One Design
One Server
One User Experience

Configuring BIRT iServer

Contains information proprietary to:
Actuate Corporation, 951 Mariners Island Boulevard, San Mateo, CA 94404

www.actuate.com
www.birt-exchange.com

Document No. 120201-2-530303 June 28, 2012

Contents

About *Configuring BIRT iServer*

Configuring BIRT iServer discusses how to set up BIRT iServer and an Encyclopedia volume. The chapters in this guide are:

- *About Configuring BIRT iServer.* This chapter provides an overview of this guide.
- *Chapter 1. Performing basic configuration tasks.* This chapter discusses how to perform basic tasks such as logging in.
- *Chapter 2. Configuring an Encyclopedia volume.* This chapter covers how to add, remove, and back up an Encyclopedia volume.
- *Chapter 3. Using diagnostic, usage, and error logging.* This chapter describes how to configure and consolidate logs.
- *Chapter 4. Configuring e-mail notification.* This chapter describes how to configure e-mail notification about the completion of iServer jobs.
- *Chapter 5. Working with services.* This chapter discusses iServer architecture and services.
- *Chapter 6. Configuring the View service.* This chapter discusses how to test and improve performance iServer viewing.
- *Chapter 7. Configuring the Factory service.* This chapter describes how to tune the Factory service and control Factory processes using resource groups.
- *Chapter 8. Using resource groups.* This chapter describes how to control Factory services using resource groups.
- *Chapter 9. Clustering.* This chapter describes how to create and manage a cluster, add and delete a node, and modify server templates.
- *Chapter 10. Configuring Integration and Caching services.* This chapter describes how to optimize performance of information object-based reporting using Actuate Caching Service.

- *Chapter 11. Configuring iServer security.* This chapter covers the Report Server Security Extension.
- *Chapter 12. Archiving files.* This chapter describes online archiving and job completion notice purging.
- *Chapter 13. Printing documents.* This chapter describes setting up printing from iServer and customizing fonts.
- *Chapter 14. Connecting to data sources.* This chapter describes how to connect iServer to data sources, such as Oracle and DB2.
- *Chapter 15. Setting miscellaneous properties.* This chapter describes how to set iServer properties such as locales, ports, and process communication settings.

Chapter

1

Performing basic configuration tasks

This chapter contains the following topics:

- Understanding Configuration Console
- Using Simple view
- Using Advanced view

Understanding Configuration Console

Configuration Console provides the administrator with a convenient graphical user interface (GUI) for configuring iServer system. Configuration Console provides two configuration perspectives:

- Simple view

  Displays immediately after logging in to Configuration Console. In Simple view, the administrator can configure the basic functionality required to get iServer running quickly, such as starting or stopping iServer, updating a license, and specifying diagnostic logging, network, and e-mail notification settings.

- Advanced view

  Accessed from Simple view. In Advanced view, the administrator has access to all iServer settings available in Configuration Console, such as options for system, server, template, volume, partition, resource group, and printer settings. The administrator can perform any task supported in Simple view in Advanced view. Some settings, such as advanced template settings, may not be available through the console. These settings require manually updating XML configuration files to fine-tune iServer.

Launching Configuration Console

The administrator accesses Configuration Console using a browser.

How to launch Configuration Console

1. In Windows, launch Configuration Console from the Start menu by choosing the following options:

   Start➤Programs➤Actuate 11➤BIRT iServer Configuration Console

   Or, in a browser, launch Configuration Console on any platform by typing the following URL:

   `http://localhost:8900/acadmin/config`

2. Log in to Configuration Console using the password specified during installation.
3. Accept the default Language and Time zone, or choose another language and time zone. Figure 1-1 shows the login page for Configuration Console.
4. Choose Log In.

Figure 1-1 Logging in to Configuration Console

Troubleshooting problems launching the Console

To launch Configuration Console, the Actuate iServer 11 service must be running. By default, the service starts each time the machine reboots.

If the service does not start automatically, use Services in the Control Panel to start the service on Windows. Alternatively, you can stop and restart the service on any platform using commands on the command line.

How to start and stop the iServer service on Windows

1. Open Command Prompt from the Start menu as follows:

   Start➤Accessories➤Command Prompt

2. In Command Prompt, type the following command to stop the service:

   ```
   net stop "Actuate iServer 11"
   ```

3. To start the service, type:

   ```
   net start "Actuate iServer 11"
   ```

How to start and stop the iServer service on UNIX and Linux

To start the iServer service each time the machine reboots, log in as root, and type:

```
./AcServer/bin/update_rclocal.sh
```

If you want to start the iServer service manually, perform the following tasks:

1. Navigate to the AC_SERVER_HOME/bin directory. For example, type:

   ```
   cd /home/Actuate/AcServer/bin
   ```

2 To run the script to start the iServer service, type:

```
./start_srvr.sh
```

The term AC_SERVER_HOME refers to the iServer installation directory. By default, iServer installs in the following directories:

Windows: C:\Program Files\Actuate11SP4\iServer

Linux and UNIX: <installation directory>/AcServer

Using Simple view

In Simple view, the administrator specifies options for the following property categories:

- License information
  Display or update a license.
- iServer System
  Start or stop iServer System.
- Diagnostic logging
  Configure diagnostic logging levels, directory location, file size, and number of log files.
- Account settings
  Change and confirm a new administrator password.
- Network settings
  Specify hostname or IP address, and Management Console, Information Console, and Web Service API (IDAPI) port numbers.
- E-mail notification settings
  Specify SMTP server name, hostname or IP address, listen port, SMTP greeting, and sender name and e-mail address.

Displaying and updating the license

In Simple view, the administrator checks license options and updates the iServer license by choosing Show license and Update License, as shown in Figure 1-2. In Advanced view, the administrator checks license options by choosing System➤Properties➤License, then chooses Update License to update the iServer license.

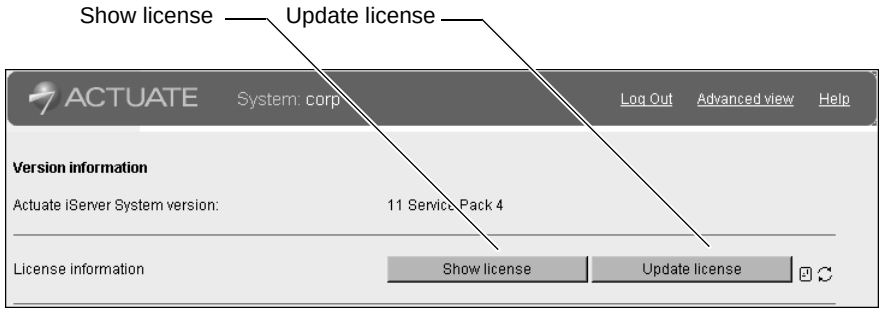

Figure 1-2 Viewing and updating the license file

To display the options covered by the license, choose Show License. Figure 1-3 shows the option list for an evaluation license.

The license for this system is a work unit license.

The CPU core limit is unlimited

Licensed for: Evaluation

Listed below are the Work Units licensed:

- BIRT Online (8)
- BIRT Factory (1)
- BIRT 360 (8)
- BIRT Data Analyzer (8)
- BIRT Studio (8)
- BIRT Spreadsheet Online (8)
- BIRT Spreadsheet Factory (1)
- e.Report Sync Factory (1)
- e.Report Viewing (1)
- e.Report Factory (1)

Listed below are the options currently licensed:

- e.Report (Actuate Basic Report) Option (Unlimited users)
- BIRT SpreadSheet Option (Unlimited users)
- Multi-Tenant Option (Unlimited users)
- e.Analysis Option (Unlimited users)
- e.Report Page Level Security Option (Unlimited users)
- Actuate Query Option (Unlimited users)
- BIRT Option (Unlimited users)
- BIRT SmartSheet Security Option (Unlimited users)
- BIRT Interactive Viewer Option (Unlimited users)
- BIRT Studio Option (Unlimited users)
- BIRT Page Level Security Option (Unlimited users)
- BIRT 360 Option (Unlimited users)
- BIRT Data Analyzer Option (Unlimited users)
- e.Report Data Connector Option (Unlimited users)

Listed below are the options currently not licensed:

- Multi-Tenant (Named User) Option

Figure 1-3 Viewing iServer license options

iServer supports the following license options:

- e.Report (Actuate Basic Report) Option
- BIRT SpreadSheet Option

- Multi-Tenant Option
- e.Analysis Option
- e.Report Page Level Security Option
- Actuate Query Option
- Actuate Analytics Option
- BIRT Option
- BIRT SmartSheet Security Option
- BIRT Interactive Viewer Option
- BIRT Studio Option
- BIRT Page Level Security Option
- BIRT 360 Option
- BIRT Data Analyzer Option
- e.Report Data Connector Option
- Multi-Tenant (Named User) or Work Unit Option

For more information on license options, see *Installing BIRT iServer for Windows* or *Installing BIRT iServer for Linux and UNIX*.

How to update an iServer System license

1. In License information, choose Update license.
2. License file appears, as shown in Figure 1-4.

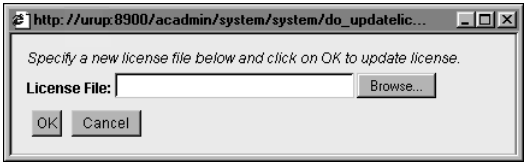

Figure 1-4 Selecting a new license file

3. Choose Browse and select a new license file.

   Choose OK. A message says that the license updated successfully. Close License file.
4. Restart iServer.
   1. In a stand-alone system, choose Stop system.
   2. Choose Start system.

The update takes effect when iServer restarts.

Stopping and starting iServer System

The administrator can stop and start iServer in both Simple or Advanced views. In Simple view, the list of user interface symbols, shown in Figure 1-5, indicate which fields require restarting a standalone iServer or, in a cluster, the entire iServer system. The asterisk symbol, *, indicates a required field that cannot be left blank. The exclamation point, !, indicates a field that takes a default value when left blank.

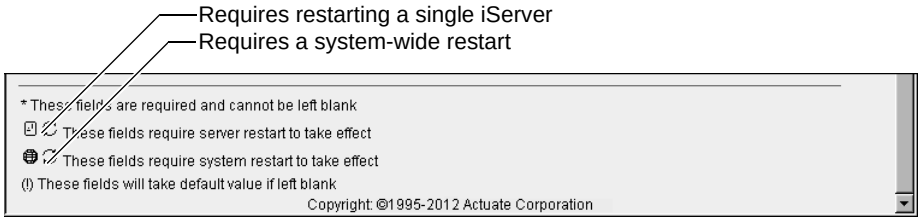

Figure 1-5 Interpreting iServer restart indicators

How to perform a system-wide restart

Simple view indicates the status of iServer system in Version information, as shown in Figure 1-6.

1 To stop a running iServer system, choose Stop system, as shown in Figure 1-6. In Version information, the status changes from online to offline, as shown in Figure 1-7.

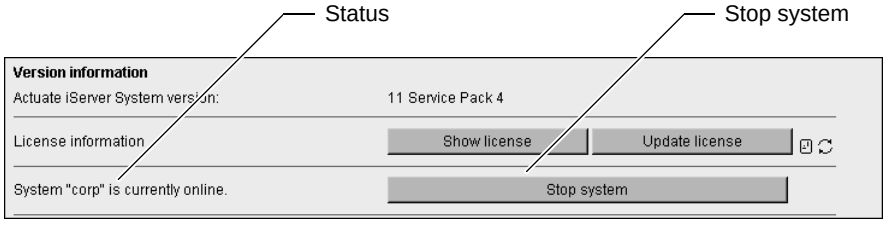

Figure 1-6 Performing a system-wide shutdown

2 To restart iServer, choose Start system, as shown in Figure 1-7.

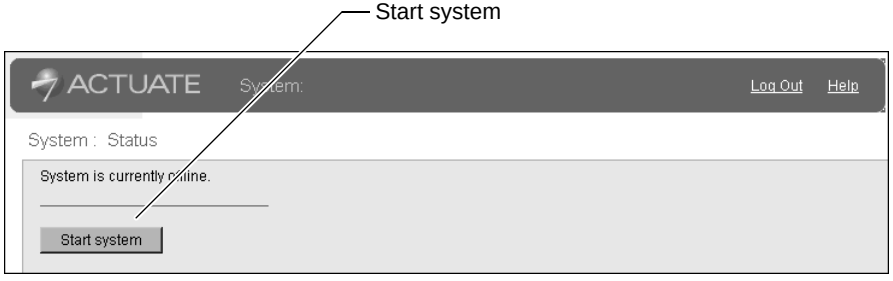

Figure 1-7 Performing a system-wide startup

Configuring diagnostic logging

iServer creates a log file when an internal process starts or other event occurs, such as an error or warning. The administrator can configure the level of diagnostic information that iServer writes to a log file when an error, warning, or other event occurs. The administrator can use this information to monitor system performance and troubleshoot problems.

Understanding diagnostic logging

The administrator configures diagnostic logging by specifying settings for the following properties:

- Level

  Specifies how much information iServer writes to a log, as shown in Figure 1-8. Select one of the following options to control the amount of information written to the diagnostic logs:

  - Severe

    Writes error messages only, as shown in Listing 1-1.

  - Warning

    Writes warning and error messages only.

  - Informational

    Writes warning and error messages including warning and error code descriptions.

  - Fine

    Writes warning and error messages with descriptions and limited diagnostic information.

  - Finest

    Writes the most detailed warning, error, descriptive, diagnostic, and tracing information possible, as shown in Listing 1-2.

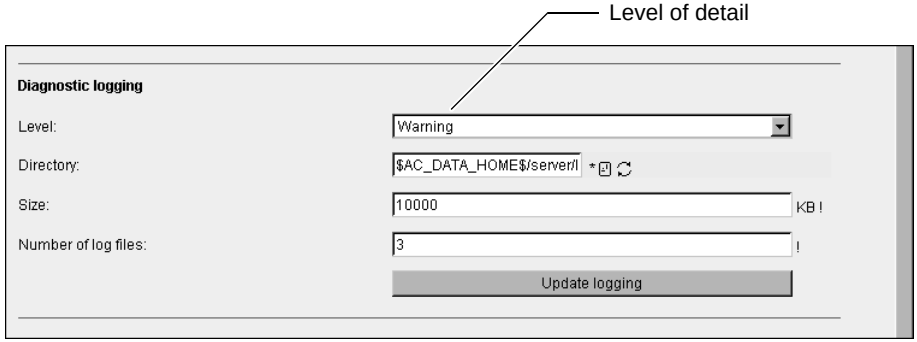

Figure 1-8 Viewing the default level of detail for diagnostic logging

Listing 1-1 contains an excerpt from a log that describes a problem caused by an application blocking port 25, the SMTP port. Setting the diagnostic logging level to Severe results in only minimal information about the problem.

Listing 1-1 Excerpt from a log containing only severe error messages

```
****0000008000*acmail.cpp*03438*09000*2012FEB26*10:18:56
 AcMailDefaultTransport: cannot open socket to
   exchangesvr.abcbank.com:25
 Error:10053
****0000008000*acmail.cpp*03438*09000*2012FEB26*10:18:59
 AcMailDefaultTransport: cannot open socket to
   exchangesvr.abcbank.com:25
 Error:10053
```

Listing 1-2 contains an excerpt from a log that describes the same problem. Setting the diagnostic logging level to Finest results in the maximum possible information.

Listing 1-2 Excerpt from a log containing the finest level of information

```
****0000008000*acmail.cpp*03943*06000*2012FEB26*10:37:49
 Message assigned to server iServer mail server
****0000008000*acmail.cpp*03438*09000*2012FEB26*10:37:49
 AcMailDefaultTransport: cannot open socket to
   exchangesvr.abcbank.com:25
 Error:10053
****0000008000*mailexcept.cpp*00322*08000*2012FEB26*10:37:49
 AcSMTPMailImpl::TryServer
 SMTP server name: iServer mail server
 Error code: 13018
 Error description: SMTP: Could not connect to the SMTP Server
   on the specified port.
 Parameter: exchangesvr.abcbank.com:25
****0000008000*mailloadbal.cpp*00073*06000*2012FEB26*10:37:49
 Moved to next server.
 Current Server Index: 0
 Current Backup Server Index: 0
****0000008000*mailloadbal.cpp*00386*06000*2012FEB26*10:37:49
 Current Index: 0
 Current Backup Index: 0
Using backup servers: false
 Server "iServer mail server": quota=0, state=Retryable
****0000008000*mailloadbal.cpp*00095*06000*2012FEB26*10:37:49
 Using backup servers.
```

Setting diagnostic logging to higher information levels can impact system performance and is recommended only for troubleshooting problems.

- Directory

  Specifies the log directory location. By default, iServer writes logs to the following directory, as shown in Figure 1-8:

  ```
  AC_DATA_HOME/server/log
  ```

  The administrator can configure iServer to write the logs to another directory in AC_DATA_HOME or other location on the network, provided permissions do not restrict access.

  A log file name contains embedded information about the process event. For example, the view server created the following log file, indicating the process name, server, date, and time:

  ```
  viewsrv11.exe.3736.urup.2012 MAR08_09_35_02_Pacific_Standard_
    Time.1.log
  ```

- Size

  Specifies the maximum log file size in kilobytes (KB). When the log file size reaches the limit, iServer starts writing to a new log file. To retain enough historical information to investigate a problem, particularly when using a verbose log level, such as Finest, increase the limit.

- Number of log files

  Specifies the maximum number of log files that iServer creates since the system last started. The administrator can use this property to manage log space usage in an environment with limited disk space.

  When a log file reaches the limit specified by Size, iServer creates another log file, until reaching the maximum number of log files specified in Number of log files. iServer then deletes the earliest log file and creates a new one. To conserve disk space usage in the log file directory, specify a lower limit.

  Stopping iServer 11 service breaks the log generation cycle. When iServer 11 service restarts, a new cycle begins. iServer creates the specified number of log files, but does not overwrite any pre-existing files.

