Person = 'EWU'	Time of Day: 08:00	Print
TRUIZ	Daily sales invoices	Daily sales invoices	WHERE Region = 'nsf'	Loaded	E-mail
NALEKSAN	Daily sales invoices	Daily sales invoices	WHERE Region = 'wsf'	Time of Day: 17:00	Print
SKAPOOR	Daily sales invoices	Daily sales invoices	WHERE Region = 'csf'	Time of Print: 17:30	File hlq(SANDEEP)
EWU	Daily sales invoices	Daily sales invoices	WHERE Region = 'esf'	All Ready	None (with Notify by e-mail check box checked)

Chapter 9. Administering ODF distributions

You can use the ARSXML batch administration program or the Content Manager OnDemand Administrator client to administer all of the objects related to your ODF distributions.

With either method, you work with the following objects:

- Recipients
- Recipient lists
- Report IDs
- Distributions
- Report bundles

To administer ODF objects, you must be a Content Manager OnDemand system administrator or a user with ODF Administration authority.

Recipients

Each person who will receive a distribution must be defined as a recipient.

Recipient lists

An ODF recipient list contains a group of individual recipients who receive the same distributions.

Report IDs

When reports are archived in Content Manager OnDemand for the application group or application that is identified by the report ID, each distribution that includes this report ID will be processed for possible distribution.

Distributions

When you create an ODF distribution, the values that you specify for the distribution fields can be inherited by the report bundle that is associated with the distribution.

To make the report bundle inherit values from the distribution, when you create the report bundle, leave those fields blank. For example, if you want the report bundle to inherit the Destination value from the distribution, when you create the report bundle, leave the Destination field blank. If you do not want the report bundle to inherit values from the distribution, then specify values for the report bundle fields when you create the report bundle. The values that you enter in the report bundle fields override the values from the distribution.

Report bundles

Report bundles are part of a distribution and are uniquely identified by the Distribution Name, Recipient/List, and Sequence number. Distributions are uniquely identified by the Distribution Name and Recipient/List.

When you create an ODF report bundle, the values for the report bundle fields can inherit values from the corresponding distribution fields of the distribution you selected. To make the report bundle inherit values from the distribution, when you create the report bundle, leave those fields blank. For example, if you want the report bundle to inherit the value of the Destination field of the distribution, when you create

the report bundle, leave the **Destination** field blank. If you do not want the report bundle fields to inherit values from the corresponding distribution fields, then specify values for the report bundle fields when you create the report bundle. The values that you enter in the report bundle fields override the values from the corresponding distribution fields.

For all optional parameters, you have two choices:

- Leave them blank. During processing, ODF inherits the values that you specified for these parameters when you created the distribution. If a value is changed in the distribution, the value in the report bundle does not need to be changed.
- Specify values for them. ODF uses the values that you specify for these parameters and overrides the values from the distribution.

Using the ARSXML program

To use the ARSXML program to create the ODF objects needed for your distributions, follow the instructions and use the examples in this section.

About this task

The ARSXML program provides a batch interface to add, update, export, and delete ODF objects by using XML. You can use the ARSXML schema to define the objects in your XML file. See the *Content Manager OnDemand Administration Guide* for your platform for more information on the ARSXML program and for the schema tables for each ODF object.

In the examples in the following sections, a z/OS server path is used to specify the path to the XML files, but you must use a path that is correct for your server.

To run the ARSXML program in each example, replace *myod* with your Content Manager OnDemand instance name, *myoduser* with a Content Manager OnDemand user that has system administrator or ODF administrator authority, *myodpwd* with the password for the user specified, and specify a file name that contains your XML.

The XML examples assume that the ondemand.xsd file is located in the same directory as the XML files. If the actual location of the `ondemand.xsd` file differs, then adjust the path accordingly.

Adding recipients

A Content Manager OnDemand administrator wants to add Andy Smith as a new ODF recipient.

The administrator created the `recipientAdd.xml` file that contains the recipient information.

To complete this task, run this command:

```
arsxml add -h myod -u myoduser  -p myodpwd -v  -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipientAdd.xml
```

The `recipientAdd.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

    <odfRecipient name="ASMITH"
                 description="San Francisco regional manager"
                 fullName="Mr. Andy Smith"
                 title="Regional Manager"
                 email="asmith@exampleco.com"
                 addr1="123 Company Way"
                 dept="Dept. 007"
                 building="Bldg. 3"
                 account="ACCT. SF001"
                 room="Room 674A"
                 banner="true"
                 header1="Hold for Andy"
                 header2="or mail to his home"
                 header3="after 7 business days" />
</onDemand>
```

Updating recipients

A new sales representative named Linn Thomas has recently joined the team and the Content Manager OnDemand administrator wants to correct the spelling of Linn's name and update the department number for all of Andy's department's recipient definitions.

The administrator created the `recipientUpdate.xml` file that contains the updated information.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipientUpdate.xml
```

The `recipientUpdate.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfRecipient name="LTHOMAS"
                 newName="LMTHOMAS"
                 dept="Dept. 777" />

   <odfRecipient name="TRUIZ"  dept="Dept. 777" />
   <odfRecipient name="NALEKSAN"  dept="Dept. 777" />
   <odfRecipient name="SKAPOOR" dept="Dept. 777" />
   <odfRecipient name="ELLENWU"   dept="Dept. 777" />

</onDemand>
```

Exporting recipients

A Content Manager OnDemand administrator wants to export Andy's department members' recipient definitions so that he can add them to a test system.

The administrator created the `recipientExport.xml` file that contains the information required for the export.

To complete this task, run this command:

```
arsxml export -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipientExport.xml
-o
    recipientExportOut.xml
```

The `recipientExport.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<odfRecipient name="ASMITH" />
<odfRecipient name="TRUIZ" />
<odfRecipient name="NALEKSAN" />
<odfRecipient name="SKAPOOR" />
<odfRecipient name="LMTHOMAS" />
<odfRecipient name="ELLENWU" />

</onDemand>
```

Deleting recipients

A Content Manager OnDemand administrator wants to delete Linn's recipient definition because he is no longer part of the company.

The administrator created the `recipientDelete.xml` file that contains the information required for the delete.

To complete this task, run this command:

```
arsxml delete -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipientDelete.xml
```

The `recipientDelete.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<odfRecipient name="LMTHOMAS" />

</onDemand>
```

Adding recipient lists

A Content Manager OnDemand administrator wants to create a new ODF recipient list.

The administrator created the `recipListAdd.xml` file that contains the recipient list information.

To complete this task, run this command:

```
arsxml add -h myod -u myoduser  -p myodpwd -v  -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipListAdd.xml
```

The `recipListAdd.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfRecipientList name="SALES-San Francisco" description="San Francisco sales team" >
      <odfRecipient name="ASMITH"     task="add" />
      <odfRecipient name="NALEKSAN"     task="add" />
      <odfRecipient name="SKAPOOR"     task="add" />
      <odfRecipient name="LMTHOMAS"     task="add" />
      <odfRecipient name="ELLENWU"     task="add" />
   </odfRecipientList>

</onDemand>
```

Updating recipient lists

A Content Manager OnDemand administrator needs to update the recipient list for Andy's department to add a new employee.

The administrator created a `recipListUpdate.xml` file, which contains the information for the new employee.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipListUpdate.xml
```

The `recipListUpdate.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfRecipientList name="SALES-San Francisco" >
       <odfRecipient name="TRUIZ"   task="add" />
   </odfRecipientList>

</onDemand>
```

Exporting recipient lists

A Content Manager OnDemand administrator wants to export the recipient list for Andy's department so that he can add it to a test system.

The administrator created the `recipListExport.xml` file that contains the information required for the export.

To complete this task, run this command:

```
arsxml export -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipListExport.xml
    -o recipListExportOut.xml
```

The `recipListExport.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfRecipientList name="SALES-San Francisco" />

</onDemand>
```

Deleting recipient lists

A Content Manager OnDemand administrator wants to delete the recipient list for Andy's department because the company has reorganized and the members of his sales team will be reassigned to other regions.

The administrator created the `recipListDelete.xml` file, which contains the information required for the delete action.

To complete this task, run this command:

```
arsxml delete -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/recipListDelete.xml
```

The `recipListDelete.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <odfRecipientList name="SALES-San Francisco" />

</onDemand>
```

Adding report IDs

A Content Manager OnDemand administrator wants to create a new ODF report ID.

The administrator created the `rptIdAdd.xml` file that contains the report ID information.