How to configure diagnostic logging

1. In Diagnostic logging, perform the following tasks:
   1. In Level, accept the default level, Warning, or choose another level for all processes that write to the log files. For example, choose Finest, as shown in Figure 1-9.

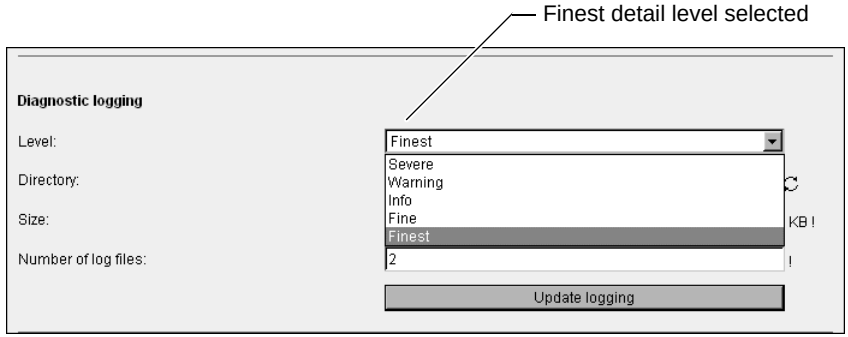

Figure 1-9 Selecting the finest detail level of detail for a log

2 In Directory, accept the default directory, AC_DATA_HOME/server/log, or specify a different directory for iServer to write log files. Ensure that the user account running iServer has permission to write to this location.

3 In Size, accept the default size limit for log files, 10000KB, or specify a different limit.

4 In Number of log files, accept the default, 3, or specify the maximum number of log files allowed in the log directory.

5 Choose Update logging, as shown in Figure 1-10.

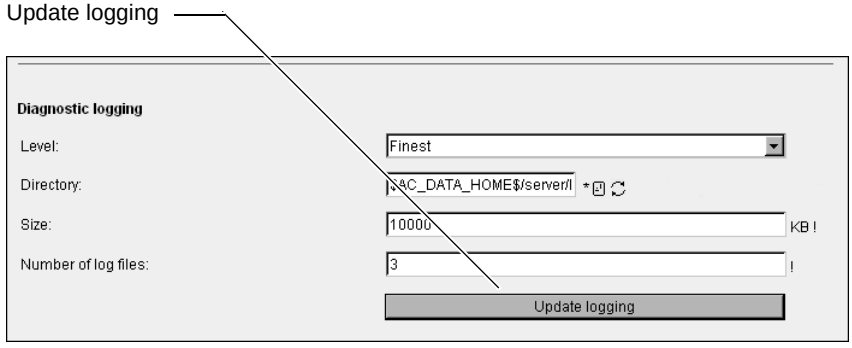

Figure 1-10 Default diagnostic logging properties

2 If you change Directory, restart iServer.

Configuring a diagnostic logging categories

A configuration change in the default Simple view applies to all categories of logging. In the default Simple view, Category does not appear in Diagnostic logging.

Advanced view—Server Configuration Templates—Settings supports changing diagnostic logging for specific types of logs. If the administrator configures one logging type differently from the others in Advanced view, Category appears in

Simple view, as shown in Figure 1-11.

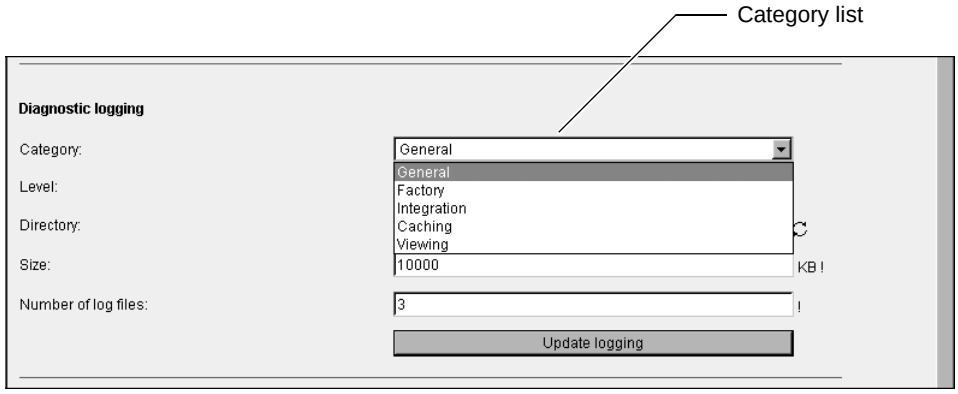

Figure 1-11 Viewing a list of categories in Simple view

The Category list includes the following log types:

- General
- Factory
- Integration
- Caching
- Viewing

Table 1-1 describes these categories, provides examples, and lists the log file names.

Table 1-1 Diagnostic logging categories and log files

Category	Logged events	Log file
General	Logs Encyclopedia volume events: ■ Validating login requests ■ Creating Encyclopedia volume folders ■ Adding files ■ Managing e-mail notification Records other events that occur outside factory, integration, caching, and viewing service processes	encycsrvr11
Factory	Logs Factory service events: ■ Running designs ■ Generating queries ■ Printing documents	facsrvr11

Table 1-1 Diagnostic logging categories and log files

Category	Logged events	Log file
Integration	Logs Integration service events regarding information objects that use data from multiple data sources	intsrvr11
Caching	Logs Caching service events regarding information object cache	cachesrvr11
Viewing	Logs Caching service events: ■ Viewing documents in DHTML format ■ Converting documents to formats, such as Excel and PDF ■ Handling requests to download files from an Encyclopedia volume	viewsrv11

How to configure diagnostic logging by category

1. In Diagnostic logging, in Category, choose a log category to configure. For example, choose General.
2. In Level, Directory, Size, and Number of log files, accept the default values or specify new values.
3. Choose Update logging.
4. If you change Directory, restart iServer.

Configuration Console writes setting changes to the acserverconfig.xml file in AC_DATA_HOME/config/11SP<service pack number>. Table 1-2 lists the property names that appear in Configuration Console with the corresponding parameter names in acserverconfig.xml, including default settings, ranges, and when the property change takes effect.

Table 1-2 Diagnostic logging parameters

Property name	Parameter name	Default	Range	Takes effect
Log level	LogLevel	8000	0 - 9000	Immediate
Log size	LogSize	10000 KB		Immediate
Number of log files	NumLogs	3		Immediate

For more information about diagnostic logging, see Chapter 3, “Using diagnostic, usage, and error logging.”

Changing the Configuration Console password

During installation, the installer specifies the Configuration Console password. In Simple View, the administrator can change this password in Account settings. In Advanced view, the administrator chooses System➤Properties to the system password.

How to change the password for configuring iServer

1. In Account settings, type the old and new passwords.
2. In Confirm system password, type the new password again.
3. Choose Change password, as shown in Figure 1-12.

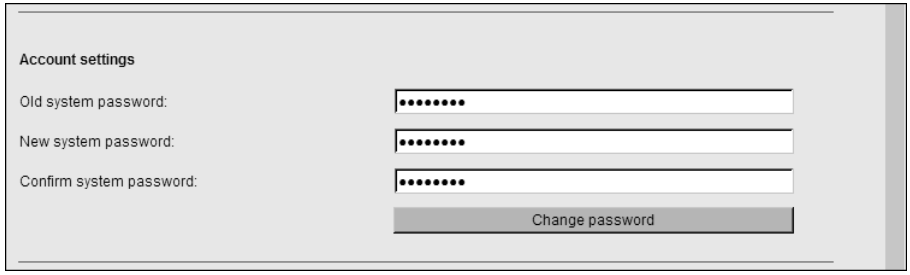

Figure 1-12 Changing the administrator password for Configuration Console

Configuring network settings

iServer communicates with Management Console, Information Console, and Web Service APIs through network ports. In Network settings, the administrator can change the following items:

- iServer hostname or IP address
- Management Console and Information Console port
- Web Service API (IDAPI) port

Any change to a network setting requires restarting iServer.

If you change the port numbers assigned by the installation program, test that these ports work with these console and web service applications. By default, the Microsoft Vista firewall blocks port 8900. Ensure this port is unblocked.

How to change network settings

1. In Network settings, in Server hostname or IP address, type the new iServer host name or IP address, as shown in Figure 1-13.

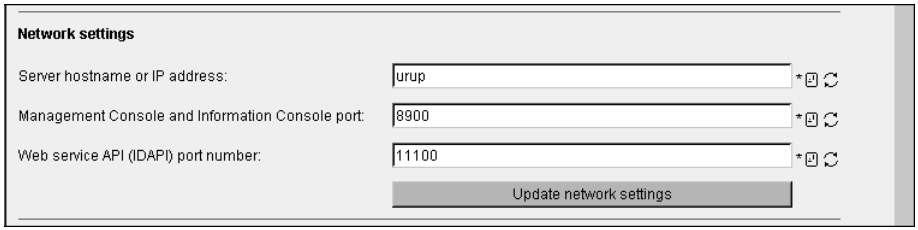

Figure 1-13 Network settings

2 In Management Console and Information Console port, type the port number used by the application container to listen for requests from the consoles.

3 In Web service API (IDAPI) port, type the port number used by the Actuate Information Delivery API.

4 Choose Update network settings.

5 Restart iServer.

Configuring e-mail notification settings

iServer can notify users by e-mail that a job completed or failed. The e-mail includes a link to the document in HTML format. iServer can attach the document in other formats to the e-mail. A template, which the administrator can modify, specifies the message content, as shown in Figure 1-14.

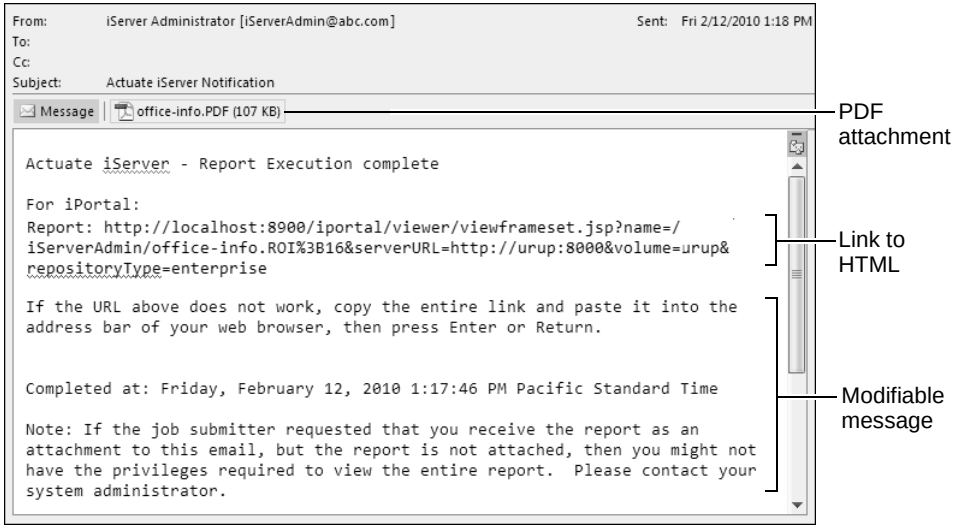

Figure 1-14 Viewing an example of an SMTP e-mail notice

The e-mail message template, acnotification.xml, is located in AC_SERVER_HOME\etc.

How to configure e-mail notification settings

To use a Simple Mail Transfer Protocol (SMTP) e-mail server for iServer e-mail notification, configure the properties in Server e-mail notification settings.

1. In SMTP Server name, type an arbitrary name for the new mail server that appears in the list of SMTP servers in Configuration Console. For example, type:

   ```
   iServer mail server
   ```

2. In Hostname or IP Address, type the IP address or the fully qualified domain name of the mail server. For example, type:

   ```
   exchangesvr.abcbank.com
   ```

3. In Listen port, accept the default port number, that iServer uses for e-mail notification, or specify the number of a free port. In Figure 1-15, the default listen port number is 25.

4. In SMTP greeting, accept the default, which is no entry. iServer sends HELO and appends the blank suffix to the greeting during protocol exchanges with the mail server. Alternatively, specify a suffix to the greeting. iServer sends HELO <suffix>.

5. In Sender e-mail address, specify the e-mail address that appears in the From line of the e-mail notification. iServer also sends an alert to this address when the mail server cannot deliver an e-mail notification to a user. For example, type:

   ```
   iServerAdmin@abc.com
   ```

6. In Sender name, specify the name that appears in the From line of the e-mail notification. For example, type:

   ```
   iServer Administrator
   ```

   Figure 1-15 shows Server e-mail notification settings.

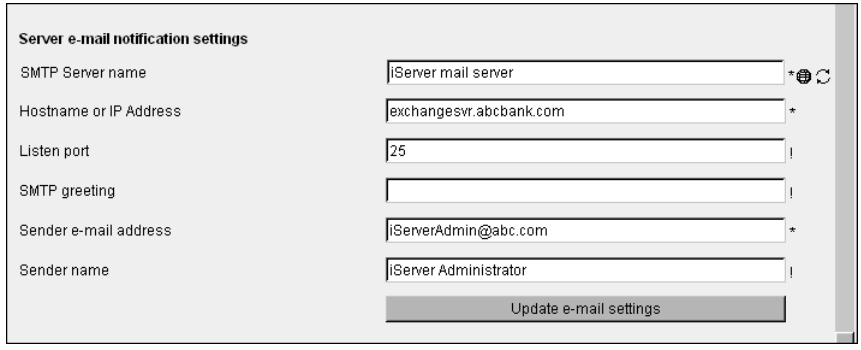

Figure 1-15 Server e-mail notification settings

7 Choose Update e-mail settings.

8 Restart iServer.

For more information about e-mail notification, including additional advanced settings, see Chapter 4, “Configuring e-mail notification.”

Configuring the URL for e-mail notification

The administrator can configure iServer to send an e-mail to an Information Console user about a completed job. When a user receives a default e-mail notice about a completed job, the e-mail message contains a link to the completed report.

How to configure the URL for e-mail notification

1 In URL for e-mail notification, in Information Console URL prefix, specify the machine name, port, and context root of Information Console, as shown in Figure 1-16. This property is also configurable in Email notification on Advanced View in Volume—Properties.

Information Console URL prefix can also be used to specify a hyperlink, such as a drill-through hyperlink, containing the appropriate Information Console context string in a PDF document. For more information on customizing e-mail notifications and Information Console URL prefix, see Chapter 4, “Configuring e-mail notification,” later in this book.

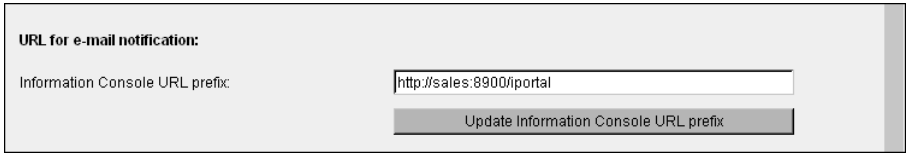

Figure 1-16 Adding a URL prefix to e-mail notices

In this example, the machine name is sales, the port is 8900, and the context root of Information Console is iportal.

2 Choose Update Information Console URL prefix.

A report opens in Information Console when a user clicks the link. When the user clicks a link in a completion notice, browser security settings can sometimes prevent the display of the page. To avoid this problem, set browser security to medium low. For example, in Internet Explorer, choose Tools➤Internet Options➤Security➤Trusted Sites➤Medium Low.

Using Advanced view

From Simple view, choose Advanced view, as shown in Figure 1-17.

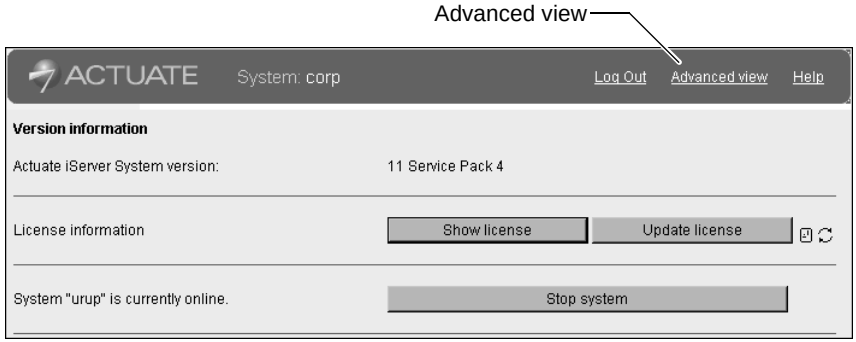

Figure 1-17 Choosing Advanced view from Simple view

To go back to Simple view, choose Simple view, as shown in Figure 1-18.

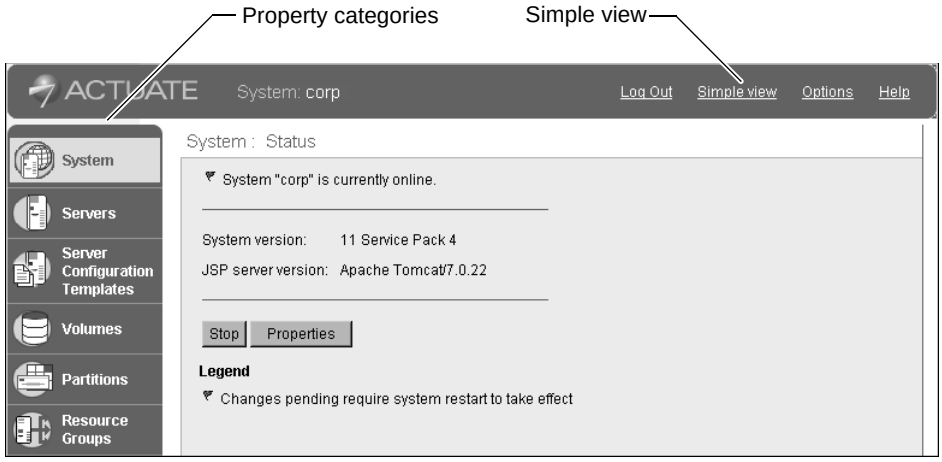

Figure 1-18 Configuring iServer in Advanced view

In addition to Simple view, Advanced view also contains the following items:

- Log Out
  Logs out of Configuration Console.
- Options
  Provides options for configuring the Configuration Console display.
- Help
  Provides access to iServer system documentation in a browsable format. For more information about installing, accessing, and navigating documentation in Help, see *Installing BIRT iServer for Windows* or *Installing BIRT iServer for Linux and UNIX*.

Figure 1-18 shows Advanced view with System selected. The left menu provides access to user interfaces that allow the administrator to view and change the settings for the following iServer system property categories:

- System
  System name, usage and error logging, notification, regional settings, license, and other advanced properties.
- Server
  Server, application container, Process Management Daemon (PMD), SOAP, and enabled service settings.
- Server Configuration Templates
  Factory, Message Distribution, Viewing, Integration, Caching services. Other advanced options, such as NetOSI File Types, filetype driver information, diagnostic logging, process management, and other settings.
- Volumes
  Metadata database, schema, and volume property settings.
- Partitions
  Partition name, path, status, and other settings.
- Resource Groups
  Name, description, status, report type, volume, work unit type, start arguments, priority settings and template assignments for resource groups associated with e.Report, information object, BIRT, and Spreadsheet jobs.
- Printers
  Name, path, spool command, and other template settings.

For more information about configuring iServer system properties, see the related chapters later in this book. The remaining sections in this chapter describe the options for configuring Configuration Console display settings.

Setting Configuration Console options

The administrator can change the way Configuration Console displays the following settings:

- Regional settings, such as locale and time zone.
- The number and order of columns that appear in server, template, volume, partition, or printer lists.
- About, providing version information on installed iServer system software

To view or change display options, choose Options in Advanced view, as shown in Figure 1-19.

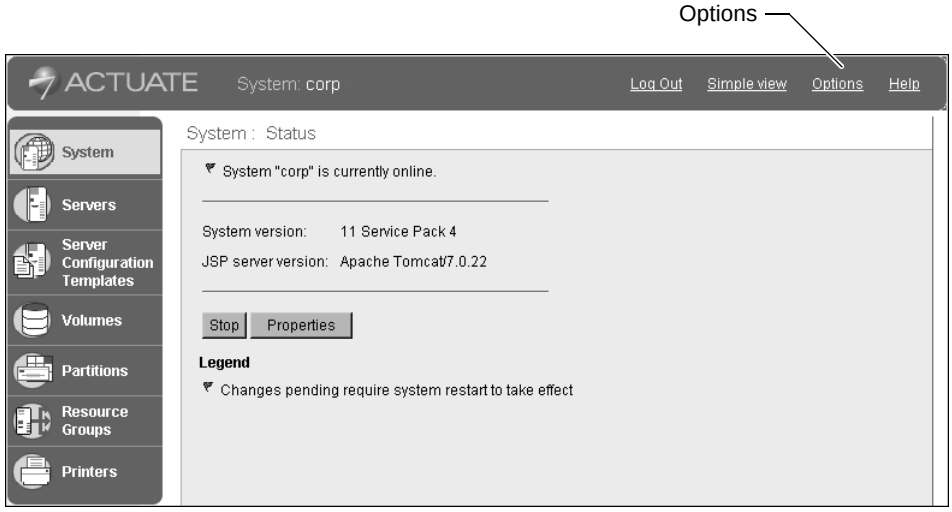

Figure 1-19 Setting options

Choose an option to modify. Figure 1-20 shows the General option selected.

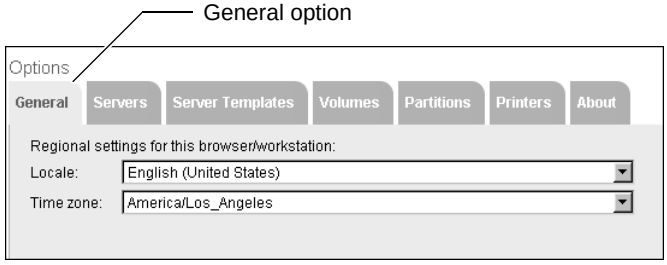

Figure 1-20 Selecting the General option

In Options—General, change the regional settings to correspond to the current locale and time zone, if different from the settings specified during installation.

Changing server, template, volume, or partition options

In Options—Servers, Server Templates, Volumes, Partitions, or Printers, change the number and order of the columns that appear in server, template, volume, partition, or printer lists by performing the following tasks:

- Use the left and right arrows to move column names between Available columns and Selected columns.
- Use the up and down arrows to change the order in which the information appears.

Figure 1-21 shows the list of available and selected columns in Options—Volumes.

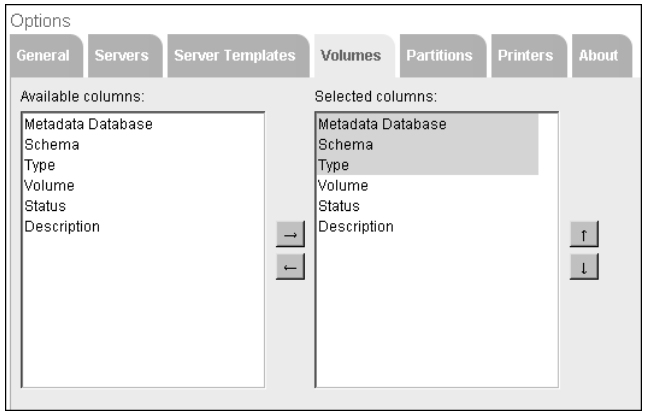

Figure 1-21 Modifying columns in Options—Volumes

In Selected columns, the order in which the columns appear vertically determines the horizontal order in the displayed list. You cannot hide or change the order of highlighted columns in Selected columns. For example, you cannot hide or change the order for the Metadata Database, Schema, or Type columns in Options—Volumes.

How to change the list of servers, volumes, or partitions

1. In Advanced view of Configuration Console, choose Options.

   Options—General appears.

2. Choose Servers, Server Templates, Volumes, Partitions, or Printers.

   The available and selected columns for the list appears.

3. Change the column display options:

   - To add columns to the list, select the columns in Available columns. Choose the right arrow to move the selected columns to Selected columns.
   - To remove columns from the list, select the columns in Selected columns, and choose the left arrow.
   - To change the order in which the columns appear in the list, select a column in Selected columns, and choose the up or down arrow to reposition the column in the list, as shown in Figure 1-21.

   Choose OK.

In the left menu of Advanced view, when you choose Servers, Server Configuration Templates, Volumes, Partitions. Resource Groups, or Printers, a related list appears. Figure 1-22 shows an example of the Volumes list.

In Figure 1-22, the list displays the Metadata database, Schema, Volume, Status, and Description columns for each volume, such as corp. The changes made in Options—Volume alter the number and order of columns that appear in this list and similar lists accessed from the left menu of Advanced view.

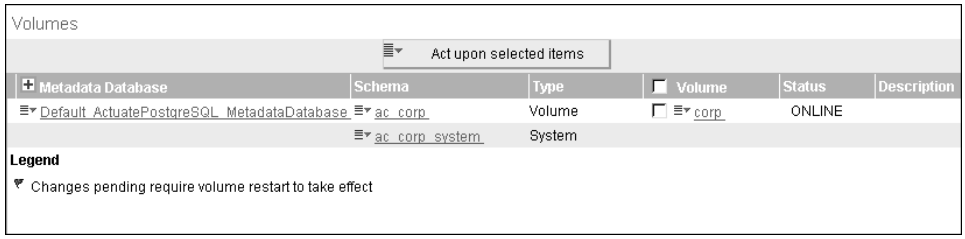

Figure 1-22 Viewing the volumes list

About the servers list

Table 1-3 describes the columns available in Options—Servers that display in the servers list.

Table 1-3 Columns available for display in the node list

Column name	Description
Name	Name of the server, typically the machine name.
Template	Name of the template that defines the configuration properties for the server.
Status	Current status of the machine. Status can be master, online, offline, or stopping. Status also lists the following services configured for iServer on the specified machine: ■ M for Message Distribution service ■ F for Factory service ■ V for View service ■ C for Caching service ■ I for Integration service
Description	Description field from the iServer definition.
Operating system and version	Server machine's operating system and version.
Actuate version	iServer release number.
Current requests	Current number of active requests.

A flag appears when you must restart iServer to apply updates to the configuration.

About the server templates list

Table 1-4 describes the columns available in Options—Server Templates that display in the server configuration templates list.

Table 1-4 Columns available for display in the server templates list

Column name	Description
Template	Name of the template that defines the configuration properties for the server.
Server	Name of the server, typically the machine name.

About the volumes list

Table 1-5 describes the columns available In Options—Volumes that display in the Encyclopedia volume list.

Table 1-5 Columns in the Encyclopedia volumes list

Column name	Description
Metadata Database	Names of the metadata databases in this iServer system.
Schema	Names of the schemas. One database can contain one or more schemas.
Type	Type of schema, either Volume or System.
Volume	Names of the volumes. One schema can contain one or more volumes.
Status	Current status of the volume. The status of the volume is online or offline.
Description	Description field from the volume definition.

About the partitions list

Table 1-6 describes the columns available in Options—Partitions that display in the partitions list.

Table 1-6 Columns in the partitions list

Column name	Description
Name	Name of partition.
Status	Current status of the partition. The status of the partition is active or inactive.
Volume	Names of the volumes associated with the partition if assigned.

About the printers list

Name is the only available column in Options—Printers. Name is the name of the printer.

Viewing release information

In Options—About, you can view version information about the iServer system software installation, as shown in Figure 1-23.

Figure 1-23 Viewing iServer System general information

Chapter

2

Configuring an Encyclopedia volume

This chapter contains the following topics:

- Understanding an Encyclopedia volume
- Configuring a partition
- Working with databases, schemas, and volumes

Understanding an Encyclopedia volume

An Encyclopedia volume consists of files such as BIRT designs, documents, spreadsheets, and information objects. iServer creates a default Encyclopedia volume during installation. The installation process names the default Encyclopedia volume the same as the machine name, as shown in Figure 2-1.

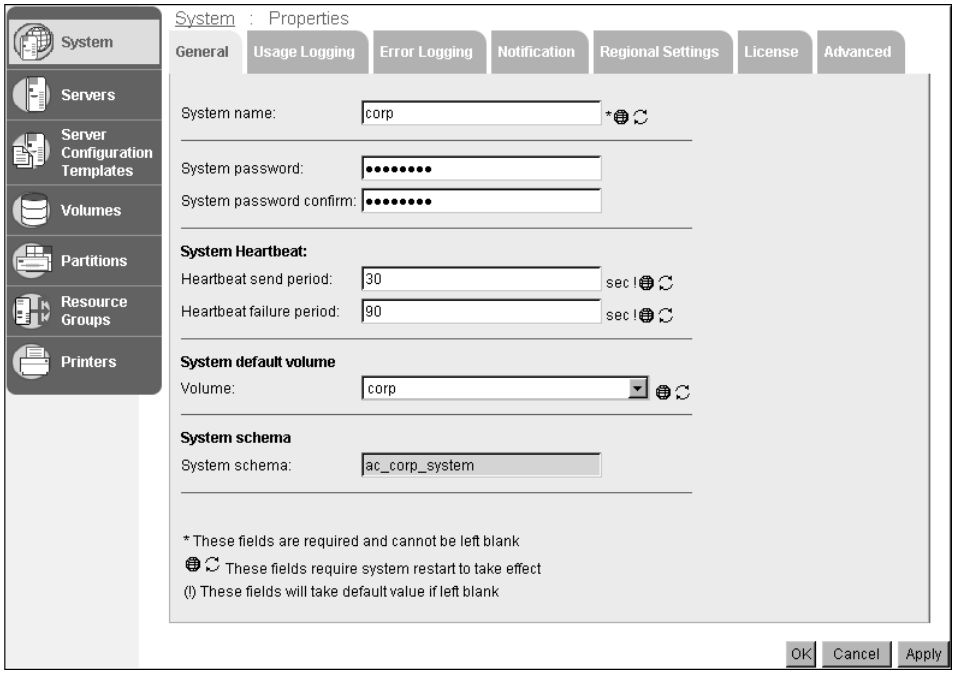

Figure 2-1 Viewing general system properties

The default Encyclopedia volume directory is AC_DATA_HOME/encyc. By default, AC_DATA_HOME on a Windows system is Actuate\iServer\data. On a Linux or UNIX system, AC_DATA_HOME is AcServer/data.

In an out-of-the-box (OOTB) installation, iServer stores Encyclopedia volume metadata, such as information about users, roles, groups, and job schedules, in the PostgreSQL database installed with iServer. The PostgreSQL database resides in AC_DATA_HOME/encyc. iServer stores configuration metadata in the database separately from data such as designs, documents, information objects, and other iServer data objects, which are stored in the file system. Files containing data have the .dat file-name extension in the file system, regardless of their Actuate file type. By default, the .dat files reside in AC_DATA_HOME /encyc/file.

Configuring a partition

An iServer partition is a physical disk location used to store Encyclopedia volume data files. Every Encyclopedia volume must have an iServer partition.

An Encyclopedia volume supports using multiple iServer partitions. You can expedite input and output processing by using multiple partitions that operate across separate physical disks. iServer transparently manages the allocation of files among partitions. Encyclopedia volumes cannot share a partition.

The default Encyclopedia volume resides in the primary partition, DefaultPartition. The default path of the primary partition is AC_DATA_HOME /encyc. Figure 2-2 shows the path of the primary partition on Windows.

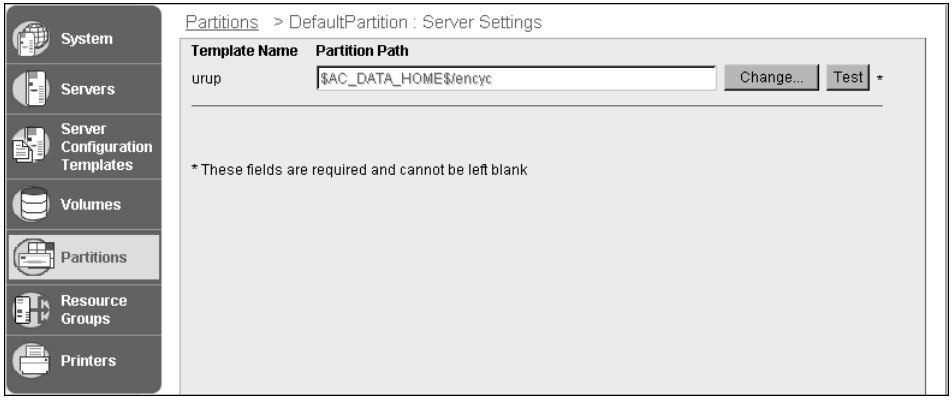

Figure 2-2 Default Partition path

The administrator can change the location of the primary partition, but cannot remove it. If no other partitions exist, the primary partition stores all designs, documents, and other data files.

Adding a partition

Add a partition before attempting to add a new Encyclopedia volume to iServer.

How to add a partition to iServer

1. Make a directory for the partition on the physical drive of the machine or storage device that iServer can access. For example, make a directory called encyc2 in AC_DATA_HOME.
2. From Advanced view of Configuration Console, choose Partitions.
3. In Partitions, choose Add partition.
4. In Partition name, specify a name. For example, name the partition, Partition2.

In Partition Path, specify the fully qualified path to the partition directory, as shown in Figure 2-3. Choose OK.

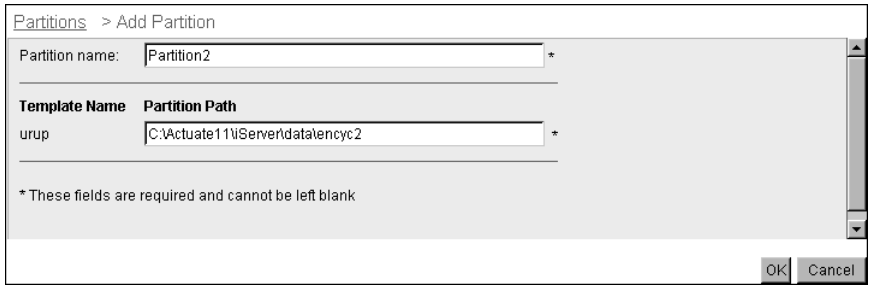

Figure 2-3 Adding a partition

5 In Partitions, choose the new partition, Partition2, from the list of partitions, as shown in Figure 2-4.

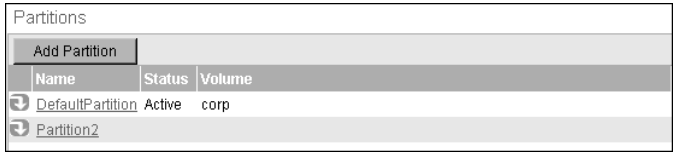

Figure 2-4 Choosing the new partition

6 In Server Settings, choose Test, as shown in Figure 2-5.

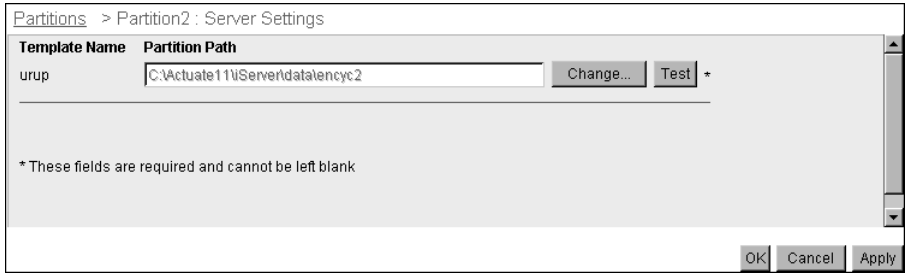

Figure 2-5 Testing the new partition

If the test succeeds, the message in Figure 2-6 appears. If the test fails, check that the directory named in the partition path exists.