To complete this task, run this command:

```
arsxml add -h myod -u myoduser  -p myodpwd -v  -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptIdAdd.xml
```

The `rptIdAdd.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfReportId name="Daily sales invoices"
                reference="SALES SUMMARY - Daily of:"
                applicationGroup="Daily sales invoices"
                application="Daily sales invoices"
                status="Active" />

</onDemand>
```

Updating report IDs

A Content Manager OnDemand administrator needs to update a report ID to change its status to inactive, because the management team has decided not to distribute this particular report until some enhancements are made to the content.

The administrator created the `rptIdUpdate.xml` file that contains the information for the update.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptIdUpdate.xml
```

The `rptIdUpdate.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfReportId name="Daily sales invoices"
                status="Inactive" />

</onDemand>
```

Exporting report IDs

A Content Manager OnDemand administrator wants to export the report ID for the Daily sales invoices so that he can add it to a test system.

The administrator created the `rptIdExport.xml` file that contains the information required for the export.

To complete this task, run this command:

```
arsxml export -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptIdExport.xml
    -o rptIdExportOut.xml
```

The `rptIdExport.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfReportId name="Daily sales invoices" />

</onDemand>
```

Deleting report IDS

A Content Manager OnDemand administrator wants to delete the report ID for the current Daily sales invoices because a new report is being created to replace it.

The administrator created the `rptIdDelete.xml` file that contains the information required for the delete.

To complete this task, run this command:

```
arsxml delete -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptIdDelete.xml
```

The `rptIdDelete.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

     <odfReportId name="Daily sales invoices" />

</onDemand>
```

Adding distributions

When you create an ODF distribution, the values that you specify for the distribution fields can be inherited by the report bundle that is associated with the distribution. To make the report bundle inherit values from the distribution, when you create the report bundle, leave those fields blank. For example, if you want the report bundle to inherit the Destination value from the distribution, when you create the report bundle, leave the **Destination** field blank. If you do not want the report bundle to inherit values from the distribution, then specify values for the report bundle fields when you create the report bundle. The values that you enter in the report bundle fields override the values from the distribution.

A Content Manager OnDemand administrator wants to create a new ODF distribution. The administrator created the `distribAdd.xml` file that contains the distribution information.

To complete this task, run this command:

```
arsxml add -h myod -u myoduser  -p myodpwd -v  -i /usr/lpp/ars/V10R5M0/bin/xml/samples/distribAdd.xml
```

The `distribAdd.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

   <odfDistribution name="Daily sales invoices"
                    recipient="SALES-San Francisco"
                    status="Active"
                    job="MSALES"
                    location="E-mail"
                    transform="None"
                    account="ACCT. 0013"
                    cwIndicator="Wait"
                    reportBreak="Yes"
                    distMethod="Loaded"
                    destination="PRINTER3"
                    writer="WRITER3"
                    forms="FORM3"
                    formDef="FORMDEF3"
                    pageDef="PAGEDEF3"
                    class="A"
                    FCB="FCB3"
                    UCS="UCS3"
                    outputLimit="100" >
      <odfReportBundle task="add"
                       sequence="10"
```

```
                        reportId="DAILY SALES INVOICES"
                        reportBuild="Query"
                        status="Active"
                        waitIndicator="Wait"
                        location="E-mail"
                        transform="None"
                        destination="PRINTER6"
                        writer="WRITER6"
                        forms="FORM6"
                        formDef="FORMDEF6"
                        pageDef="PAGEDEF6"
                        job="XMLF0013"
                        class="A"
                        FCB="FCB6"
                        UCS="UCS6"
                        burst="Yes"
                        flash="FLASH6"
                        copies="1"
                        COPYM="COPYM6"
                        outputLimit="Unlimited" >
                        <sql>1;REPORT_NUMBER;'YE1234';Where REPORT_NUMBER = ?</sql>
  </odfReportBundle>
  </odfDistribution>

</onDemand>
```

Updating distributions

A Content Manager OnDemand administrator needs to update a distribution to change the way this distribution is delivered from email to print, in order to reduce the volume of data being sent across their network.

The administrator created the `distribUpdate.xml` file that contains the information for the update.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/distribUpdate.xml
```

The `distribUpdate.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

     <odfDistribution name="Daily sales invoices"
                      recipient="SALES-San Francisco"
                      location="Print" />

</onDemand>
```

Exporting distributions

A Content Manager OnDemand administrator wants to export the distribution and all its report bundles for the Daily sales invoices so that he can add them to a test system. The administrator created the `distribExport.xml` file that contains the information required for the export.

Tip: Exporting the distribution will export the distribution itself and all of the report bundles that belong to the distribution.

To complete this task, run this command:

```
arsxml export -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/distribExport.xml
    -o distribExportOut.xml
```

The `distribExport.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

     <odfDistribution name="Daily sales invoices"
                     recipient="SALES-San Francisco" />

</onDemand>
```

Deleting distributions

A Content Manager OnDemand administrator wants to delete the distribution for the Daily sales invoices because management has decide to redefine the set of reports sent out monthly.

The administrator created the `distribDelete.xml` file, which contains the information required for the delete.

Important: Deleting the distribution definition deletes all of the report bundle definitions that belong to it, without having to explicitly delete each report bundle.

To complete this task, run this command:

```
arsxml delete -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/distribDelete.xml
```

The `distribDelete.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

     <odfDistribution name="Daily sales invoices"
                     recipient="SALES-San Francisco" />

</onDemand>
```

Adding report bundles

When you create an ODF report bundle, the optional values for the report bundle fields can be inherited from the distribution to which this report bundle is associated.

To make the report bundle inherit values from a distribution, when you create the report bundle, leave those fields blank. For example, if you want the report bundle to inherit the **Destination** value from the distribution, when you create the report bundle, leave the **Destination** field blank. If you do not want the report bundle to inherit values from the distribution, then specify values for the report bundle fields when you create the report bundle. The values that you enter in the report bundle fields override the values from the distribution.

For all optional parameters, you have two choices:

- Leave them blank. During processing, ODF inherits the values that you specified for these parameters when you created the distribution. If a value is changed in the distribution, the value in the report bundle does not need to be changed.
- Specify values for them. ODF uses the values that you specify for these parameters and overrides the values from the distribution.

A Content Manager OnDemand administrator wants to create a new ODF report bundle. The administrator created the `rptBundleAdd.xml` file that contains the report bundle information.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v  -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptBundleAdd.xml
```

The `rptBundleAdd.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<odfDistribution name="Daily sales invoices"
                  recipient="SALES-San Francisco" >
   <odfReportBundle task="add"
                      sequence="20"
                      reportId="DAILY SALES INVOICES"
                      reportBuild="Full"
                      status="Active"
                      waitIndicator="Wait"
                      location="E-mail"
                      transform="None"
                      destination="PRINTER6"
                      writer="WRITER6"
                      forms="FORM6"
                      formDef="FORMDEF6"
                      pageDef="PAGEDEF6"
                      job="XMLF0006"
                      class="A"
                      FCB="FCB6"
                      UCS="UCS6"
```

```
                        burst="Yes"
                        flash="FLASH6"
                        copies="1"
                        COPYM="COPYM6"
                        outputLimit="Unlimited" />
    </odfDistribution>
</onDemand>
```

Updating report bundles

A Content Manager OnDemand administrator needs to update a report bundle so that the entire report is selected, instead of a subset that had been defined by an SQL query previously.

The administrator created the `rptBundleUpdate.xml` file that contains the information for the update.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptBundleUpdate.xml
```

The `rptBundleUpdate.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<odfDistribution name="Daily sales invoices"
                 recipient="SALES-San Francisco" >
      <odfReportBundle task="update"
                       sequence="20"
                       reportId="DAILY SALES INVOICES"
                       reportBuild="Full" />
   </odfDistribution>


</onDemand>
```

Exporting report bundles

A Content Manager OnDemand administrator wants to export a report bundle so that he can add it to a test system.

To do so, he must export the distribution to which the report bundle belongs. This action also exports all of the other report bundles that belong to the same distribution. See “Exporting distributions” on page 43 for command and a sample XML file to perform this task.

Deleting report bundles

A Content Manager OnDemand administrator wants to delete a report bundle that will no longer be used.

The administrator created the `rptBundleDelete.xml` file, which contains the information required for the delete action.

Tip: The ARSXML command uses the update function rather than the delete function. The command updates the distribution by removing a bundle.

To complete this task, run this command:

```
arsxml update -h myod -u myoduser  -p myodpwd -v -x -i /usr/lpp/ars/V10R5M0/bin/xml/samples/rptBundleDelete.xml
```

The `rptBundleDelete.xml` file might look like this:

```
<?xml version="1.0" encoding="UTF-8" ?>
<onDemand xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

     <odfDistribution name="Daily sales invoices"
                     recipient="SALES-San Francisco" >
         <odfReportBundle task="delete"
                          sequence="20"
                          reportId="DAILY SALES INVOICES" />
     </odfDistribution>

</onDemand>
```

Using the administration client

To use the Content Manager OnDemand Administrator client to administer the ODF objects needed for your distributions, follow the instructions in this section.