Figure 2-6 Choosing OK after successful partition path test

Choose OK.

Viewing partitions

After the administrator assigns a partition to an Encyclopedia volume, Partitions displays one of the following status conditions and the name of the Encyclopedia volume.

- Active
  An Encyclopedia volume is using the partition.
- Phaseout
  An Encyclopedia volume is moving data out of the partition.
- Unused
  The partition is assigned to an Encyclopedia volume, but it is not using the partition.

From Advanced view of Configuration Console, choose Partitions to view the partitions that are available, as shown in Figure 2-7.

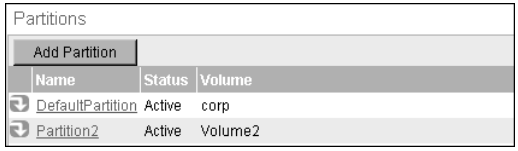

Figure 2-7 Displaying the list of partitions

Working with databases, schemas, and volumes

Beginning with Actuate 11, iServer stores information related to users, roles, groups, files and folders, or metadata, in a third-party relational database management system (RDBMS). By default, iServer uses the PostgreSQL RDBMS for this purpose. In Release 11, iServer also supports the use of an Oracle, DB2, and SQLServer database.

A database can contain one or more schemas. An Encyclopedia volume schema can relate to one or more volumes. When installing BIRT iServer Release 11 with either the OOTB PostgreSQL or a third-party database, the install program creates two schemas, one for Encyclopedia metadata, and one for the system metadata. In Advanced view of Configuration Console, on Volumes, the metadata database appears in the first column, the schemas the database contains appear in the second column, and the volume the Encyclopedia volume schema relates to appears in the third column, as shown in Figure 2-8.

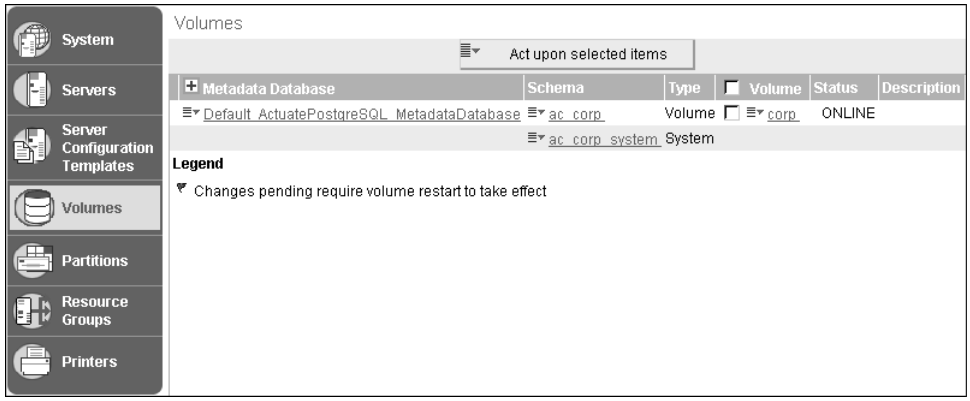

Figure 2-8 Viewing Volumes on Advanced view of Configuration Console

Configuring a metadata database

On Volumes, an administrator can create, configure, or delete a metadata database used to contain Encyclopedia volume schema.

How to add a metadata database

1. Log in to Configuration Console and choose Advanced view. From the side menu, choose Volumes.
2. Point to the icon next to the Metadata Database heading and choose Add new metadata database, as shown in Figure 2-9.

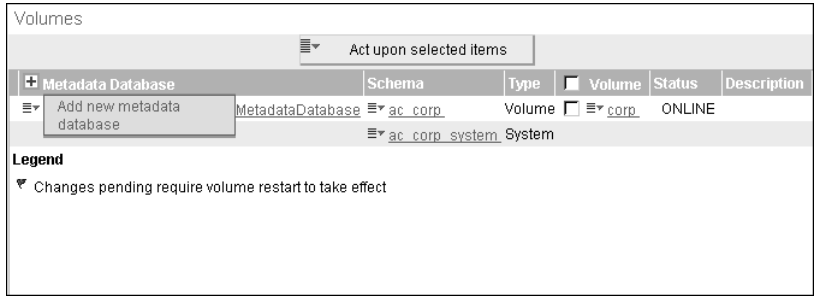

Figure 2-9 Choosing to add a new metadata database

3. On New Metadata Database, as shown in Figure 2-10, perform the following tasks:
   1. In Metadata database name, type a name for the metadata database.
   2. In Database type, select the type of database to create.
   3. In Database server, type the host name of the machine containing the database, such as localhost.

4 In Database name, type a name for the database.

5 In Connection login, type the database user name.

6 In Connection password, type the database user name password.

7 In Database port, specify a port number, or accept the default value.

Choose OK.

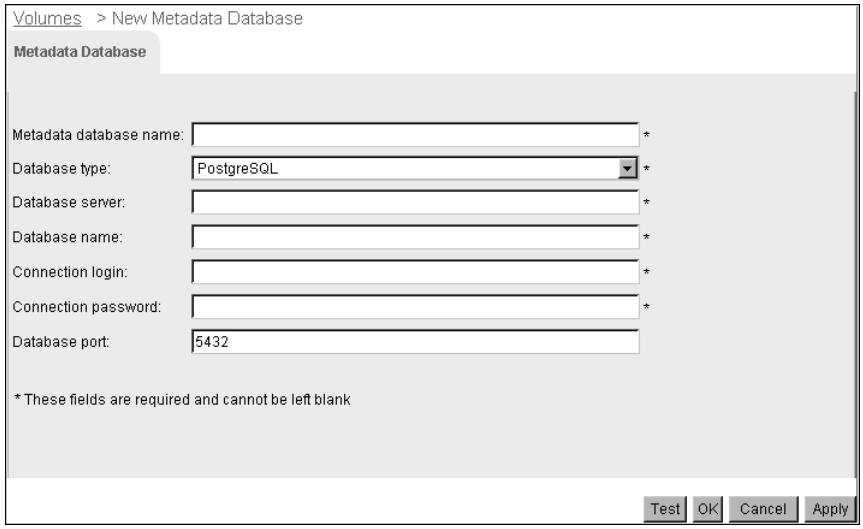

Figure 2-10 Adding a new metadata database

How to edit metadata database properties

1 On Volumes, point to the icon next to a database and choose Properties.

2 On Metadata Database, the settings are the same as when you add a new metadata database, except that when editing database properties, Metadata Database includes the option to confirm the connection password if you change it. Choose Advanced.

3 On Advanced, choose MetadataDatabaseProperties.

4 On MetadataDatabaseProperties, as shown in Figure 2-11, you can specify the following additional database properties:

- Type of database
- Database tablespace name
- Database tablespace path
- Super user name
- Super user password

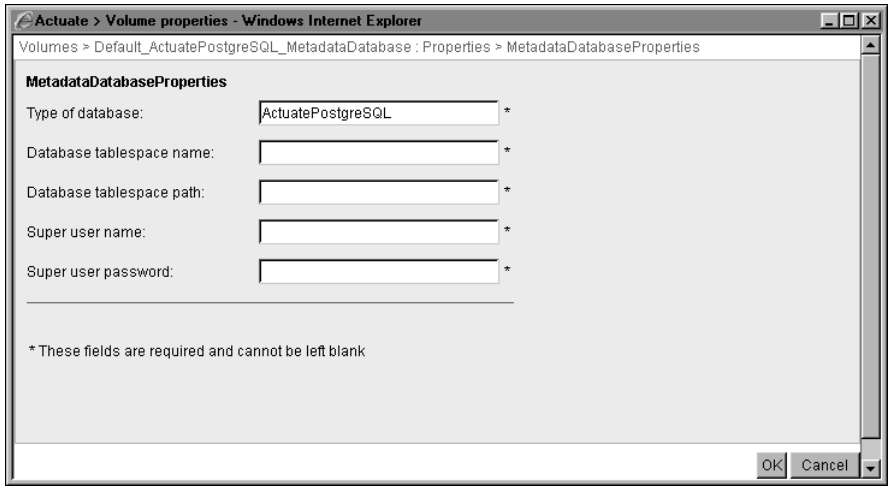

Figure 2-11 Setting additional metadata database properties

5 On Advanced, choose Database Connection Pool Manager Settings.

6 On Database Connection Pool Manager Settings, as shown in Figure 2-12, you can specify the following database connection properties:

- Initial size of the connection pool
- Minimum number of open connections in the connection pool
- Maximum number of open connections in the connection pool
- Maximum number of open connections in the internal connection pool
- Maximum number of open connections in the system connection pool
- Maximum number of open connections in the cluster connection pool
- Maximum idle time for a connection before removing it from the connection pool
- Max connection wait time in seconds
- Max query execution time in seconds (use zero value to indicate no limit)
- PreparedStatement cache size per connection (use zero value to indicate no caching)

To acquire a new JDBC connection for each database query, type a value of 0 in Minimum number of open connections in the connection pool and in Maximum number of open connections in the connection pool.

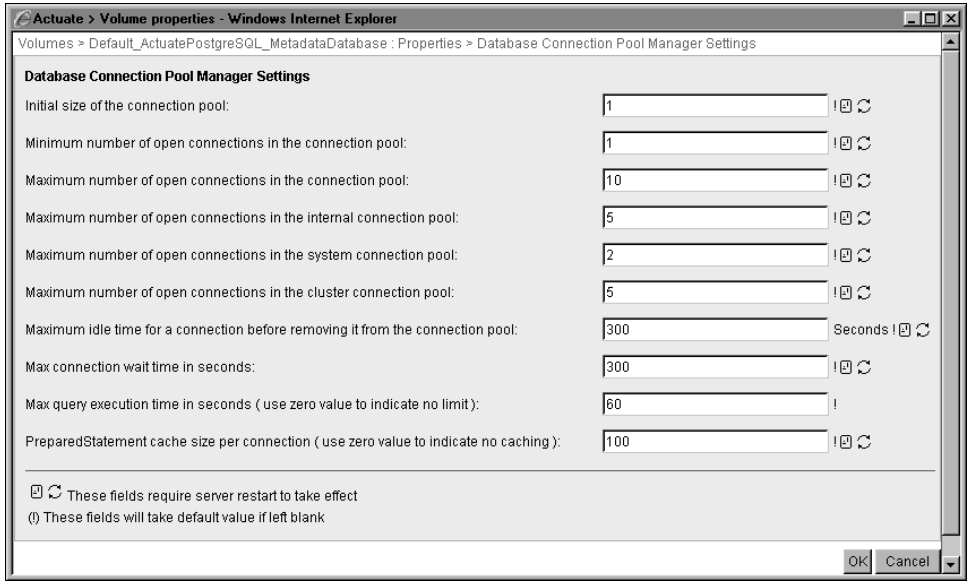

Figure 2-12 Configuring Database Connection Pool Manager Settings

Configuring a schema

Additionally, the administrator can add, configure, or remove a schema used to contain Encyclopedia volume metadata.

How to add a schema

1. Log in to Configuration Console and choose Advanced view. From the side menu, choose Volumes.
2. On Volumes, point to the icon next to a metadata database and choose either Add system schema, or Add volume schema. Figure 2-13 shows Add volume schema. The procedure for adding either schema type is the same. The default metadata database is Default ActuatePostgreSQL MetadataDatabase.

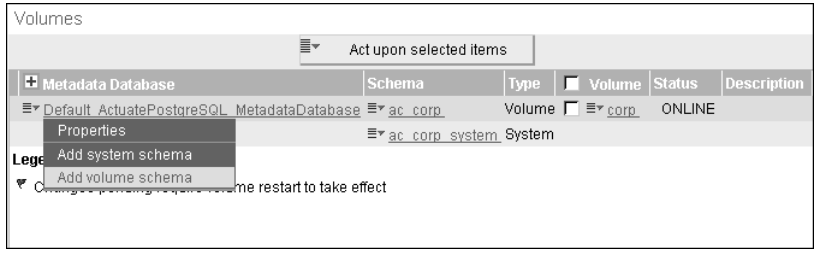

Figure 2-13 Choosing to add a schema

3 On New Volume Schema, as shown in Figure 2-14, perform the following tasks:

1 In Schema name, type a name for the new schema. The name must be less than 30 characters.

2 In Schema owner name, type the schema owner name.

3 Type and confirm a password for the Schema owner.

4 In Database superuser, type the database superuser name. For the PostgreSQL RDBMS that installs with iServer by default, the PostgreSQL superuser name is postgres.

5 In Database superuser password, type the password that the installer specified for the database superuser during the iServer installation.

Choose OK.

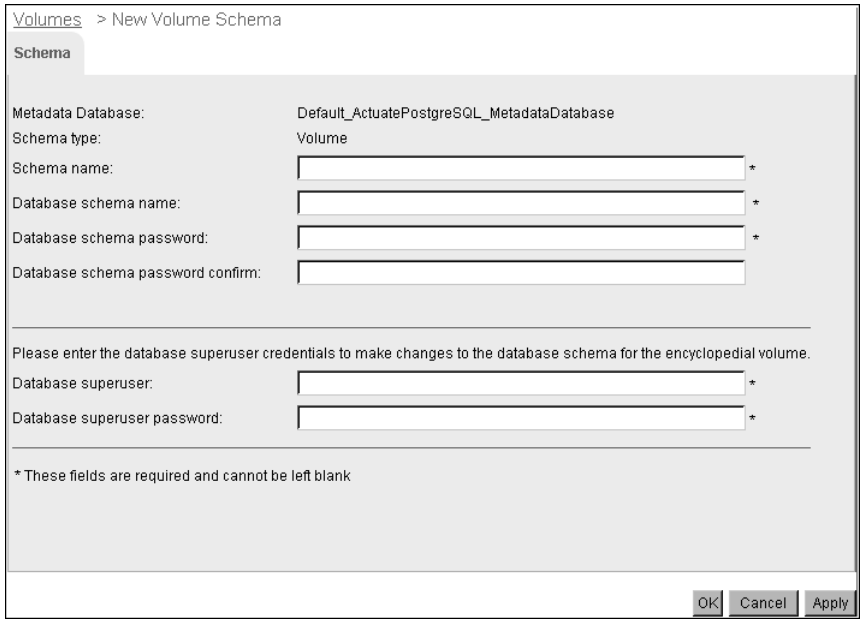

Figure 2-14 Adding a new schema

How to edit schema properties

1 On Volumes, point to the icon next to a schema and choose Properties, as shown in Figure 2-15.

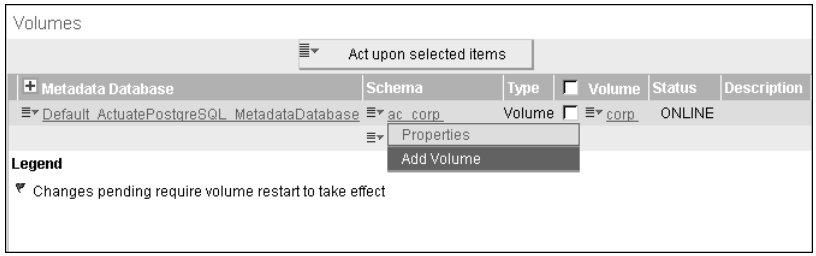

Figure 2-15 Choosing to edit schema properties

2 In Schema, you can change the schema owner name and password, as shown in Figure 2-16.

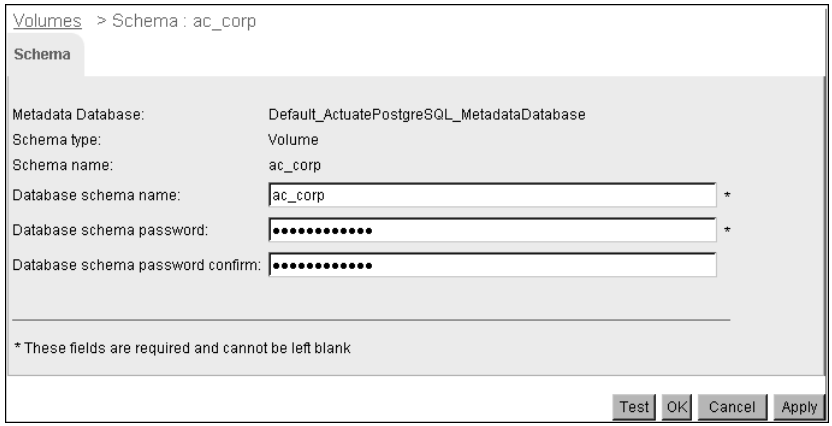

Figure 2-16 Editing schema properties

3 To test the connection, choose Test.

Choose OK.

Configuring an Encyclopedia volume

The administrator can also add, configure, or remove a volume contained in a schema within a metadata database.

Adding an Encyclopedia volume

Unless a partition, unassigned to an Encyclopedia volume, already exists, you must first create a new partition to assign to the Encyclopedia volume. Next, add the Encyclopedia volume to iServer by performing the following tasks:

How to add an Encyclopedia volume

1 Log in to Configuration Console and choose Advanced view. In Advanced view, choose Volumes.

2 On Volumes, point to the icon next to a schema and choose Add Volume, as shown in Figure 2-17.

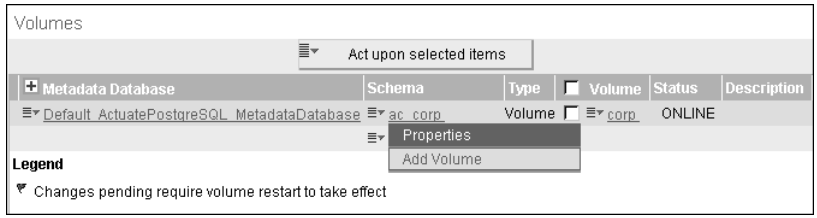

Figure 2-17 Choosing to add a volume

3 On New Volume—General, perform the following steps:

1 Type a name for the new volume. For example, type Volume2, as shown in Figure 2-18.

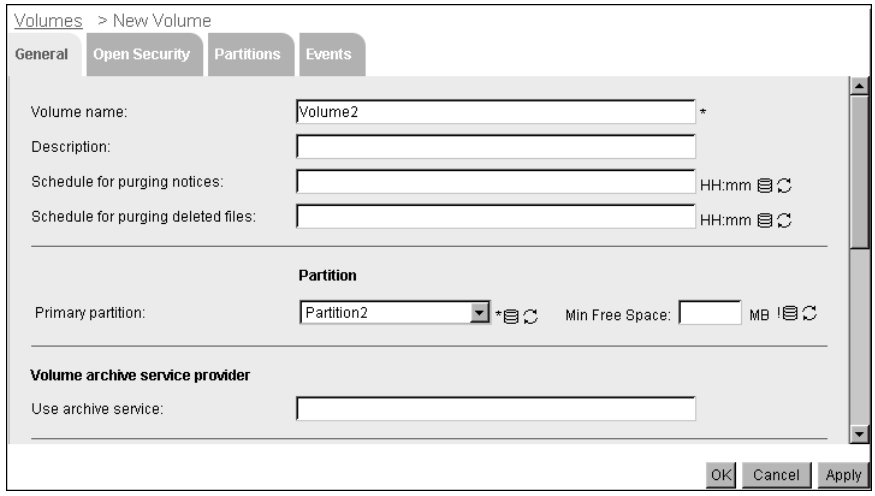

Figure 2-18 Specifying general properties

2 In Primary partition, accept the default or select another unassigned partition. For example, accept Partition2.

4 Choose Partitions, and start the partition for the new Encyclopedia volume by performing the following steps:

1 In Available partitions, select a partition, then move it to Selected by choosing the right arrow.

2 In Selected partitions, select the partition. Choose Start, as shown in Figure 2-19.

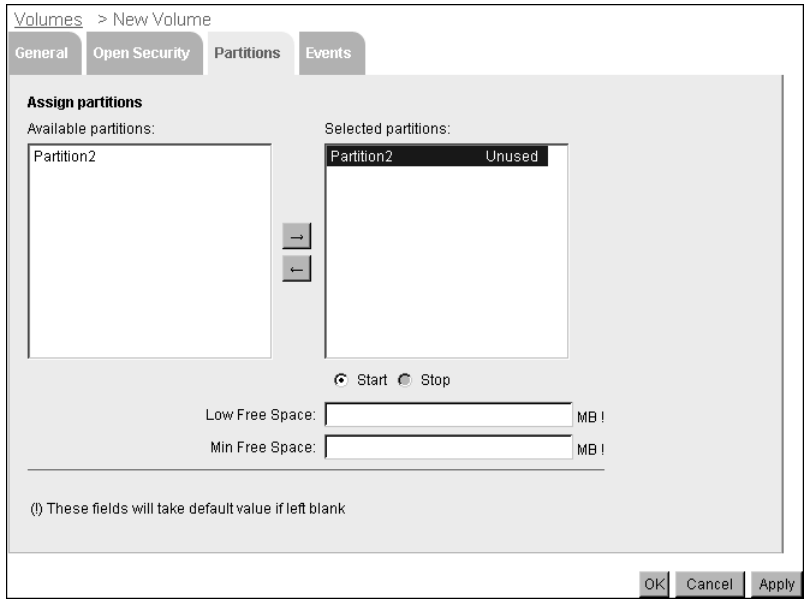

Figure 2-19 Starting the partition

Choose OK.

5 In Volumes, point to the arrow next to the new volume name, and choose Take online, as shown in Figure 2-20.

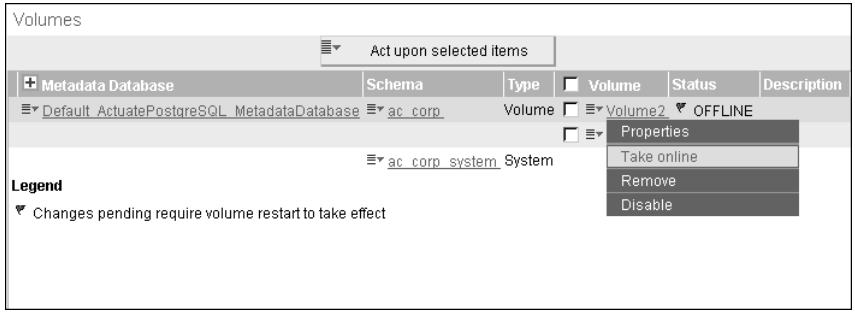

Figure 2-20 Taking a new volume online

6 In Volumes, check that the status of the new volume changes to ONLINE, as shown in Figure 2-21.

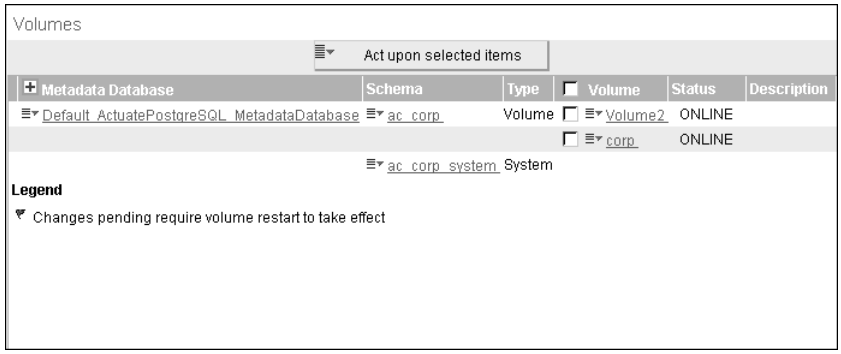

Figure 2-21 Confirming that the volume is online

If the volume does not go online, check for insufficient free disk space for the partition and consider configuring the free space threshold.

Editing Encyclopedia volume properties

On Volumes, an administrator can edit Encyclopedia volume properties in Volumes—Properties—General.

How to edit Encyclopedia volume properties

1 On Volumes, point to the icon next to a volume name and choose Properties, as shown in Figure 2-22.

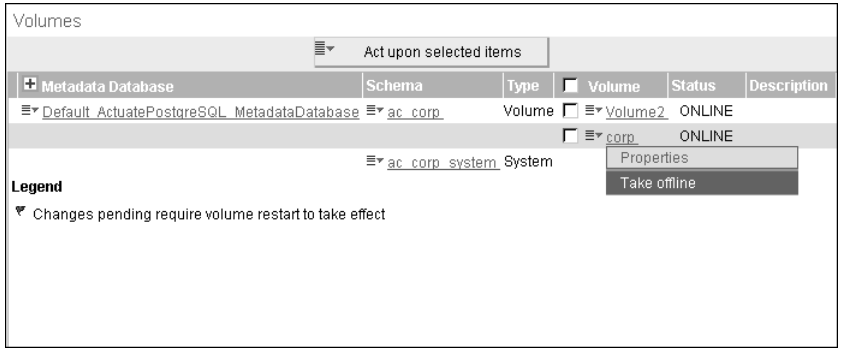

Figure 2-22 Choosing Encyclopedia volume properties

2 On Properties, the available properties are the same as when you add a new volume, with the exception that you can set Advanced properties, as shown in Figure 2-23.

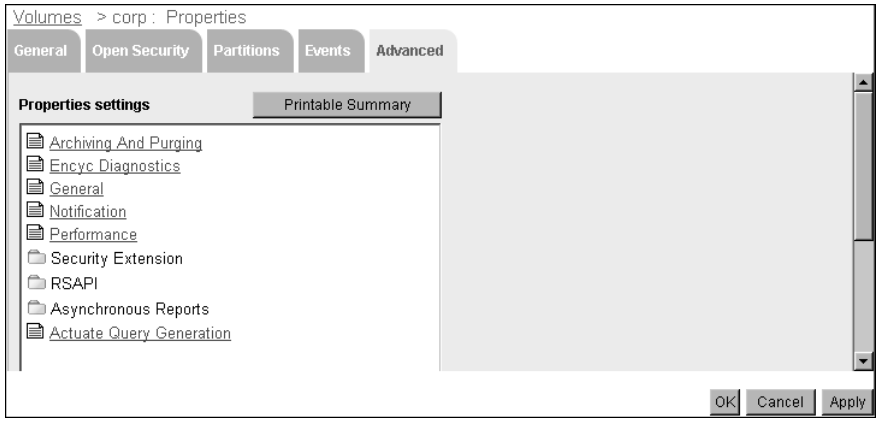

Figure 2-23 Viewing advanced volume properties

Table 2-1 lists some of the properties that appear on Volumes—Properties—General.

Table 2-1 New Encyclopedia volume general properties

Property	Description
Volume name	Name of the Encyclopedia volume.
Description	Optional description of the Encyclopedia volume.
Schedule for purging notices	Time or times at which iServer deletes job completion notices. The value format is a semicolon-separated list of times, in ascending order. Use a 24-hour format. For example: `03:15;16:15`
Schedule for purging deleted files	Time or times at which iServer removes deleted files from the system. The value format is a semicolon-separated list of times, in ascending order. Use a 24-hour format. For example: `03:15;16:15`
Primary partition	Name of the primary partition for the Encyclopedia volume.
Min Free Space for the primary partition	Minimum amount of free disk space that the primary partition requires. If the free space falls below the specified minimum, iServer does not create the file.
Use archive service	Specify the command-line string iServer runs to start the SOAP-based Inline Archive Driver application.

(continues)

Table 2-1 New Encyclopedia volume general properties (continued)

Property	Description
Information Console URL prefix	URL prefix, for example http://sales:8900/iportal, to add to the hypertext link in an e-mail notification sent to Information Console users. Use also to specify a hyperlink, such as a drill-through hyperlink, containing the Information Console context string in a generated PDF document. For more information on customizing e-mail notifications, see Chapter 4, "Configuring e-mail notification," later in this book. For more information on configuring Information Console URL prefix to specify the context root for a hyperlink in a PDF document, see "Specifying a hyperlink in a PDF document," later in this chapter.

The following examples describe how to configure various general volume properties.

Scheduling purging of job completion notices

The administrator can disable purging of job completion notices, but this action can cause too many notices to build up. The best practice is to configure iServer to purge job completion notices during times of light use and at least one hour before or after autoarchiving.

Use 24-hour clock time points, separated by a semicolon. For example, the default value: 2:15 runs a purge task at 2:15 A.M. every day. A blank disables the job and notice purging.

Scheduling purging of deleted files

The physical file for an Encyclopedia volume file appearing in Management Console resides in the Encyclopedia volume file folder as a .dat file. For a file a user deletes from a volume in Management Console, iServer purges the corresponding .dat file from the volume file folder at the time or times the Schedule for purging deleted files property specifies. Specify multiple purge times using 24-hour clock time points, separated by a semicolon.

Configuring an Encyclopedia volume partition

View and modify the partitions assigned to an Encyclopedia volume in Volumes—Properties—Partitions.

Assigning a partition to an Encyclopedia volume partition

To associate a partition with a volume, assign the partition to the volume, as shown in Figure 2-24.

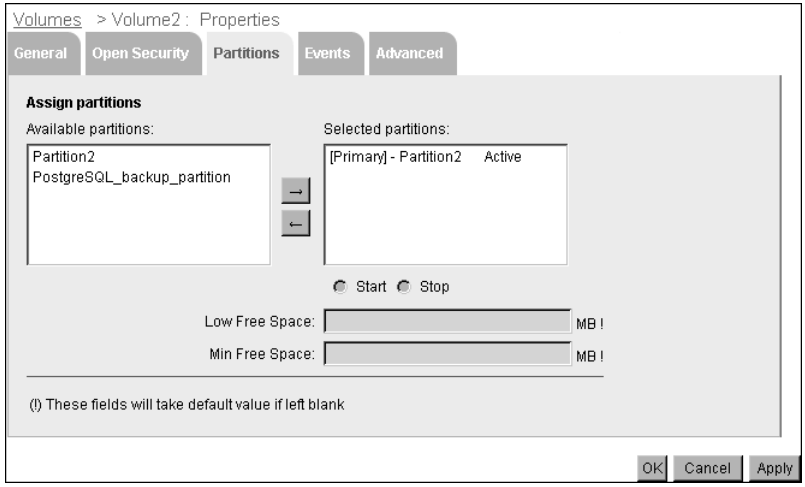

Figure 2-24 Assigning a partition to an Encyclopedia volume

How to assign a partition to an Encyclopedia volume

1. In Volumes—Properties—Partitions, use the left and right arrows to move iServer partition names between Available partitions and Selected partitions.

   Available partitions lists the partitions that are available for use by an Encyclopedia volume. Selected partitions lists the partitions iServer uses to store Encyclopedia volume files.

2. Use the up and down arrows to change the order of the partitions in Selected partitions.

   Primary indicates which of the selected partitions is the primary partition. Active indicates which of the selected partitions are actively in use.

3. Select a partition in Selected partitions and choose Start or Stop to change the status of a partition.

   Choosing Start activates an inactive partition. You must activate a partition before using it for an Encyclopedia volume. Choosing Stop changes the status of a partition to Phasing out or Not in use. When a partition is stopped, you cannot use it.

4. Select a partition in Selected partitions and specify free disk space properties.

Configuring free space on an Encyclopedia volume partition

To prevent possible Encyclopedia volume corruption and operational problems with the machine due to lack of free disk space, iServer shuts down the volume when the disk space available to the partition for the volume falls below a minimum, 128MB by default. The administrator can change the minimum by configuring Min Free Space. Increase Min Free space if iServer, or any other application making use of the same physical disk, consumes disk space at a rapid rate, or if fragmentation occurs.

In Volume—Properties—General, the administrator can configure the minimum free disk space for a partition by typing a new value in Min Free Space for the primary partition. For example, increase the value to 256, as shown in Figure 2-25.

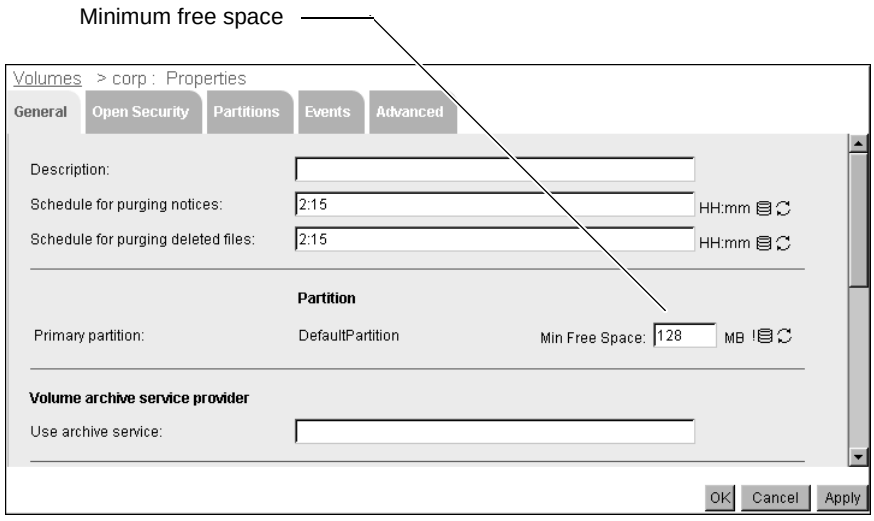

Figure 2-25 Increasing the minimum free space for a partition

Table 2-2 lists these additional properties that appear on Volumes—Properties—Partitions.

Table 2-2 Volume partition properties

Property	Value
Low Free Space	Amount of free space, in megabytes (MB), below which iServer displays a warning message. If a user tries to create a file in a partition with less than the specified low amount of free disk space, file creation succeeds, and a warning message is displayed. The default value is 512 MB.

Table 2-2 Volume partition properties

Property	Value
Min Free Space	Minimum amount, in megabytes (MB), of free space that the Encyclopedia volume partition must maintain. If a user tries to create a file that would put the partition under its minimum free space limit, iServer does not create the file. The default value is 128 MB.

Moving an Encyclopedia volume to another partition

Volumes—Properties—General lists the primary partition for the volume. Volumes—Properties—Partitions, shown in Figure 2-26, lists all the partitions available to the Encyclopedia volume for storing volume files.

The volume uses the primary partition for volume administrative information and to store volume files. To move an Encyclopedia volume to another partition, the administrator must copy the Encyclopedia volume files to the new partition, and then update the old partition path to the new location.

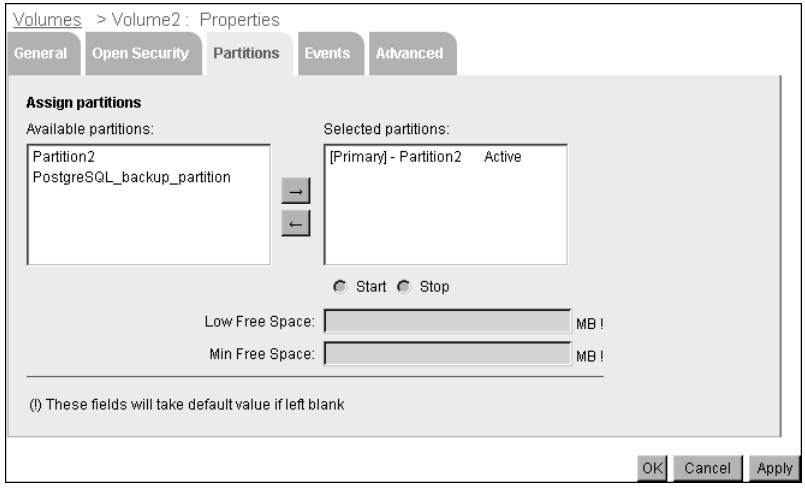

Figure 2-26 Moving an Encyclopedia volume

How to move an Encyclopedia volume

1. Set up one or more partitions.
2. Put the volume offline.
3. For each partition, specify the partition location on Partitions—Template Settings.