About this task

The Content Manager OnDemand Administrator client provides a GUI interface to add, update, export, and delete ODF objects. See the Content Manager OnDemand Administrator client online help for details about the fields on each GUI client panel.

Procedure

To start the Content Manager OnDemand Administrator client:

1. Click **Start** > **All Programs** > **IBM OnDemand Clients V10.5** > **OnDemand Administrator V10.5**.
2. If you do not already have a server, follow the instructions in the online help to add a server and ensure that ODF is available on that server.
3. Double-click the name of your server to open the **Logon** dialog box.
4. Enter your user ID and password for your Content Manager OnDemand server.
5. Expand **OnDemand Distribution Facility** to see the ODF objects.
6. Create your ODF objects as needed. For example, to create a recipient, use one of the following methods:
   - Click **Recipients** in the navigator pane of the window and then select **File** > **New Recipient**.
   - Click **Recipients** in the navigator pane of the window and then right-click and select **New Recipient**.
   - Point to an existing ODF recipient, right-click and select **Copy**.
7. If you need to create a report bundle, use one of these methods instead:
   - Create a report bundle.
     a. Display a list of Distributions in the list pane.
     b. Right-click the Recipient/List name of the distribution where you want to add the report bundle and select **Add Report Bundle**. The **Add a Report Bundle** dialog is displayed. You cannot modify the Distribution Name and Recipient/List because you are adding the report bundle to the distribution identified by the Distribution Name and Recipient/List.
   - Copy an existing report bundle.
     a. Display a list of Report Bundles in the list pane.
     b. Right-click the Recipient/List name of the source report bundle and select **Copy**. The **Add a Report Bundle** dialog is displayed. The values from the source report bundle are used as the default values for the new report bundle. The sequence number is changed because it is used to uniquely identify the report bundle within the distribution.

Chapter 10. Printing and distributing Content Manager OnDemand reports

ODF process

The ODF process is a multithreaded process consisting of a main thread and at least four other threads.

The threads are:

- Scheduler
- Distribution
- Continuation
- Report bundle processing (one or more threads)

The ODF process is started by issuing the ARSODF command. The ARSODF process creates the other threads which run on a periodic basis. The periods are controlled by the parameters set in the ARS.CFG file. See "Verifying the ARS.CFG file" on page 15 for more information on the parameters.

Scheduler thread

The scheduler thread runs on a periodic basis based on the value specified in the **ARSODF_SCHDSLEEP** parameter in the ARS.CFG file.

The scheduler thread schedules all distributions that have been defined with a distribution scheduling method of **All Ready** that are not already scheduled. It schedules distributions defined with a method of **Time of Day** if the specified time has been reached and the distribution has not already been scheduled or processed for that day. It marks a distribution with a method of **Time of Print** ready to be submitted if the time of day has been reached and the distribution has not yet been submitted for report bundle processing. The scheduler thread does not schedule any distributions defined with a method of **External** or **Loaded**.

Distribution thread

The distribution thread runs on a periodic basis based on the value specified in the **ARSODF_DISTSLEEP** parameter in the ARS.CFG file.

Based on the distribution scheduling method that is defined and the report bundles that have been loaded, the distribution thread determines whether the distribution should be submitted for report bundle processing. If a distribution is ready, it submits the distribution for report bundle processing by the report bundle processor.

Continuation thread

The continuation thread runs on a periodic basis based on the value specified in the **ARSODF_CONTSLEEP** parameter in the ARS.CFG file.

The continuation thread determines if there are any reports available to submit for report bundle processing.

Report bundle processor

The report bundle processor processes each distribution that has been submitted for report bundle processing.

The report bundle processor, also referred to as the print processor, retrieves the Content Manager OnDemand report, or a subset of the report (if a query is specified) from Content Manager OnDemand. It processes the report based on the definitions in the report bundle and sends the report to the location specified.

Report bundle processing is a multi-threaded process. The number of threads that will be started is determined by the value specified in the ARS.CFG file parameter named **ARSODF_PRINTTASKS**. There might be multiple threads running at one time.

z/OS When the distribution is defined with a location of **Print**, the report bundle processor will dynamically allocate JES spool file data sets based on the allocation parameters specified in the report bundle definition. If the report bundle processor is running as a thread, the JES spool file data sets inherit the jobname from the ODF main thread. The naming convention for the JES spool file data sets is P*xxnnnnn*, where *xx* is the thread identifier and *nnnnn* is an incremental number that is generated by ODF to identify a unique JES spool file data set name. The grouping of JES spool file data set output is controlled by the **Report Break Indicator** defined in the distribution. If the indicator is set to **Yes**, a new JES spool file data set will be allocated for each report bundle in the distribution. If the indicator is set to **No**, the report bundle output will be grouped into the same JES spool file data set. If, however, the report bundle definition contains output values that differ from the distribution or each other, then a new JES spool file data set is allocated with the new output parameters.

Submitted job processing (z/OS®)

On z/OS, to improve ODF performance, you can use a submitted job and the persistence feature.

When you use a job name on distributions, ODF uses a feature of z/OS that allows jobs to run in created address spaces. The started job persistence feature allows a started job to remain active for a period of time waiting for additional distributions to be run. You can control the period of time by using the timing parameters in the ARS.CFG file.

The started job feature is designed for distributions that have JOB NAME controls that are established by report administration. Address spaces that are dynamically created for Job Name controlled distribution processes remain persistent for as long as there is distribution work to be handled. Persistent distribution report bundle processing improves performance by reusing resources.

If the distribution is defined with a jobname, then the report bundle processor is started by a submitted job. There might be multiple started jobs running at one time. The maximum number of jobs that can run at one time is specified in the ARS.CFG file parameter named **ARSODF_MAXJOBS**.

If the report bundle processor is started as a submitted job then the jobname is used for the JES spool file data set name as well as the stepname. If the jobname within a distribution changes by having different jobnames defined in the report bundle definitions, then a new JES spool file data set will be created with the new jobname. The naming convention for JES spool file data sets with a jobname is P99*nnnnn*, where *nnnnn* is an incremental number that is generated by ODF to identify a unique JES spool file data set name. If the jobname or output definitions are different within the report bundles, then a new JES spool file data set is allocated for each new jobname or output parameters. The Report Break Indicator is not used when the report bundle processor is running as a submitted job (jobname specified).

Important: If you are trying to run multiple levels of ODF on the same LPAR, the use of jobnames can result in unexplained problems. If you are not using job names for distribution there is no issue. If you do try to run multiple levels of ODF on the same LPAR and have unexplained results, stop all instances of ODF, run the ARSRASC9 job and then restart only one ODF.

Managing JES spool file data sets (z/OS®)

About this task

On z/OS, for submitted job report bundle processing only, ODF writes messages to the ODF message file. The message file is only deleted if all report bundles in a distribution are processed successfully with a return code of 0. All messages are also written to the Content Manager OnDemand System Log and are not deleted.

Terminating address space jobs (z/OS®)

On z/OS, because started jobs can remain present in the system waiting for work for up to the time limit that is specified by the **ARSODF_JOBSLEEP** parameter in the ARS.CFG file, you can use a utility to force these jobs to end.

About this task

Module ARSRASC9 is modified to accept an EXEC statement PARM string. This PARM string can be any of the following values:

PURGE
Requests the destruction of the environment and control block structure that was created by ARSRASCR. Use this function only in an emergency situation when it appears that the Address Space Create services are not succeeding and all other attempts at correcting the situation do not alleviate the problem. The PURGE function can cause ARSRASCR created address spaces that are running at the time of the Purge request is processed to ABEND, and CSA storage will be lost. However, this might allow the Address Space Create functions to run successfully after the PURGE function has completed.

RESET
Requests the resetting of the Address Space Create Suppression control. This control is set by ARSRASCR when any target address space ends without completing its application-level initialization. The control is also set when ARSRASC9 is run with the **FORCETERM** parameter.

RESET is the default if no PARM text is supplied.

FORCETERM
Requests that all active address spaces that are created by ARSRASCR be terminated. This also resets the Address Space Create Suppression control. Therefore, any subsequent requests to create address spaces are rejected. Running ARSRASC9 with the **RESET** parameter resets this suppression state.

QUIESCE
Identical to FORCETERM in all respects with one exception. A FORCETERM operation will ABEND target address spaces that have not ended within the permissible maximum time period following the receipt of the termination request. A QUIESCE operation allows target address spaces to continue processing their current unit of work without any imposed time limit.

ARSRASC9, if invoked with the **FORCETERM** parameter, operates in the following way:

Procedure

1. A WTOR is produced requesting confirmation of the FORCETERM function. If the confirmation is denied, no further action is taken.
2. The "Mass Terminate" function of ARSRASCR is invoked, and a status message reflecting the processing performed by ARSRASCR is produced.
3. Exit.