4 Copy the Encyclopedia volume files from the location specified by the old partition path to the location specified by the new partition path. In a cluster, all machines must be able to access the path specified by the partition.

5 Take the volume online.

Supporting application-level partitioning

In a cluster, you can use Encyclopedia volumes to support application-level partitioning by configuring different applications to use different Encyclopedia volumes. In some cases, such as when an application service provider hosts services, a separate physical structure is part of the logical design. For example, business needs can require the separation of billing information from sales information. To use multiple Encyclopedia volumes without changing the logical design of the application, an application designer can use links in the application folder structure to redirect users to the appropriate Encyclopedia volume.

To make two Encyclopedia volumes work as a single unit, you need a single source of user information, such as user names, passwords, and role membership. The application designer can externalize Encyclopedia volume user information using the Actuate Open Security feature and centralize the user information in an external security source. For example, iServer can store the Encyclopedia volume user information in an LDAP server. Multiple Encyclopedia volumes can use the information from the LDAP server.

Specifying a hyperlink in a PDF document

The context string in which a job runs is not available in the background job that generates a PDF document. To specify a hyperlink, such as a drill-through hyperlink, containing the appropriate Information Console context string in a PDF document, the administrator must specify the Information Console URL prefix by performing one of the following tasks:

- In Configuration Console, update the setting for Information Console URL prefix in Volumes➤Properties
- In acserverconfig.xml, manually edit the setting for IPortalURLPrefix

How to update the setting for Information Console URL prefix

1 Open Configuration Console. Choose Advanced View.

2 In Advanced View, navigate to Volumes➤Properties.

3 In Information Console URL prefix, type the context string, as shown in Figure 2-27. For example, type:

```
http://urup:8900/iportal
```

4 Choose OK.

Configuration Console updates the setting for the system variable, IPortalURLPrefix, in acserverconfig.xml in AC_DATA_HOME/config.

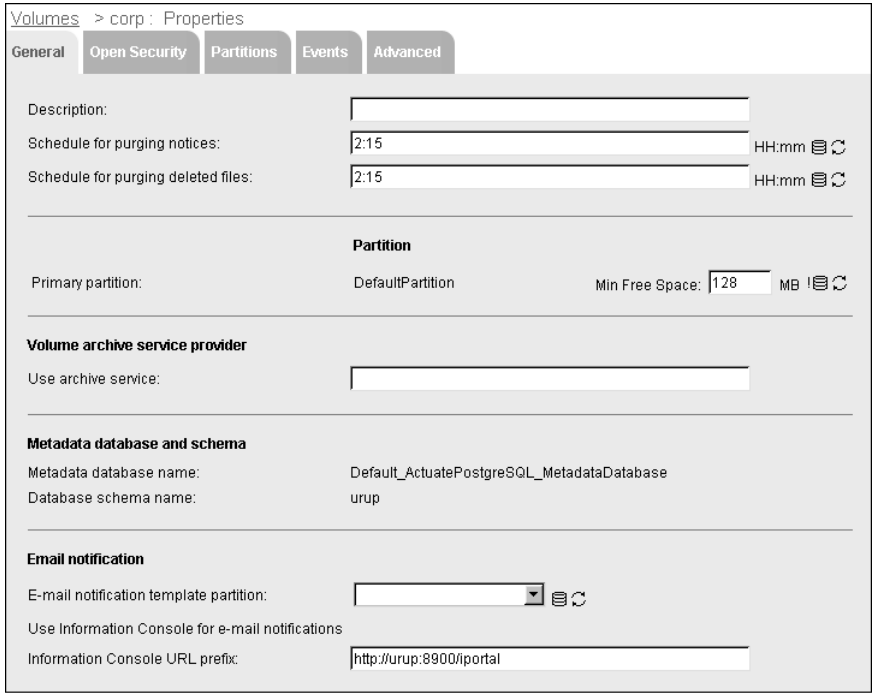

Figure 2-27 Specifying Information Console URL prefix

How to manually edit the setting for IPortalURLPrefix

1 In Windows Explorer, navigate to AC_DATA_HOME/config.

2 Open acserverconfig.xml in a text editor.

3 In acserverconfig.xml, edit the <Volume> setting, IPortalURLprefix, to contain the appropriate Information Console context string, as shown in Listing 2-1. For example, type:

```
IPortalURLPrefix="http://urup:8900/iportal"
```

4 Save the file.

Listing 2-1 shows an example of an edited setting for IPortalURLprefix in acserverconfig.xml.

Listing 2-1 Creating a .bat file to point iServer to portserv.exejj

```
<Volumes>
   <Volume
      Name="urup"
      EmailURLType="IPortal"
      RSSESOAPPort="8900"
      PrimaryServer="$$ServerName$$"
      IPortalURLPrefix="http://urup:8900/iportal"
```

It is not necessary to restart iServer after changing the setting for IPortalURLPrefix. In AC_SERVER_HOME/etc, acmetadescription.xml specifies that any change to IPortalURLPrefix takes effect immediately, as shown in the following code excerpt:

```
<Variable
   Name="IPortalURLPrefix"
   Type="String"
   Access="Public"
   Required="False"
   DisplayName="IPortal URL Prefix"
   TakesEffect="Immediate"/>
```

Configuring events

In Volumes—Properties—Events, you set values used with event schedules, enable custom event schedules, and specify the web service that a custom event uses. Figure 2-28 shows Volumes—Properties—Events.

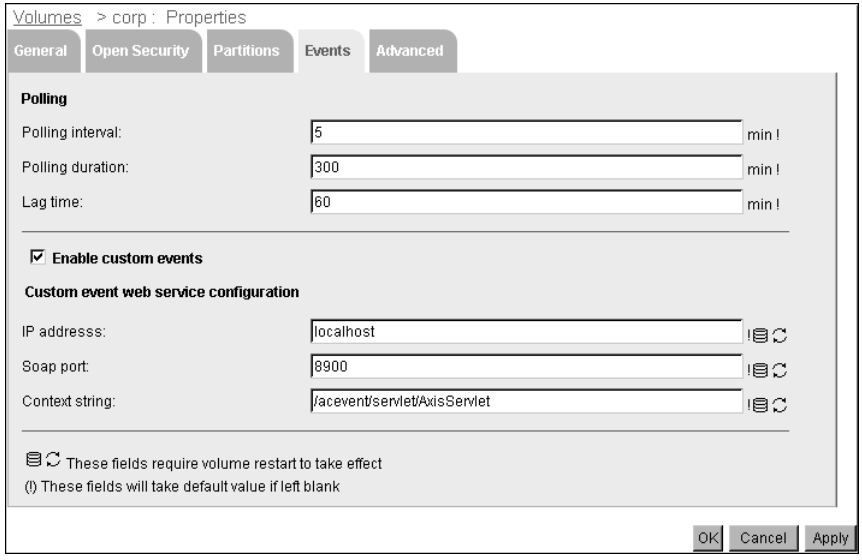

Figure 2-28 Specifying event-based job configuration values

When you deploy a BIRT design to an Encyclopedia volume, you must provide iServer with access to Java classes that the design uses. You package these classes as JAR files that a BIRT iServer Java factory process accesses. There are two ways to deploy Java classes:

- Deploy the JAR files to the Encyclopedia volume

  Supports creating specific implementations for each volume in iServer. This method of deployment requires packaging the Java classes as a JAR file and attaching the JAR file as a resource to the design file. You treat a JAR file as a resource in the same way as a library or image. Using this method, you publish the JAR file to iServer every time you make a change in the Java classes.

- Deploy the JAR files to the following iServer subdirectory:

  AC_SERVER_HOME\resources

  This method uses the same implementation for all volumes in BIRT iServer. You do not have to add the JAR file to the design Resource property. Deploying JAR files to an iServer /resources folder has the following disadvantages:

  - You must restart iServer after deploying the JAR file.
  - The JAR file is shared across all volumes. This can cause conflicts if you need to have different implementations for different volumes.

Table 2-3 describes the properties that appear on Volumes—Properties—Events.

Table 2-3 Event-based job configuration properties

Property	Value
Polling interval	The frequency in minutes that iServer checks for a system event.
Polling duration	The duration in minutes that iServer checks for an event. If the event does not occur within the allotted time, iServer marks it as expired. A user can customize this value when creating an event-driven schedule. This value applies to all types of system events.
Lag time	The minutes that iServer scans for completed jobs to determine if an event occurred. For example, if you submit an event-based schedule with the default event lag time, iServer checks the status of jobs for the previous 60 minutes. If the event occurred within the previous 60 minutes, it sets the event status to satisfied.

(continues)

Table 2-3 Event-based job configuration properties (continued)

Property	Value
Enable custom events	A flag that enables iServer custom event processing for a scheduled job. If the value is true, the service is enabled. If you change the value to false, all the existing instances of scheduled jobs using the custom event fail. This configuration value also affects the EnableEventService property value in the Actuate IDAPI GetVolumeProperties response.
IP address	The server name or IP address where the custom event service resides. The default value is localhost.
Soap port	The number of a valid, used port for the custom event service. iServer uses an application container to host web services applications.
Context string	The context string of the request URL for sending a message to the custom event service. Default value is /acevent/servlet/AxisServlet.

Configuring advanced volume properties

Volumes—Properties—Advanced contains a variety of miscellaneous volume property settings that an administrator can use to make performance-tuning adjustments for a mix of features. Figure 2-29 shows the Advanced properties list.

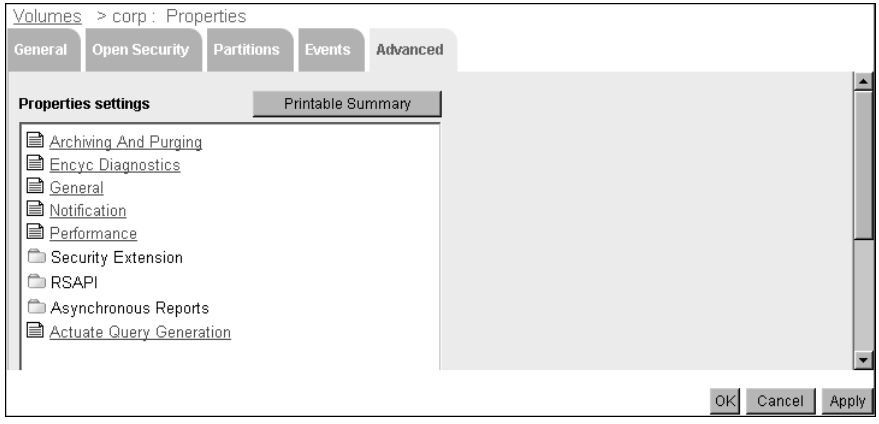

Figure 2-29 Specifying advanced Encyclopedia volume property settings

Printing a summary of advanced volume properties

In Volumes—Properties—Advanced, you can view or print a summary of Encyclopedia volume properties and their values. First, select the category

properties, then choose Printable Summary to print the information. You can select one of following categories of properties:

- Archiving And Purging
- Encyc Diagnostics
- General
- Notification
- Performance
- Security Extension
- RSAPI
- Asynchronous Reports
- Actuate Query Generation

Retrying failed asynchronous jobs

In Volumes—Properties—Advanced—Asynchronous Reports—Asynchronous Job Retries, specify how to retry running a scheduled job or query, as shown in Figure 2-30.

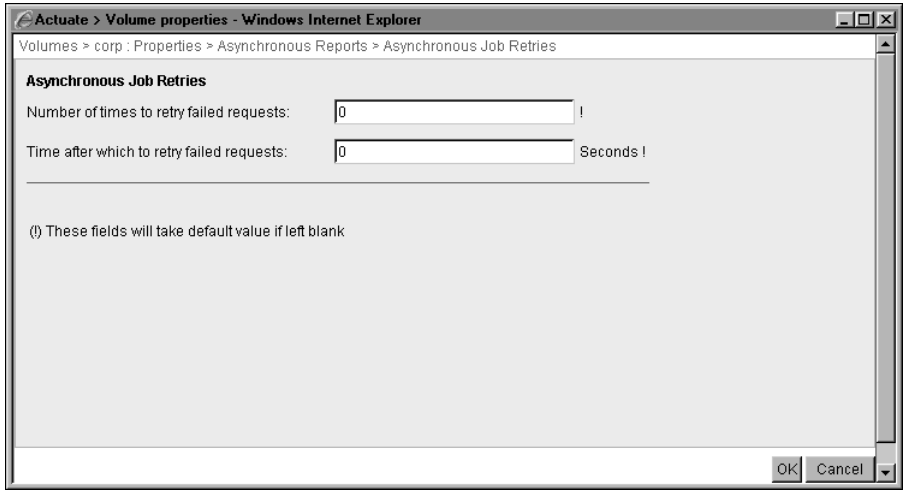

Figure 2-30 Specifying values for retrying failed asynchronous jobs

If the job retry options are set to retry a job if it fails, the job remains active if the node the job is running on fails. For example, if the node crashes, iServer tries to run the job again when the node restarts.

Configuring RSAPI

Actuate supports applications that call the Report Server API (RSAPI) from Actuate Basic designs. If a single machine manages multiple Encyclopedia volumes, only one volume has the RSAPI enabled. By default, the volume with RSAPI enabled is the first volume added to the machine. For example, if iServer manages an Encyclopedia volume named sales, and you add an Encyclopedia volume named production, RSAPI applications can access only the volume named sales. RSAPI applications cannot access the volume named production.

To enable RSAPI on a different Encyclopedia volume, set the Volume used by the Requester and Report Server API parameter. In Volumes—Properties—Advanced—RSAPI—Fetch Limits, you can limit the number of items that an Actuate Report Server API (RSAPI) request returns, as shown in Figure 2-31.

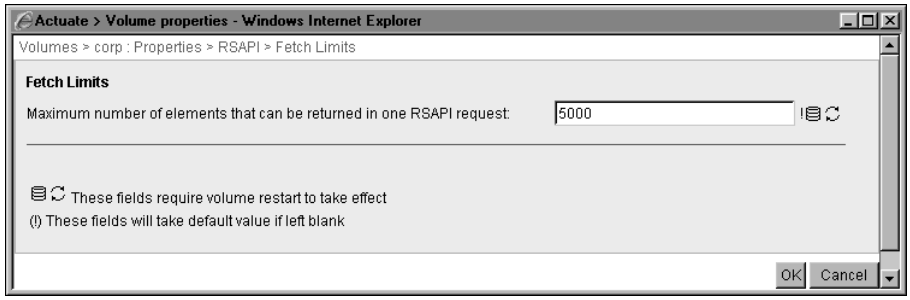

Figure 2-31 Specifying the fetch limit for an RSAPI request return

In Volumes—Properties—Advanced—RSAPI—Fetch Limits, you set Maximum number of elements that can be returned in one RSAPI request. This parameter controls the maximum number of items that iServer returns in a list. Specify a value for this parameter if you use RSAPI applications. iServer truncates the list if a RSAPI call returns a list that contains more elements than the value of this parameter. The default value is 5000. Actuate recommends that you use the default value. The minimum value is 0.

Actuate recommends using incremental fetch features to retrieve large lists. Increase the value to the size of the longest list that iServer sends if you cannot use incremental fetch features.

When iServer encounters performance problems, such as excessive memory usage by a RSAPI application that frequently retrieves large lists, decrease the value of this parameter to improve performance. Decreasing the value of this parameter can cause the RSAPI application to function incorrectly.

Changing the status of an Encyclopedia volume

The status of an encyclopedia volume can be disabled, enabled, offline, or online. You change the status of a volume on Volumes, in Advanced view of Configuration Console.

Taking an Encyclopedia volume offline

To disable or remove a volume, you must first take it offline.

How to take an Encyclopedia volume offline

1. From Advanced view of Configuration Console, choose Volumes.
2. In Volumes, point to the arrow next to a volume name and choose Take offline, as shown in Figure 2-32.

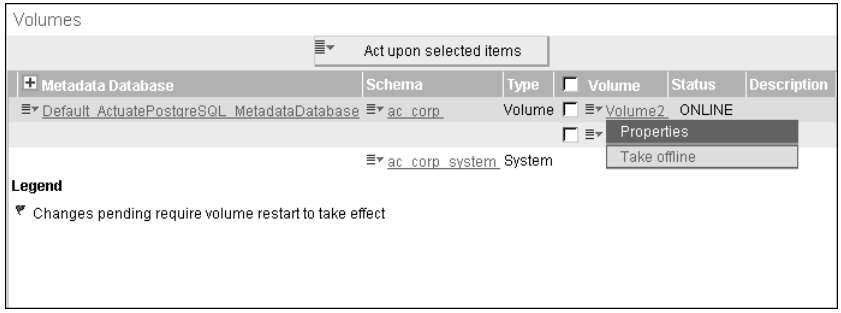

Figure 2-32 Taking a volume online or taking a volume offline

3. On Volume Offline Grace Period, choose OK, as shown in Figure 2-33. You can adjust the time a volume takes to go offline in Grace period.

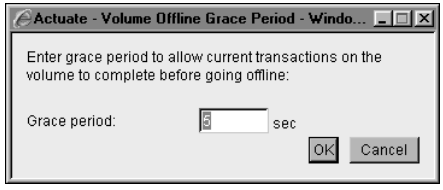

Figure 2-33 Choosing OK to take the volume offline

Disabling an Encyclopedia volume

The difference between an offline volume and a disabled volume is that iServer takes an offline volume online when you start the iServer system or the iServer service. A disabled volume does not come online when you start the iServer system or service. The volume remains disabled.

How to disable a volume

1. In Volumes, point to the icon next to a volume name and choose Take offline.
2. Point to the icon next to the offline volume name and choose Disable, as shown in Figure 2-34. Confirm that you want to disable the volume.

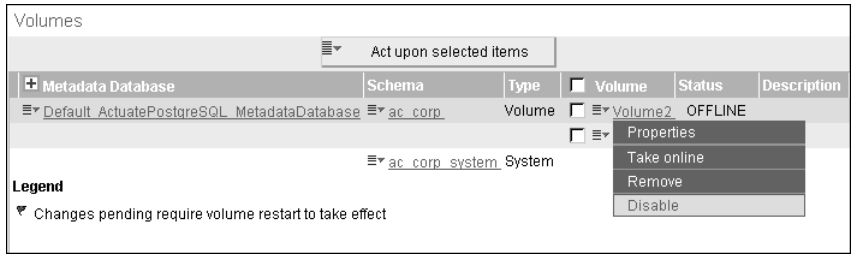

Figure 2-34 Disabling a volume

The volume status changes to DISABLED.

Enabling an Encyclopedia volume

Before you can take a disabled volume online, you must enable it. To enable a volume, point to the icon next to a disabled volume name and choose Enable, as shown in Figure 2-35.

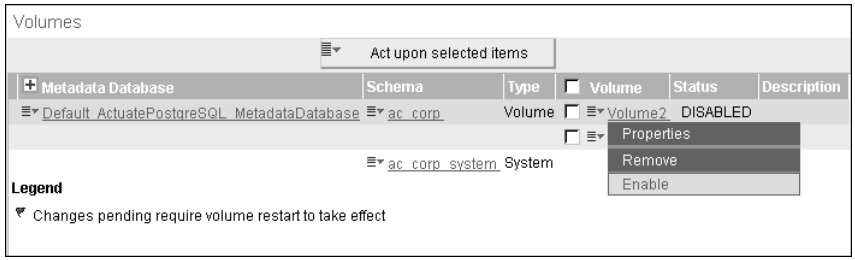

Figure 2-35 Enabling a volume

The volume status changes to OFFLINE.

Taking an Encyclopedia volume online

When the status of a volume is OFFLINE, you can take the volume online. To be available to users, an Encyclopedia volume must be online. To take a volume online, on Volumes, point to the icon next to the offline volume name and choose Take online, as shown in Figure 2-36.

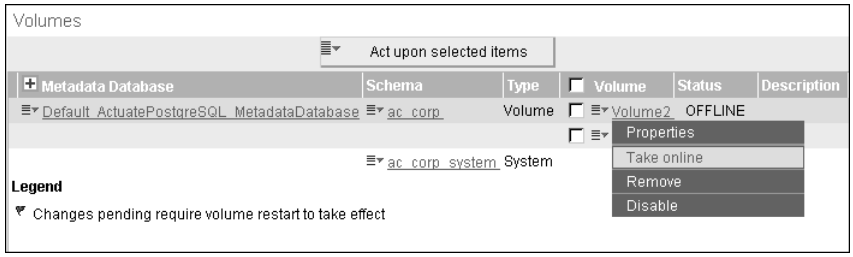

Figure 2-36 Taking a volume online

Removing an Encyclopedia volume

When you remove an Encyclopedia volume from iServer, the following changes occur:

- Encyclopedia volume users no longer see the volume in the list of volumes available to iServer or a cluster.
- Encyclopedia volume users cannot log in to the Encyclopedia volume.
- iServer rejects requests to access the content of the Encyclopedia volume.
- The volume no longer appears in the configuration file.
- iServer changes an Encyclopedia volume's resource group volume assignment to All volumes and disables the resource group.

Deleting an Encyclopedia volume does not delete the partitions assigned to the volume.

How to remove an Encyclopedia volume from iServer

To completely remove an Encyclopedia volume from iServer, perform the following tasks:

1. Before removing an Encyclopedia volume, note which partitions the volume uses.
    1. From Advanced view of Configuration Console, choose Volumes.
    2. On Volumes, point to the icon next to a volume name and choose Properties.
    3. In Volumes—Properties, choose Partitions.
    4. In Partitions, note the list of partitions that appears in Selected partitions. From the side menu, choose Volumes.
2. Point to the icon next to a volume name and choose Take offline.

    In Volume Offline Grace Period, choose OK.

    The volume's status changes to OFFLINE.
3. On Volumes, point to the icon next to a volume name and choose Remove, as shown in Figure 2-37.

    A confirmation dialog box appears.
4. To delete the Encyclopedia volume from the machine, choose OK.

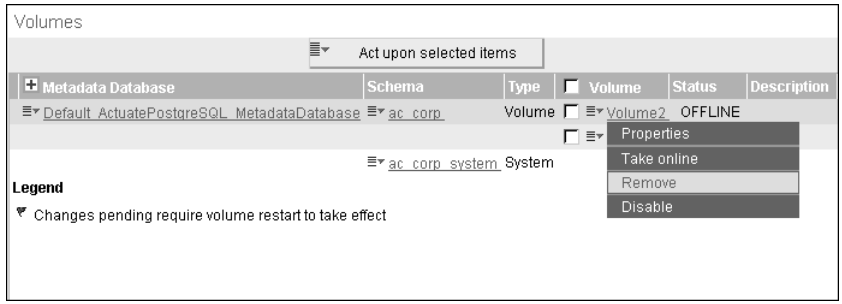

Figure 2-37 Removing an Encyclopedia volume

Renaming the default Encyclopedia volume

The login page of Management Console lists the default Encyclopedia volume. In System—Properties—General, change the System default volume, as shown in Figure 2-38.

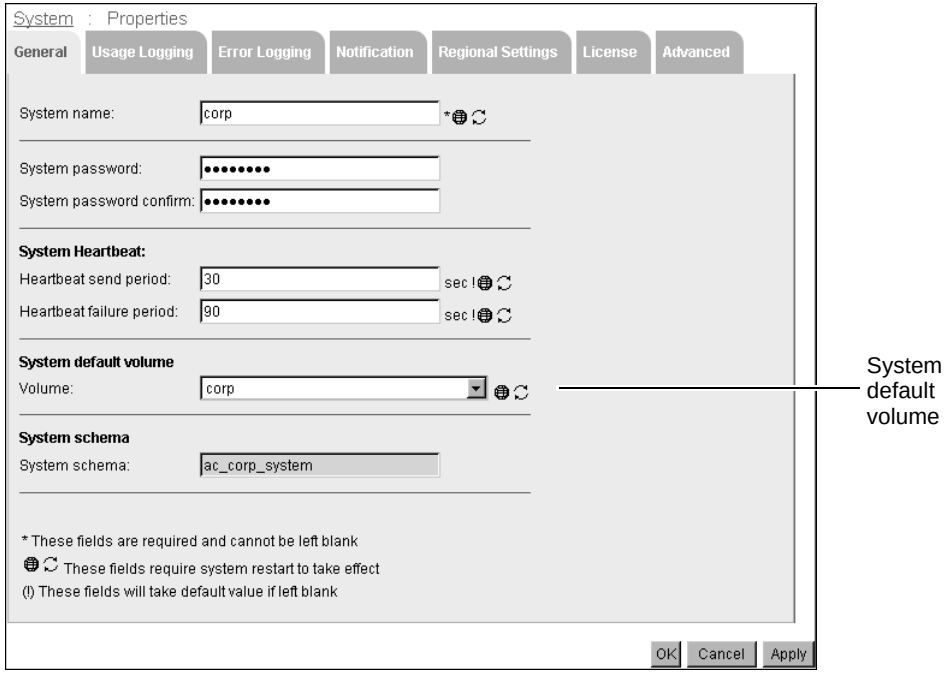

Figure 2-38 Specifying the default Encyclopedia volume

Removing a partition

Before you remove a partition, you must check the list of partitions in Configuration Console to see which partitions are available and which are already assigned to Encyclopedia volumes. Assign a different partition to the Encyclopedia volume to which the partition is assigned, or back up the Encyclopedia volume.

If the partition is a secondary partition, you delete the partition by changing its configuration. iServer starts a background process of moving files from one partition to another. The operation fails if the other partitions drop below their minimum free space limits. When the process finishes, the state of the secondary partition changes to inactive, and you can remove it.

How to remove a partition

1. From Advanced view of Configuration Console, choose Volumes.
2. On Volumes, point to the arrow next to the Encyclopedia volume that uses the partition and choose Properties.
3. In Volumes—Properties, choose Partitions.
4. On Volumes—Properties—Partitions, in Selected partitions, select the partition to remove. Select Stop. Choose OK.

   iServer moves data from the partition to other available volume partitions. If there is not enough room to store the data from the partition being deleted, a failure message appears.
5. After iServer finishes phasing out the partition, repeat steps 2 and 3.
6. On Volumes—Properties—Partitions, in Selected partitions, select the partition to remove. Choose the left arrow to move the partition from Selected partitions to Available partitions.
7. Choose Partitions from the side menu.
8. In Partitions, confirm that the partition is not assigned to an Encyclopedia volume and that it is unused. Then, point to the arrow next to the partition name to display the drop-down list for the partition.
9. Choose Delete. To confirm the deletion, choose OK to, as shown in Figure 2-39.

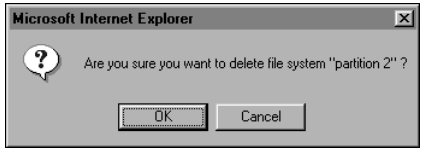

Figure 2-39 Deleting a partition

Chapter

3

Using diagnostic, usage, and error logging

This chapter contains the following topics:

- Configuring diagnostic logging
- Configuring usage and error logging

Configuring diagnostic logging

iServer performs various types of diagnostic logging with configurable levels of detail. An administrator can experiment with the available levels to collect information on iServer system to detect problems and improve performance. Setting log files to higher levels of information collection increases disk space usage and can decrease iServer performance.

The administrator configures diagnostic logging for iServer process by choosing Change in Server Configuration Templates—Settings, as shown in Figure 3-1.

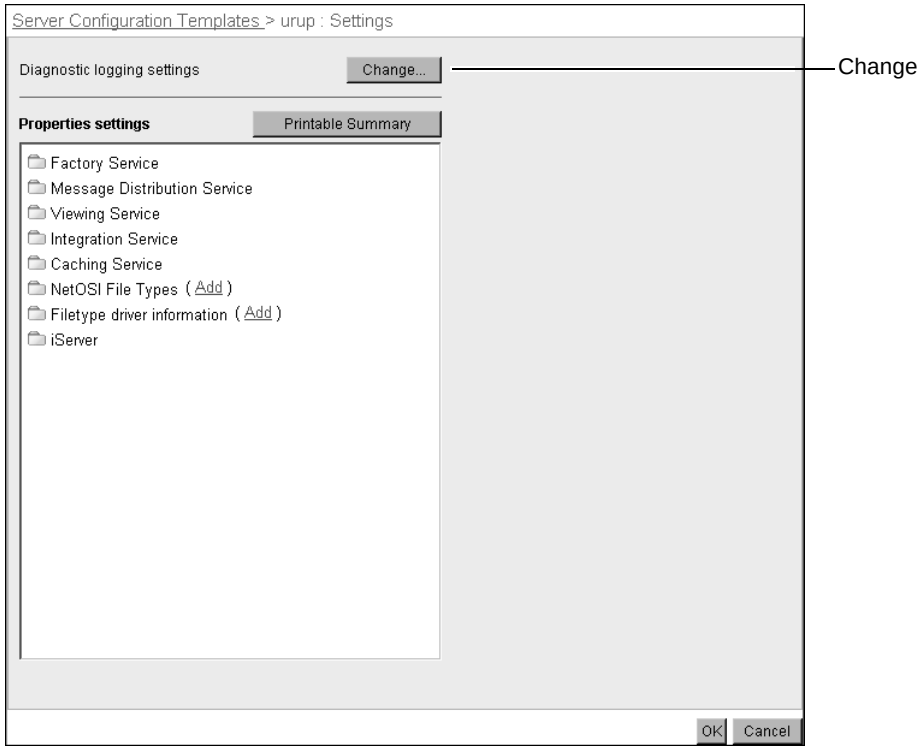

Figure 3-1 Changing diagnostic logging settings

Server Configuration Templates—Settings allows the administrator to specify the following diagnostic logging settings, as shown in Figure 3-2:

- Enable logging
  Enable or disable diagnostic logging.
- Log level
  Level of detail in the log.

- Log directory
  Location of the log.
- Log size
  Maximum size of the log before iServer starts overwriting the log.
- Number of log files
  Maximum number of logs that iServer creates.

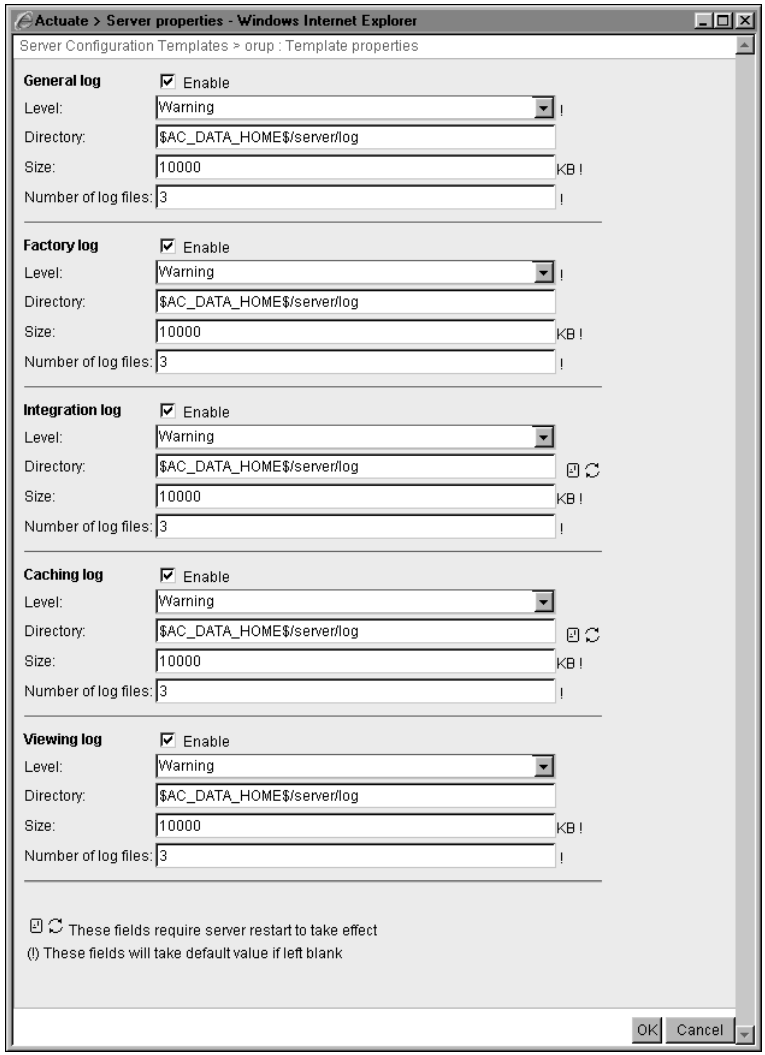

Figure 3-2 Configuring diagnostic logging

By default, diagnostic logging is enabled for all iServer processes. The administrator can disable diagnostic logging for a process by deselecting Enable.

Changing the default log file location in acserverconfig.xml

In Release 11 Service Pack 4, iServer writes log entries by default to AC_DATA_HOME\server\log. The log location is no longer specified in acserverconfig.xml.

To specify a different location or manually update a previously edited log setting, the administrator must modify acserverconfig.xml to contain a LogDirectory attribute that points to a non-default location by performing the following tasks:

1. Open AC_DATA_HOME\config\11SP4\acserverconfig.xml in a text editor.
2. Insert or edit a LogDirectory attribute that specifies the path to a different log folder.

For example, in Listing 3-1, an inserted LogDirectory attribute specifies the non-default log location, C:\Actuate\iServer\log, for the following properties:

- Template
- Reporting Service
- Viewing Service
- Integration Service
- Caching Service

Listing 3-1 acserverconfig.xml edited log file location settings

```
<Template
   Name="urup"
   PMDPort="8100"
   ActuateBuild="DEV120208"
   LogDirectory="C:\Actuate\iServer\log"
   ActuateVersion="11 Service Pack 4 Development"
   AppContainerPort="8900"
   ServerSOAPPortBase="9700"
   RequesterRSAPIVolume="urup">
   <ReportingService
      LogDirectory="C:\Actuate\iServer\log"
      EnableGenerationService="true"/>
   <SOAPDispatchService
      EnableRequestService="true"
      SOAPDispatchSOAPPort="8000"/>
   <ViewingService
      EnableRender="false"
```

```
      LogDirectory="C:\Actuate\iServer\log"
      EnableViewingService="true"
      MSAnalysisTunnelPort="9021"/>
   <IntegrationService
      LogDirectory="C:\Actuate\iServer\log"
      EnableIntegrationService="true"/>
   <CachingService
      LogDirectory="C:\Actuate\iServer\log"
      EnableCachingService="true"/>
   ...
</Template>
```

Configuring additional iServer diagnostic logging properties

In Server Configuration Templates—Settings, the administrator can also configure the following additional iServer property settings, as shown in Figure 3-3:

- iServer Diagnostic Logging
- Logging for Encyclopedia Database Queries

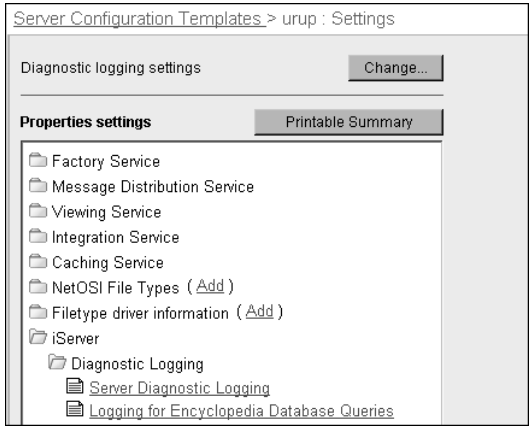

Figure 3-3 Choosing iServer Diagnostic Logging

About iServer Diagnostic Logging

The following parameters define the levels of iServer Diagnostic Logging detail you can specify.