Results

The Mass Terminate function of ARSRASCR operates in the following order:

1. The Address Space Create Suppression control is set to inhibit the creation of new address spaces.

2. The New Work/Shutdown ECB for each target address space is posted with a POST code indicating this is a shutdown request. The code that is running in each of these address spaces should respond to this POST action as quickly as possible and proceed with orderly termination.
3. ARSRASCR allows each target address space 60 seconds to terminate voluntarily. After this time period expires, any remaining address spaces are forcibly terminated through the z/OS CALLRTM interface. This style of termination is similar to the results of the z/OS CANCEL operator command; the address space is ABENDed with an indication that Recovery Routines are allowed to run but they might not schedule Retry Routines.
4. ARSRASCR returns to the caller.

Forcible and quiesced mass termination of ARSRASCR created address spaces (z/OS)

On z/OS, you can control the termination of the submitted job address space.

The ARSRASC9 program is used to control the termination of the submitted job address space. When the ARSRASC9 program is invoked to process a FORCETERM or QUIESCE request, the following messages are produced:

```
ARS04502I ODF STARTED ADDRESS SPACE TERMINATION REQUESTED,
          MODE=termination_mode

ARS04503A REPLY 'Y' TO CONFIRM mode ADDRESS SPACE
          TERMINATION.
```

where

termination_mode
FORCETERM or QUIESCE, corresponding to the type of termination processing that is requested.

mode
FORCETERM or QUIESCE, corresponding to the type of termination processing that is requested.

If the response to message ARS04503A is anything other than the single character Y, the mass termination request is not performed and the following message is produced:

```
ARS04504I mode ADDRESS SPACE TERMINATION ABANDONED
```

Otherwise, this message is produced:

```
ARS04504I mode ADDRESS SPACE TERMINATION PROCEEDING
```

Now, all target address spaces (that is, address spaces that were created by ARSRASCR) are signaled to voluntarily (and promptly) terminate.

For a QUIESCE operation, all target address spaces are allowed to continue processing their current unit of work. After this work is completed, the address space should terminate. Any request for a new unit of work that originates from any target address space is rejected.

For a FORCETERM operation, all target address spaces have one minute to end. If all of the target address spaces have ended before one minute has elapsed, then no other action is taken.

Otherwise, the remaining target address spaces are forcibly ended using ABEND (each address space will be ABENDed with a U0013 ABEND, Reason Code x'0B0F'). The following message is also produced:

```
ARS04505I AT LEAST ONE ADDRESS SPACE HAS BEEN ABENDED
```

The following are a few sample scenarios of forcible and quiesced mass termination of ARSRASCR created address spaces:

Initiating a forcible mass termination request and then rejecting it

```
ARS04502I ODF STARTED ADDRESS SPACE TERMINATION REQUESTED,
          MODE=FORCETERM
*113 ARS04503A REPLY 'Y' TO CONFIRM FORCIBLE ADDRESS SPACE
               TERMINATION.
R 113,X
ARS04504I FORCIBLE ADDRESS SPACE TERMINATION ABANDONED
ARS04507I ARSRASC9 PROCESSING ENDED, RC=00000000,
          FCN=FORCETERM
```

Initiating a forcible mass termination request and allowing it to proceed. All address spaces terminate within one minute

```
ARS04502I ODF STARTED ADDRESS SPACE TERMINATION REQUESTED,
          MODE=FORCETERM
*109 ARS04503A REPLY 'Y' TO CONFIRM FORCIBLE ADDRESS SPACE
                TERMINATION.
R 109,Y
ARS04504I FORCIBLE ADDRESS SPACE TERMINATION PROCEEDING
ARS04507I ARSRASC9 PROCESSING ENDED, RC=00000000,
          FCN=FORCETERM
```

Initiating a forcible mass termination request and allowing it to proceed. Not all address spaces terminate within one minute

```
ARS04502I ODF STARTED ADDRESS SPACE TERMINATION REQUESTED,
          MODE=FORCETERM
*112 ARS04503A REPLY 'Y' TO CONFIRM FORCIBLE ADDRESS SPACE
                TERMINATION.
R 112,Y
ARS04504I FORCIBLE ADDRESS SPACE TERMINATION PROCEEDING
ARS04505I AT LEAST ONE ADDRESS SPACE HAS BEEN ABENDED
ARS04507I ARSRASC9 PROCESSING ENDED, RC=00000000,
          FCN=FORCETERM
```

You also see messages similar to the following for each address space that is ABENDed:

```
IEA995I SYMPTOM DUMP OUTPUT
   USER COMPLETION CODE=0013 REASON CODE=00000B0F
```

Initiating a Mass Quiesce request and allowing it to proceed

```
ARS04502I ODF STARTED ADDRESS SPACE TERMINATION REQUESTED,
          MODE=QUIESCE
*110 ARS04503A REPLY 'Y' TO CONFIRM QUIESCED ADDRESS SPACE
                TERMINATION.
R 110,Y
ARS04504I QUIESCED ADDRESS SPACE TERMINATION PROCEEDING
ARS04507I ARSRASC9 PROCESSING ENDED, RC=00000000,
          FCN=QUIESCE
```

When ARSRASC9 is invoked with the PURGE or RESET PARM text, it does not ask for confirmation. It simply goes ahead with the request. The only message that you should expect to see is ARS04507I as in the following example:

```
ARS04507I ARSRASC9 PROCESSING ENDED, RC=00000000, FCN=RESET
```

Required one-time preparatory actions (z/OS®)

On z/OS, before you use the Address Space Create component services, you must follow these steps.

About this task

Before you attempt to use the Address Space Create component services, perform the following actions:

- Terminate all running instances of ARSODF.
- Terminate all running instances of address spaces that were started by ARSODF.
- Run module ARSRASC9.

```
      //PURGE  EXEC  PGM=ARSRASC9,PARM='PURGE'
      //STEPLIB  DD  DISP=SHR,DSN=ARS.V10R5M0.SARSLOAD
```

The following Write-to-Operator message is produced upon the successful running of ARSRASC9:

```
ARS04507I ARSRASC9 PROCESSING ENDED, RC=00000000,FCN=PURGE
```

Running ARSRASC9 frees shared persistent in-memory structures. The structures are automatically reconstructed when the Address Space Create Component services are subsequently invoked by ARSODF.

Perform these actions only one time on a given z/OS system.

Failure to perform these actions results in subsequent failures that are reported by ARSODF and other ODF modules when they request the services of the Address Space Create component. The most common messages produced by ARSODF in this situation are similar to the following:

```
ARS04275I ARSRASCR NON-ZERO RET REASON CODE REASON CODE = 13
ARS04275I ARSRASCR NON-ZERO RET RETURN CODE RETURN CODE = 2
ARS04299S *** *** *** PROGRAM LOGIC ERROR - ABENDING ** ***
```

Required additions to the ODF started address space JCL procedure (z/OS)

These JCL changes are required to avoid ABEND conditions.

About this task

On z/OS, the Language Environment® runtime services that are run within an ODF Started Address Space reference the following JCL DDNAMEs:

CEEDUMP
CEESNAP
SYSERR
SYSOUT
SYSPRINT
SYSTERM

Each of these DDNAMEs must be statically allocated as `DD DUMMY` in the JCL procedure that is associated with dynamically started ODF address spaces.

If any of these named DD statements is not statically allocated, then Language Environment attempts to dynamically allocate the DD as a SYSOUT. This, in turn, can cause multiple ABEND conditions to occur in both the ODF started address space in which the dynamic allocation was performed and in the JES address space. Typical ABENDs are S013-0C0, S001, S614, and S0C4. These failures might also be accompanied by HASP708 error messages.

Externally submitting distributions

Externally submitting distributions for report bundle processing performs the same, but limited, processing as the ODF main thread processor in batch mode.

This process creates report bundle processing requests only for documents that exist in the archive, and processes only one specific distribution at a time. This function is designed to work with user created scheduled report records. The user creates the scheduled report records with all the correct fields and sets the scheduled report status field to a value of x'40' or decimal 64.

To externally submit distributions for report bundle processing, run the ARSODF program with the **-d**, **-r**, and **-e** parameters. See Chapter 7, “Running ODF,” on page 31 for more information about externally submitting distributions and Chapter 12, “ARSODF,” on page 57 for more information about the ARSODF program.

Creating the external information

The user is responsible for creating the scheduled report records for the reports to be processed by the ODF report bundle processor. The scheduled reports must be properly constructed with a valid report ID,

application group, application, and load ID for a defined distribution with the status field set to a value of x'40' or decimal 64.

About this task

When externally submitting distributions for report bundle processing, ODF validates the distribution, locates the scheduled reports, and finds the matching report bundles that are defined for the distribution. After all scheduled reports are processed for every report bundle, a distribution request is created to indicate to ODF that the distribution is ready for processing.

Chapter 11. Obtaining status and monitoring distributions and reports

The Content Manager OnDemand Monitor is an interactive workstation client program that allows you to check the status of distributions submitted for processing and to monitor ODF activity on Content Manager OnDemand servers beginning at version 9.5.

About this task

Previous versions of ODF on Content Manager OnDemand for z/OS use an ISPF monitor that is replaced by the Content Manager OnDemand Monitor. The ISPF monitor is not supported beginning at version 9.5.

Installing the monitor

The Content Manager OnDemand Monitor is installed when you install the Content Manager OnDemand Administrator and end-user clients.

About this task

During the installation of the Content Manager OnDemand clients, select **Custom** instead of **Typical** for the installation type. Then click to select Monitor along with any other clients you require. Click **Next** to begin the installation.

Using the monitor

Follow these steps to launch and use the Content Manager OnDemand Monitor.

To launch the Content Manager OnDemand Monitor, follow these steps:

1. Select **Start** > **All Programs** > **IBM OnDemand Clients V10.5** > **OnDemand Monitor V10.5**.
2. Select your Content Manager OnDemand server from the drop-down list of servers. The Content Manager OnDemand Monitor does not support servers at versions earlier than version 9.5.
3. Enter your Content Manager OnDemand user ID and password to that server. The user ID must be defined as a Content Manager OnDemand System Administrator or a user with ODF Administration authority.
4. Click **OK** to log on.

The monitor has three sections that contain information related to ODF.

- The **Statistics** section of the panel displays the current number of distributions, scheduled reports, and report bundles by status. Click the down arrow to expand the System statistics section if it is not already expanded. To change the date range for the statistics, modify the dates in the **Filter** section and then click **Refresh**.
- The **Filter** section of the panel allows you to specify selection criteria used to subset the results listed under each tab in the lower section of the panel. Click the down arrow to expand the Filter section if it is not already expanded. The Filter section reflects the selection values used to populate the rows in the tab sections.
- Each of the tabs in the last section displays a list of the particular ODF objects that match the search criteria in the Filter section. You can select from the following tabs:
  - Defined Distributions - All defined distributions and most recent request information
  - Requested Distributions - All distribution requests
  - Defined Report Bundles - All defined report bundles and most recent request information

- Scheduled Reports - All scheduled or processed reports
- Processed Report Bundles - All scheduled or processed report bundles

See the Content Manager OnDemand Monitor online help for more details about the fields in each section.

Chapter 12. ARSODF

Purpose

The ARSODF program is the primary ODF processing program.

The ARSODF program is used to start and terminate ODF processing and to schedule individual distributions for processing, based on the parameters you specify.

Run the ARSODF program only on the library server.

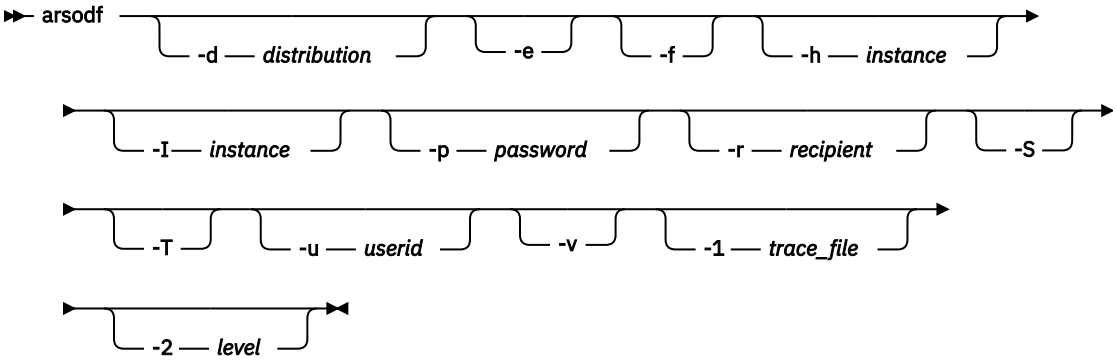
IBM i On IBM i, the ARSODF program runs in the QSHELL environment. The ARSODF program requires the QIBM_MULTI_THREADED environment variable to be set to a value of 'Y' which can be done by using the following command:
`ADDENVVAR ENVVAR(QIBM_MULTI_THREADED) VALUE('Y') LEVEL(*xxx)` where *xxx is either *SYS or *JOB. For more information about environment variables and using the QSHELL environment, see the *Content Manager OnDemand for i Common Server Administration Guide*.

The ARSODF program replaces the programs that are used for ODF processing in previous versions:

- Windows Linux AIX On Multiplatforms servers, the ARSODF program replaces the ARSRD program.
- z/OS On z/OS, the ARSODF program replaces the ARSRMAIN, ARSBDIST, and ARSRDFGO programs, depending on the parameters you specify.

For more information, see Chapter 7, “Running ODF,” on page 31 before you use the ARSODF program.

Syntax



Description

The ARSODF program provides the following ODF functions:

- Start ODF processing
- Terminate ODF processing
- Initiate individual ODF distributions

Parameters

-d distribution
Specifies the name of an individual ODF distribution to be scheduled. This parameter is only used when using the external scheduling process. When you use the **-d** parameter, the **-r** parameter must also be specified. Do not use the **-S** or **-T** parameter when you use this parameter.

-e
This parameter will submit the distribution specified by the **-d** and **-r** parameters for report bundle processing. When you use the **-e** parameter, the **-d** and **-r** parameters must also be specified.

-f
Process report bundle if needed. This option will ensure that the selection criteria specified in the SQL statement of the report bundle definition is found in the available report before creating the report bundle processing request. This option will eliminate processed report bundles ending in error message ARS04414E and distributions ending in error because the SQL specified was not found in the report. If none of the available reports match the SQL statement, then the distribution will be marked complete and no processed reports will be created.

-h instance
The name of the Content Manager OnDemand instance to process. The ARSODF program will attempt to locate the specified instance name in the ARS.INI file (AIX, Linux, z/OS, or IBM i 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 ARSODF 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:

- z/OS Windows Linux AIX The name of your Content Manager OnDemand instance is not ARCHIVE.
- IBM i The name of your Content Manager OnDemand instance is not QUSROND or the instance name specified as your default in the QDFTINST data area.
- You are running more than one instance on the same system and you need to specify an instance other than the default instance.

The ARSODF 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.

-I instance
The name of the Content Manager OnDemand instance to process. The **-I** parameter provides the same function as the **-h** parameter. If you specify both, the value of the last parameter specified is used.

-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 ""**). If you do not specify the **-p** parameter while you specify the **-u** parameter, then the program retrieves the password for the user ID from the stash file specified in ARS.INI file for that instance.

-r recipient
Specifies the name of an individual ODF recipient to receive the distribution to be scheduled. If you are not using the **-e** parameter, the **-r** parameter can also specify the name of a recipient list. This parameter is only used when using the external scheduling process. When you use the **-r** parameter, the **-d** parameter must also be specified. Do not use the **-S** or **-T** parameter when you use this parameter.

-S
Start ODF processing. See Chapter 7, “Running ODF,” on page 31 for more information.

-T
Terminate ODF processing. See Chapter 7, “Running ODF,” on page 31 for more information.

-u userid
The Content Manager OnDemand user that is permitted to perform the specified function. The ARSODF program requires the following permissions:

- The user ID that you specify is a valid Content Manager OnDemand user in the instance that you name with the **-h** or **-I** parameter.
- The user ID is a Content Manager OnDemand System Administrator or has ODF Administration authority.
- The user ID is authorized to the application group and application to be distributed.

If you do not specify the **-p** parameter with the **-u** parameter, the ARSODF program retrieves the password for the user ID from the stash file specified in the ARS.INI file for that instance.

-v
Enables verbose mode, which displays all messages (informational and error). By default, the ARSODF program displays error messages.

-1 trace_file
Specify a fully qualified trace file name as directed by IBM Software Support.

-2 level
Specify a numeric value as directed by IBM Software Support.

Examples

1. z/OS The following example shows how to start ODF processing, using unified logon to provide the Content Manager OnDemand user and password:

```
arsodf -I instancename -S
```

2. The following example shows how to terminate ODF processing, obtaining the Content Manager OnDemand password for the user ID from a stash file:

```
arsodf -I instancename -T -u myuser
```

3. The following example shows how to manually schedule a distribution. This method replaces the ARSRDFGO function in previous versions of ODF on z/OS:

```
arsodf -I instancename -d 'Daily sales invoices' -r ASMITH
```

4. The following example shows how to externally submit a distribution for report bundle processing. This method replaces the ARSBDIST function in previous versions of ODF on z/OS:

```
arsodf -I instancename -e -d 'Daily sales invoices' -r ASMITH
```

Files

AIX **/opt/IBM/ondemand/V10.5/bin/arsodf**
The AIX executable program.

Linux **/opt/ibm/ondemand/V10.5/bin/arsodf**
The Linux executable program.

Windows **\Program Files\IBM\OnDemand\V10.5\bin\arsodf.exe**
The Windows executable program.

z/OS **/usr/lpp/ars/V10R5M0/bin/arsodf**
The z/OS executable program.

IBM i **/usr/bin/arsodf**
The IBM i executable program.

Chapter 13. ODF user exits

ODF provides several user exits that you can customize.

The user exits include:

ODFProcessDist.java
Processed distribution exit

z/OS **arsodfxa**
Spool file data set allocation attributes exit

z/OS **arsodfxb**
Banner/header/trailer exit

z/OS **arsodfxm**
Bundle manifest exit

z/OS ODF versions earlier than V9.5 supported the following user exits:

- ARSRPREA - replaced by arsodfxa
- ARSRBAN - replaced by arsodfxb
- ARSRMFST - replaced by arsodfxm
- ARSRATTE - replaced by ODFProcessDist.java
- ARSRNOTE - replaced by ODFProcessDist.java
- ARSRSECR - no longer supported

z/OS If you have customized one of the user exits from a version of ODF earlier than version 9.5 that is replaced by a new version 9.5 or later exit, you must rename your user exit program to the new user exit name, update the parameter list to match the version 10.5 copybook, update your user exit program to use the new parameter names, update your output to add a new line delimiter character, and then compile and link-edit the updated program. If you currently use an unmodified version of one of the user exists that is being replaced by a new version 9.5 or later exit, no action is required. If you currently use one of the exits that is no longer supported, you must find an alternative to accomplish the function that your current user exit program performs.

See the PROLOG and block comments in each sample user exit for detailed information about what the user exit does and any requirements it has.

ODFProcessDist.java - Processed distribution exit

Customize your ODF environment in several ways by using the `ODFProcessDist.java` user exit program.

Use the `ODFProcessDist.java` program to modify your ODF environment in the following ways:

- Customize the details and format of the outgoing emails that contain distributions when the distribution **Location** value is set to **E-mail**. Customize the format for any **Location** value with the **Notify by e-mail** check box selected. For each distribution location type, you can customize the email content and the maximum size for email attachments within a single email.
- Customize the details of distribution output for all other distribution types on Content Manager OnDemand for Multiplatforms and IBM i servers, and for all distribution types on z/OS except when **Location** value is set to **Print**.
- Specify your SMTP server name to use for outgoing email.
- Specify whether to enable the secure sockets layer (SSL) when you use the SMTP server to send email.
- Specify trace parameters.

- IBM i | Windows | Linux | AIX On Content Manager OnDemand for Multiplatforms and IBM i servers, specify the name of the command to use to submit ODF print requests and the name of the printer queue to use.

The ODFProcessDist.java program uses the ARSODF.XML file as input. A compiled version of the sample ODFProcessDist.java program is shipped with ODF. You can use the program as it exists as shown in the sample program, or you can modify the sample program and recompile it to further customize outgoing distribution details. For more information about the ARSODF.XML file, see "Verifying the ARSODF.XML file" on page 19.

Linux | AIX The sample ODFProcessDist.java program is delivered as a default executable file in the *installpath*/jars directory, where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand. A source code sample is provided in the *installpath*/exits/odf directory.

Windows The sample ODFProcessDist.java program is delivered as a default executable file in the *installpath*/jars directory, where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand. A source code sample is provided in the *installpath*/exits/odf directory.

z/OS The sample ODFProcessDist.java program is delivered as a default executable file in the *installpath*/jars directory, where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand. A source code sample is provided in the *installpath*/samples directory.

IBM i The sample ODFProcessDist.java program is delivered as a default executable file in the /QIBM/ProdData/OnDemand/jars IFS directory. A source code sample is provided in the /QIBM/ProdData/OnDemand/jars IFS directory.

If you need to change the ODFProcessDist.java program, copy the source code sample to a different directory, such as /ODF/modified/samples. With this copy step, you can preserve the original copy if you need to use it later. Then, modify the ODFProcessDist.java program that you copied. Compile the program and place the updated ODFProcessDist.class file in the IBM-supplied jars directory, retaining the original ODFProcessDist name.

If you encounter errors when processing distributions, check for messages in the ODFProcessDist0.log file. The log file is created for you if it does not already exist. The file is located in the temporary directory specified by the ARSODF_TMP entry in your ARS.CFG configuration file. If the ARSODF_TMP entry does not exist, the log file is located in the directory specified by the ARS_TMP entry in the ARS.CFG file. If ARSODF_TMP and ARS_TMP entries do not exist, the default directory is /tmp. To control the level of messages that are written to the log file, modify the <tracelevel> element in the ARSODF.XML file. See Configuring ARSODF.XML elements for more information on the <tracelevel> element.

arsodfxa - Spool file data set allocation attributes exit

On z/OS, you can use the arsodfxa spool file data set preallocation exit to modify the currently-defined ODF JES spool file data set output parameter definitions that are used for dynamic allocation of the report and manifest JES spool file data sets. The arsodfxa exit is called when ODF detects a non-blank **Customer Variables** field in either the ODF distribution or report bundle definitions, but only if the field value is not set to **DO NOT SCHED** or **NOSCHED**.

The arsodfxa exit is passed the output parameters that are specified in the report bundle definition and the output parameter string. The exit can modify the output parameter string. The string that is returned from the user exit will be used to allocate the JES spool file data sets for the report bundle and manifest JES spool file data sets.

Figure 2 on page 63 shows an example of an output parameter string.

```
CLA(X)
DES(XXXXXXXXXXXXXXXXXXX)
FORMS(XXXXXXXX)
WRI(XXXXXXXX)
FORMD(XXXXXX)
COP(XXX)
PRM(XXXXXXXX)
TRC/NOTRC FCB(XXXX)
UCS(XXXX)
BURST/NOBURST MOD(XXXX)
FLA(XXXX)
PAG(XXXXXX)
CHA(WWWW,XXXX,YYYY,ZZZZ)
```

Figure 2. Example of an output parameter string

The return output string must adhere to the syntax shown or an allocation failure will occur.

arsodfxa parameters

The following table shows the parameters for the arsodfxa user exit program.

Table 8. arsodfxa parameters

arsodfxa parameter	Size*	Explanation
01 ODFATTRS-Parms.		
05 ODFATTRS-RECIPIENT	Pointer.	Recipient name
05 ODFATTRS-DIST-NAME	Pointer.	Distribution name
05 ODFATTRS-SEQ-NUM	COMP PIC S9(8).	Report bundle sequence number
05 ODFATTRS-REPORTID	Pointer.	Report ID
05 ODFATTRS-APPLGRP-NAME	Pointer.	Application group name
05 ODFATTRS-APPL	Pointer.	Application name
05 ODFATTRS-CREATE-DATE	Pointer.	Creation date
05 ODFATTRS-PROCESS-DATE	Pointer.	Processed date
05 ODFATTRS-JOBNAME	Pointer.	Jobname that is specified in the distribution definition
05 ODFATTRS-FORMDEF	Pointer.	Formdef that is specified in the distribution definition
05 ODFATTRS-PAGEDEF	Pointer.	Pagedef that is specified in the distribution definition
05 ODFATTRS-FORMSID	Pointer.	Forms ID that is specified in the distribution definition
05 ODFATTRS-DEST	Pointer.	Destination that is specified in the distribution definition
05 ODFATTRS-CLASS	Pointer.	Class that is specified in the distribution definition
05 ODFATTRS-WRITER	Pointer.	Writer that is specified in the distribution definition
05 ODFATTRS-FCB	Pointer.	FCB that is specified in the distribution definition
05 ODFATTRS-UCS	Pointer.	UCS that is specified in the distribution definition

Table 8. arsodfxa parameters (continued)		
arsodfxa parameter	**Size***	**Explanation**
`05 ODFATTRS-BURST`	Pointer.	BURST that is specified in the distribution definition
`05 ODFATTRS-COPYM`	Pointer.	COPYM that is specified in the distribution definition
`05 ODFATTRS-FLASH`	Pointer.	FLASH that is specified in the distribution definition
`05 ODFATTRS-CUSTVR`	Pointer.	Customer variables that are specified in the distribution definition
`05 ODFATTRS-ACCOUNT`	Pointer.	Account that is specified in the distribution definition
`05 ODFATTRS-EXITINFO`	Pointer.	Exit information that is specified in the distribution definition
`05 ODFATTRS-OUTLIM`	COMP PIC S9(8).	Output limit that is specified in the distribution definition
`05 ODFATTRS-NCOPY`	COMP PIC S9(8).	NCOPY that is specified in the distribution definition
`05 ODFATTRS-PRMODE`	Pointer.	PRMODE that is specified in the distribution definition
`05 ODFATTRS-CHAR1`	Pointer.	CHAR1 that is specified in the distribution definition
`05 ODFATTRS-CHAR2`	Pointer.	CHAR2 that is specified in the distribution definition
`05 ODFATTRS-CHAR3`	Pointer.	CHAR3 that is specified in the distribution definition
`05 ODFATTRS-CHAR4`	Pointer.	CHAR4 that is specified in the distribution definition
`05 ODFATTRS-TRC`	PIC X.	TRC that is specified in the distribution definition
`05 ODFATTRS-REPRINT`	PIC X.	Reprint flag
`05 ODFATTRS-TRACE`	PIC X.	Trace flag
`05 ODFATTRS-OUTDEST`	Pointer.	Output destination
* All parameters with Size defined as Pointer contain the address of a null terminated string that contains the actual parameter value.		

Customizing arsodfxa

A sample COBOL program for arsodfxa is delivered as a default executable in the *installpath*/bin/exits directory, where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand.

About this task

Source code samples and the JCL statements that are used to compile the exits are provided in the ARS.V10R5M0.SARSINST library. Parameters are declared in COBOL copybook named arsodfxc. The sample compile JCL is named ARSCODFX.

If ODF distribution JES spool file data set output parameters need to be modified before dynamic allocation, the arsodfxa sample exit program must be compiled and link-edited into the *installpath*/bin/exits directory where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand.

Procedure

To customize the arsodfxa exit, do the following steps:

1. Create a backup copy of the user exit.
2. Make the desired modifications to the exit.
3. Modify and run the sample JCL ARSCODFX in ARS.V10R5M0.SARSINST to compile and link the user exit into the *installpath*/bin/exits directory where *installpath* is the path you chose when you installed Content Manager OnDemand.

arsodfxb - Banner/header/trailer exit

On z/OS, the arsodfxb exit enables you to customize the banner information that is written out to the JES spool file data sets. Banner information is written to the JES spool file data set when the recipient definition requests a banner to be printed and the location of the report bundle is print.

ODF calls the arsodfxb exit for three different types of banner data:

Banner Page
Information to be written out before the first report bundle in the distribution is written out to the JES spool file data set. The exit is called at the start of processing the first report bundle within the distribution with `ODFBANER-REQUEST-TYPE = '1'` to process Banner information.

Header Page
Information to be written out before the second and each subsequent report bundle in the information. The exit is called before each subsequent report bundle within the distribution with `ODFBANER-REQUEST-TYPE = '2'` to process the Header information.

Trailer Page
Information to be written out to the JES spool file data set after the report bundle has been written out. The exit is called after each report bundle is processed with `ODFBANER-REQUEST-TYPE = '3'`. The exit is passed information about the report bundle and recipient and uses this information to format the lines to display.

The exit returns a buffer of data. The maximum size is 10240 bytes. The exit formats the data and adds a new line character x'15' wherever the data should start on a new line in the spool file.

arsodfxb parameters

The following table shows the parameters for the arsodfxb user exit program.

Table 9. arsodfxb parameters

arsodfxb parameter	Size*	Explanation
`01 ODFBANER-Parms`		

Table 9. arsodfxb parameters (continued)		
arsodfxb parameter	**Size***	**Explanation**
05 ODFBANER-LIST-NAME	Pointer.	Recipient list name
05 ODFBANER-RECIPIENT	Pointer.	Recipient who receives the distribution
05 ODFBANER-DISTRIBUTION-NAME	Pointer.	Distribution name
05 ODFBANER-SEQUENCE-NUMBER	COMP PIC S9(8).	Report bundle sequence number
05 ODFBANER-REPORTID	Pointer.	Report ID
05 ODFBANER-APPLGROUP	Pointer.	Application group name
05 ODFBANER-APPLNAME	Pointer.	Application name
05 ODFBANER-PROCESS-DT	Pointer.	Process date. Controlled by TZ parameter in ars.cfg.
05 ODFBANER-HDR1	Pointer.	HDR1 value that is specified in the recipient definition
05 ODFBANER-HDR2	Pointer.	HDR2 value that is specified in the recipient definition
05 ODFBANER-HDR3	Pointer.	HDR3 value that is specified in the recipient definition
05 ODFBANER-HDR4	Pointer.	HDR4 value that is specified in the recipient definition
05 ODFBANER-HDR5	Pointer.	HDR5 value that is specified in the recipient definition
05 ODFBANER-HDR6	Pointer.	HDR6 value that is specified in the recipient definition
05 ODFBANER-HDR7	Pointer.	HDR7 value that is specified in the recipient definition
05 ODFBANER-HDR8	Pointer.	HDR8 value that is specified in the recipient definition
05 ODFBANER-JNAME	Pointer.	Jobname that is specified in the report bundle definition
05 ODFBANER-PPT-FORMDEF	Pointer.	Formdef that is specified in the report bundle
05 ODFBANER-PPT-PAGEDEF	Pointer.	Pagedef that is specified in the report bundle
05 ODFBANER-CUSTOMER-VARIABLES	Pointer.	Customer variables that are specified in the report bundle
05 ODFBANER-EXITINFO	Pointer.	Exit information that is specified in the report bundle
05 ODFBANER-ACCOUNT	Pointer.	Account value that is specified in the recipient definition
05 ODFBANER-ADDRESS1	Pointer.	Address1 that is specified in the recipient definition

Table 9. arsodfxb parameters (continued)		
arsodfxb parameter	**Size***	**Explanation**
05 ODFBANER-ADDRESS2	Pointer.	Address2 that is specified in the recipient definition
05 ODFBANER-ADDRESS3	Pointer.	Address3 that is specified in the recipient definition
05 ODFBANER-ADDRESS4	Pointer.	Address4 that is specified in the recipient definition
05 ODFBANER-BUILDING	Pointer.	Building that is specified in the recipient definition
05 ODFBANER-DEPT	Pointer.	Department that is specified in the recipient definition
05 ODFBANER-NAME	Pointer.	Name that is specified in the recipient definition
05 ODFBANER-ROOM	Pointer.	Room that is specified in the recipient definition
05 ODFBANER-TITLE	Pointer.	Recipient title that is specified in the recipient definition
05 ODFBANER-EMAIL	Pointer.	Email address that is specified in the recipient definition
05 ODFBANER-DESC	Pointer.	Recipient description that is specified in the recipient definition
05 ODFBANER-SQL	Pointer.	SQL query
05 ODFBANER-EMAILNOTIFY	PIC X.	Notify by email flag
05 ODFBANER-LOC	PIC X.	Location that is specified in the report bundle definition - x'01' = email, x'02' = file, x'03' = print, x'04' = distribution loaded
05 ODFBANER-DOC-TYPE	PIC X.	Document type - x'41' = afp, x'42' = bmp, x'45' = email, x'47' = gif, x'4A' = jfif, x'4B' = djde, x'4C' = line, x'4D' = meta, x'4E' = none, x'4F' = oddoc, x'50' = pcx, x'51' = png, x'52' = pdf, x'53' = scs, x'54' = tiff, x'55' = usrdef, x'58' = scs-ext, x'5A' = xml
05 ODFBANER-DOC-FORMAT	PIC X.	Document format - x'00' = fixed, x'01' = variable, x'02' = stream
05 ODFBANER-RECLEN	COMP PIC S9(8).	Record length
05 ODFBANER-RECDELIM	Pointer.	Record delimiter
05 ODFBANER-CODEPAGE	COMP PIC S9(8).	Code page
05 ODFBANER-CCTYPE	PIC X.	Carriage control - 'A' = ANSI, 'M' = machine, 'N' = none
05 ODFBANER-TRC	PIC X.	TRC
05 Reserved	PIC X(2).	Reserved

Table 9. arsodfxb parameters (continued)		
arsodfxb parameter	**Size***	**Explanation**
`05 ODFBANER-PRMODE`	Pointer.	PRMODE
`05 ODFBANER-CHAR1`	Pointer.	CHAR1 that is specified in the application definition
`05 ODFBANER-CHAR2`	Pointer.	CHAR2 that is specified in the application definition
`05 ODFBANER-CHAR3`	Pointer.	CHAR3 that is specified in the application definition
`05 ODFBANER-CHAR4`	Pointer.	CHAR4 that is specified in the application definition
`05 ODFBANER-REQUEST-TYPE`	PIC X.	Request type - x'01' = first header, x'02' = header, x'03' = trailer
`05 ODFBANER-PRINT-FLAG`	PIC X.	Print flag
`05 Reserved`	PIC X(2).	Reserved
`05 ODFBANER-BANNER-LENGTH`	COMP PIC S9(8).	Number of bytes that the exit is returning
`05 ODFBANER-BANNER`	PIC X(10240).	Banner data that is returned by the exit
* All parameters with Size defined as Pointer contain the address of a null terminated string that contains the actual parameter value.		

Customizing arsodfxb

A sample COBOL program for arsodfxb is delivered as a default executable in the *installpath*`/bin/exits` directory, where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand.

About this task

Source code samples and the JCL statements that are used to compile the exits are provided in the ARS.V10R5M0.SARSINST library. Parameters are declared in COBOL copybook named arsodfxc. The sample compile JCL is named ARSCODFX.

If changes are made to the program, the arsodfxb exit program must be compiled and link-edited into the *installpath*`/bin/exits` directory where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand.

Procedure

To customize the arsodfxb exit, do the following steps:

1. Create a backup copy of the user exit.
2. Make the desired modifications to the exit.
3. Modify and run sample JCL ARSCODFX in ARS.V10R5M0.SARSINST to compile and link the user exit into the *installpath*`/bin/exits` directory where *installpath* is the path you chose when you installed Content Manager OnDemand.

arsodfxm - Bundle manifest exit

On z/OS, the sample arsodfxm user exit is a COBOL program that enables you to customize the manifest output.

The manifest consists of a Header and Detail Lines. ODF calls the bundle manifest exit with two different functions: 1 to process the header section of the manifest and 2 to process the detail lines.

When the request type is Header, the exit returns a buffer of data. The maximum size is 1024 bytes. The exit formats the data and adds a new line character of x'15' wherever the data should start on a new line in the spool file.

arsodfxm parameters

The following table shows the parameters for the arsodfxm user exit program.

Table 10. arsodfxm parameters

arsodfxm parameter	Size*	Explanation
`01 ODFMNFST-Parms`		
* Input to exit for every function *		
`05 ODFMNFST-CURRENT-TIMESTAMP`	Pointer.	Timestamp when the user exit program was called
`05 ODFMNFST-RECIPIENT`	Pointer.	Recipient name
`05 ODFMNFST-DISTNAME`	Pointer.	Distribution name
`05 ODFMNFST-JOBNAME`	Pointer.	Jobname
`05 ODFMNFST-REQUEST-TYPE`	PIC X.	Request type - x'01' = manifest header, x'02' = manifest detail
`05 ODFMNFST-PRINT-FLAG`	PIC X.	Set by the user exit program - 1 = include manifest information in output, 0 = do not include manifest information in output
`05 Reserved`	PIC X(2).	Reserved
* Output from exit for function = H *		
`05 ODFMNFST-HEADER-LEN`	COMP PIC S9(8).	Total length of the manifest header; set by the user exit program
`05 ODFMNFST-HEADEROUT`	Pointer.	Header information, with a maximum length of 1025
`05 Reserved`	PIC X(3).	Reserved
* Input to exit for function = D *		
`05 ODFMNFST-REPORTID`	Pointer.	Report ID of the report bundle
`05 ODFMNFST-SEQNUM`	COMP PIC S9(8).	Sequence number of the report bundle
`05 ODFMNFST-APPLGRP`	Pointer.	Application group name
`05 ODFMNFST-APPL`	Pointer.	Application name
`05 ODFMNFST-SCHEDTIME`	Pointer.	Timestamp when the distribution was scheduled for processing

Table 10. arsodfxm parameters (continued)		
arsodfxm parameter	**Size***	**Explanation**
`05 ODFMNFST-PROCESS-TIME`	Pointer.	Timestamp when the distribution was processed
`05 ODFMNFST-REPRINT`	PIC X.	Reprint flag - Y = This is a reprint of a report bundle
`05 ODFMNFST-DOC-TYPE`	PIC X.	Document type - x'41' = afp, x'42' = bmp, x'45' = email, x'47' = gif, x'4A' = jfif, x'4B' = djde, x'4C' = line, x'4D' = meta, x'4E' = none, x'4F' = oddoc, x'50' = pcx, x'51' = png, x'52' = pdf, x'53' = scs, x'54' = tiff, x'55' = usrdef, x'58' = scs-ext, x'5A' = xml
`05 Reserved`	PIC X(2).	Reserved
`05 ODFMNFST-PAGE-COUNT`	COMP PIC S9(8).	Number of pages in the report bundle
`05 ODFMNFST-DOC-NOT-FOUND`	PIC X.	Document not found flag - 1 = Document was not found, 0 = Document was found
`05 ODFMNFST-SEVERE-ERROR`	PIC X.	Severe error flag - 1 = An error occurred while processing the report bundle, 0 - No error occurred
* Output from exit for function = D *		
`05 ODFMNFST-DETAIL`		
`10 DETAIL-LINE`	PIC X(121).	Detailed report bundle information to include in the manifest; set by the user exit program
* All parameters with Size defined as Pointer contain the address of a null terminated string that contains the actual parameter value.		

Customizing arsodfxm

A sample COBOL program for arsocfxm is delivered as a default executable in the *installpath*`/bin/exits` directory, where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand.

About this task

Source code samples and the JCL statements that are used to compile the exits are provided in the ARS.V10R5M0.SARSINST library. Parameters are declared in COBOL copybook named arsodfxc. The sample compile JCL is named ARSCODFX.

If changes are made to the program, arsodfxm exit program must be compiled and link-edited into the *installpath*`/bin/exits` directory where *installpath* is the installation directory path you chose when you installed Content Manager OnDemand.

Procedure

To customize the arsodfxm exit, do the following steps:

1. Create a backup copy of the user exit.
2. Make the desired modifications to the exit.
3. Modify and run the sample ARSCODFX in ARS.V10R5M0.SARSINST to compile and link the user exit into the *installpath*/bin/exits directory where *installpath* is the path you chose when you installed Content Manager OnDemand.

Chapter 14. Performing routine maintenance

ODF routine maintenance should be performed on a periodic basis.

About this task

Routine maintenance of ODF tables is performed by the ARSMAINT program with the -o parameter.

The ARSODF program performs routine maintenance automatically each time it is started. It also automatically performs routine maintenance if nothing is processed by ODF for 24 hours. This is the same processing as if ARSMAINT with the -o parameter was entered on the command line.

Previous versions of ODF on z/OS used the ARSRDCLN program for routine maintenance. Previous versions of Report Distribution on AIX, Linux, and Windows used the ARSMAINT program with the -l parameter for routine maintenance. These methods are no longer supported.

ODF batch database cleanup

Overview

Performing ODF routine maintenance purges old information from the following ODF operational tables:

- Processed Report Bundle Table (PPT)
- Requested Distribution Table (DRT)
- Scheduled Report Table (DST)

Important: The tables must not be shared with any other tasks when ODF routine maintenance is in process.

Processing description

ODF routine maintenance removes all entries from the operational tables that are older than the number of days specified by the **ARSODF_EXPIRE_DAYS** parameter in the ARS.CFG configuration file.

Parameter values

The ARSMAINT program is typically run with the following parameters:

-I
The name of the Content Manager OnDemand instance to process.

-o
Expire OnDemand Distribution Facility table entries. Expired entries in the ARSDFPPT, ARSDFDRT, and ARSDFDST tables will be deleted. By default, entries in these tables older than 30 days are considered expired. The number of days can be changed by using the **ARSODF_EXPIRE_DAYS** parameter in the ARS.CFG configuration file. See “Verifying the ARS.CFG file” on page 15 for more information.

Sample ARSMAINT program call

The following shows how to run the ARSMAINT program with the option to expire ODF table entries:

```
arsmaint -I instancename -o
```

where *instancename* is the name of your Content Manager OnDemand instance.

IBM i On IBM i, the ARSMAINT program must be run in the QSHELL environment. See the *Content Manager OnDemand for i Administration Guide* for information about using the QSHELL environment.

Chapter 15. Accessibility information for Content Manager OnDemand

For information about accessibility features that are supported by this product, see the *IBM Content Manager OnDemand Administration Guide* for your platform.

Notices

This information was developed for products and services that are offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBMproducts. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).
Portions of this code are derived from IBM Corp. Sample Programs.
© Copyright IBM Corp. _enter the year or years_.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at www.ibm.com/privacy and IBM’s Online Privacy Statement at www.ibm.com/privacy/details the section entitled “Cookies, Web Beacons and Other Technologies” and the “IBM

Software Products and Software-as-a-Service Privacy Statement" at www.ibm.com/software/info/product-privacy.

Index

Q

R

S

T

U

X

Product Number: 5724-J33
5697-CM1
5770-RD1

SC19-3358-03