How to configure additional diagnostic logging settings

1 In Server Configuration Templates, choose the name of a server template, as shown in Figure 3-4.

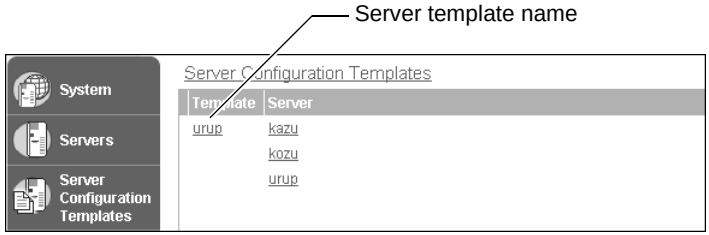

Figure 3-4 Choosing a server template name

2 On Server Configuration Templates—Settings, expand iServer, expand Diagnostic Logging, then choose iServer Diagnostic Logging, as shown in Figure 3-3.

3 On Server Diagnostic Logging, accept the default value for each setting, as shown in Figure 3-5, or alternatively, specify a new value.

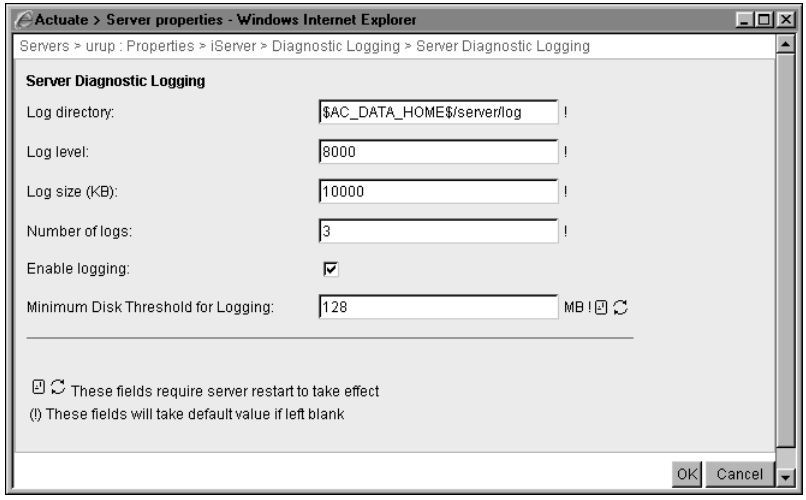

Figure 3-5 Specifying Server Diagnostic Logging property settings

Choose OK.

4 If you change Minimum Disk Threshold for Logging, restart iServer.

About Logging for Encyclopedia Database Queries

In Logging for Encyclopedia Database Queries, the administrator can specify the level of diagnostic log detail that iServer provides when an SQL statement fails to execute successfully. The following parameters define the levels of logging detail you can specify:

- LogSQLQuery
  iServer logs the query sent to the database.

- LogQueryPrepareTime
  iServer logs the time spent doing a prepare against the database.
- LogQueryExecutionTime
  iServer logs the time spent executing the query against the database.

How to configure logging for Encyclopedia database queries

1 In Server Configuration Templates, choose the name of a server template, as shown in Figure 3-6.

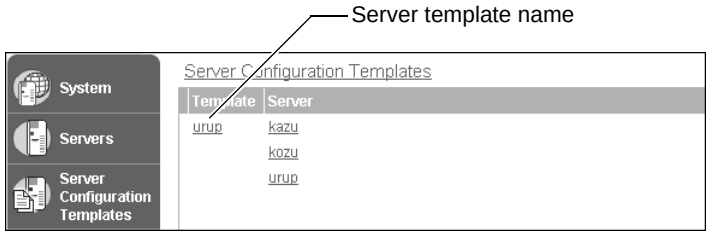

Figure 3-6 Choosing a server template name

2 On Server Configuration Templates—Settings, expand iServer, Diagnostic Logging, then choose Logging for Encyclopedia Database Queries, as shown in Figure 3-7.

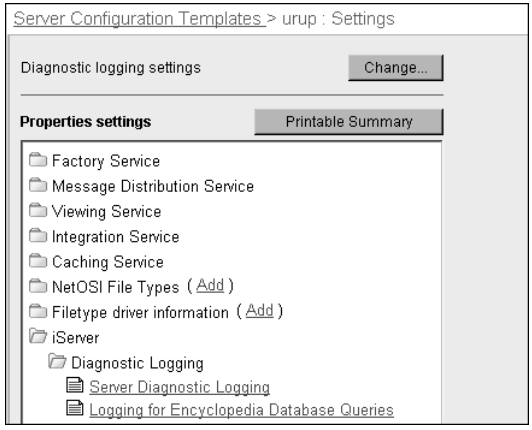

Figure 3-7 Choosing Logging for Encyclopedia Database Queries

3 In Logging for Encyclopedia Database Queries, select an option from Select a specific option and move it to Diagnostic Logging for Encyclopedia Database Queries, using the right arrow, as shown in Figure 3-8.

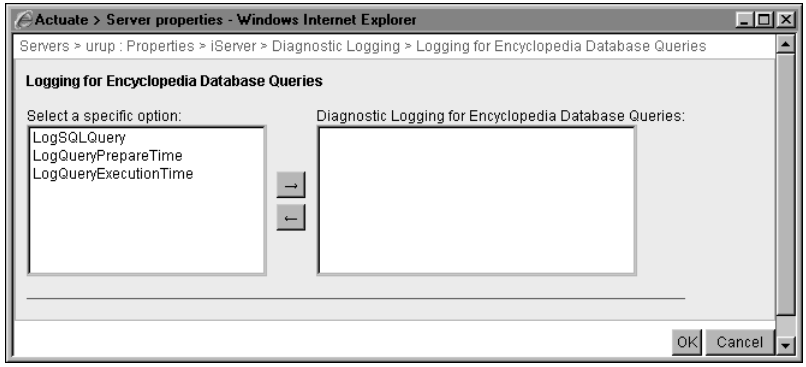

Figure 3-8 Choosing a logging level

Choose OK.

For more information about diagnostic logging properties, see "Configuring diagnostic logging" in Chapter 1, "Performing basic configuration tasks."

The environment variable AC_SERVER_FILE_CREATION_MASK on UNIX and Linux, and registry key on Windows, specify the permissions for log files and directories in /UsageErrorLogs and /iServer/log. The default value in UNIX and Linux is owner read-write (066 UMASK), and in Windows is 0 (read-only).

Configuring usage and error logging

iServer can capture usage monitoring and error information in log files. To understand resource usage and troubleshoot problems, use usage and error logging.

In the Advanced view, choose System. Choose Properties, then choose Usage Logging. Figure 3-9 shows Usage Logging.

The following Actuate usage and error logging applications that ship with iServer record usage activity and error information in a log file:

- Usage Logging Extension
  Writes the usage information to a comma-delimited (.csv) file, usage_log.csv
- Error Logging Extension
  Writes the usage information to a comma-delimited (.csv) file, error_log.csv

The usage logging and error logging applications are open framework applications. These applications are DLLs in Windows and shared libraries in UNIX or Linux. By default, the usage and error logging extensions reside in the following location:

```
AC_SERVER_HOME/bin
```

iServer creates a primary log directory that contains the usage and error log records for the default volume in the following location:

```
AC_SERVER_HOME/UsageErrorLogs/primary
```

iServer creates secondary log directories for additional volumes as required in the following location:

```
AC_SERVER_HOME/UsageErrorLogs/secondary_$VOLUMENAME
```

The directories for usage and error log files are not configurable.

In an iServer cluster, the asynchronous design execution log entry is in the log of the Encyclopedia process node. For a synchronous or transient document generation, the log entry is in the log file of the node that runs the design. Before you configure error logging for a cluster, ensure that each node in a cluster has the same logging applications installed.

A developer can customize the way the DLL or shared library handles the usage and error log information. iServer Integration Technology provides a reference implementation for this purpose.

iServer supports consolidating log information into a database. iServer Integration Technology provides a reference implementation for the log consolidator application. For more information, see *Using BIRT iServer Integration Technology*.

Configuring usage logging

Configure usage log settings for the following user operations:

- Document viewing
  A viewing log records document-viewing events that use the View process
- Document printing
  A printing log records document-printing events
- Document generation
  A Factory log records document-generation events
- Document deletion
  A deletion log records internal deletion operations, such as deletions that occur from archiving volume files
- Admin
  An admin log records Encyclopedia volume management user operations, such as:
  - Managing users, roles, groups, and channels
  - Uploading and deleting files

- Changing file and folder properties

- Data integration
  A Data Integration log records AIS events, such as information object jobs

How to configure usage logging

To configure usage logging, perform the following tasks:

1. Log in to Configuration Console and choose Advanced view.
2. In System—Status, choose Properties.

   System—Properties—General appears.
3. Choose Usage Logging.

   System—Properties—Usage Logging appears, as shown in Figure 3-9.

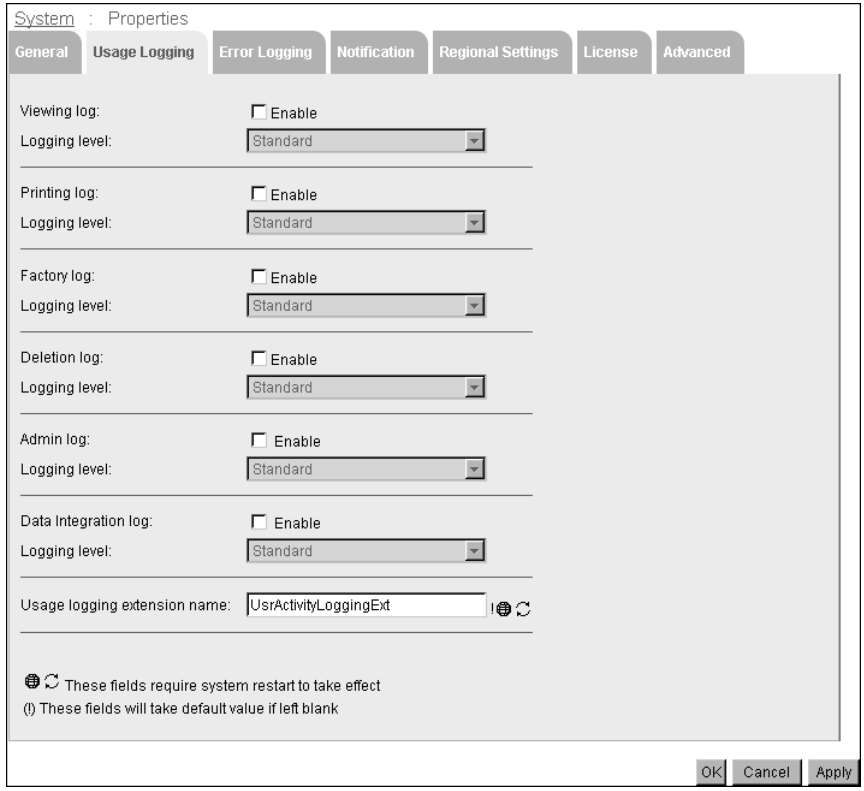

Figure 3-9 Configuring usage logging

4. On Usage Logging, select the usage logging information you want to capture from the following list of logging options:

- Viewing
- Printing
- Factory
- Deletion
- Admin
- Data Integration

1. Select Enable to activate the logging option.
2. Select Standard or Detail for the logging level.
   For viewing, deletion, and printing logging, standard and detail information are the same in logging application that ships with iServer.

   For Factory logging, detailed information includes design parameters. Logging detailed Factory information, instead of standard Factory information, decreases performance.

5. In Usage logging extension name, enter the name of the usage logging extension.

   UsrActivityLoggingExt is the name of the default usage logging extension. Do not append DLL.

   Choose OK.

About types of recorded events

For each type of event, you can set the logging level to Standard or Detail. If you are using the default usage logging extension, UsrActivityLoggingExt, the logging level does not affect how the file records the following types of events:

- Document viewing
- Document printing
- Document deletion

If you set the logging level for document generation or factory events to Detail, the usage log includes design parameters. Setting the logging level to Detail for document generation events decreases performance.

Before you set up system usage for a cluster, you must ensure that each node in a cluster has the same logging applications installed.

About the usage logging extension

You can use the default usage logging extension, UsrActivityLoggingExt, or you can specify the name of your custom DLL or shared library that generates the log files. The default location of the usage logging extension is AC_SERVER_HOME

/bin. Type the name of the DLL or shared library without the file-name extension. For example, on a Windows system, if the DLL is CustomUsage.dll, type:

```
CustomUsage
```

Understanding a usage log entry

Each usage log entry is a comma-separated list containing up to 40 fields of information about an event. The following example describes a delete user event:

```
3272649170,5,1,3272649170,3272649170,-,-,0,Administrator,3,
   urup,urup,urup,User,testUser,-,-,-,-,-,-,-,-,-,-,
   2,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

A dash in the usage log entry means the field is not used. The usage log organizes the entry fields into the following information groups:

- Fields 1 through 10 contain general information:
  - Fields 1, 4, and 5 contain the log file time stamp, start time, and finish time. The time is in seconds since 00:00:00, Jan. 1, 1901, GMT.
  - Field 2 contains the event type. The numeric values in Table 3-1 indicate the event types.

Table 3-1 Event types and the corresponding event values

Event type	Event value
DocumentGeneration	1
DocumentPrinting	2
DocumentViewing	3
DocumentDeletion	4
Admin	5
Query	6
Search	7

  - Field 3 contains the event result. The event result value is either 1 or 0, indicating success or failure.
  - Fields 6 through 8 contain document output information, indicating the file name, version, and file size. The document output group information appears only with document events.
  - Fields 9 and 10 contain execution information, indicating the user name and the iServer subsystem where the operation executed. The numeric values in Table 3-2 indicate the iServer subsystems.

Table 3-2 iServer subsystems and the corresponding ID numbers

Subsystem	ID number
ReportEngine	1
ViewEngine	2
EncycEngine	3
IntegrationEngine	4
Cache	5

- Fields 11 through 25 contain operational information in string format, including the Encyclopedia volume, iServer, and cluster names. Fields 26 through 40 contain operational information in numeric format.

  The values in these fields depend on the value for the event type in field 2. Table 3-3 summarizes some of the information available for each event type at Standard level.

Table 3-3 Examples of information that is available about the different types of events

Event type	Event value	Operation data available
Document generation	1	String fields 11 through 21 display the following information: – `,executable name, executable version, volume name, server name, cluster name, resource group name, node running request, page count, job name, request ID` Numeric fields 26 through 29 display the following information: `number of pages,submit time, job type, job priority`

Document printing	2	String fields 11 through 18 display the following information:
		`page numbers printed, volume name, printer name, server name, clustername, node sent to, file type, server request id`
		Numeric fields 26 through 29 display the following information:
		`number of pages printed, submit time, job type, job priority`

(continues)

Table 3-3 Examples of information that is available about the different types of events (continued)

Event type	Event value	Operation data available
Document viewing	3	String fields 11 through 18 display the following information: `output format, page numbers, volume name, server name, cluster name` Numeric field 26 displays the number of pages viewed.
Administrative	5	String fields 11 through 13 display the following information: `volume name, server name, cluster name` Numeric field 26 displays an operation ID for an administration event. The following list provides the event name for each operation ID: ■ 1 Create ■ 2 Delete ■ 3 Modify ■ 4 Login

Actuate Integration service	6	String fields 11 through 14 display the following information:
		`volume name, server name, cluster name, server request id`
		Numeric fields 26 and 27 display the following information:
		`request wait time, request generation time`

Search	7	String fields 11 through 15 display the following information: `document format, page numbers, volume name, server name, cluster name` Numeric field 26 displays the number of pages viewed.

Configuring error logging

The error log, error_log.csv, is a comma-separated values (CSV) file.If you use the default error logging extension, ErrorLoggingExt, you can set the logging level to the following settings:

- Information
  The error log records messages that trace iServer behavior.
- Warning
  The error log records warnings. The errors do not necessarily affect the operation of iServer.
- Severe
  The error log records errors that can result in iServer failure if you do not correct them.
- Fatal
  The error log records critical errors from which iServer cannot recover and that can result in failure.

How to configure error logging

1. Log in to Configuration Console and choose Advanced view.
2. In System—Status, choose Properties.

   System—Properties—General appears.
3. Choose Error Logging.

   System—Properties—Error Logging appears, as shown in Figure 3-10.

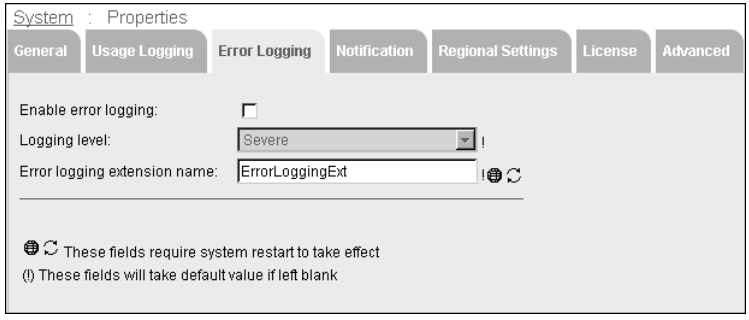

Figure 3-10 Configuring error logging

4. Select Enable error logging.
5. Select the error logging level you want to capture from the following list of options:
   - Information

- Warning
- Severe
- Fatal

6 In Error logging extension name, enter the name of the error logging extension.

ErrorLoggingExt is the name of the default error logging extension.

Choose OK.

Understanding an error log entry

Each error log entry is a comma-separated list containing up to 12 fields about an error-related event. The following example describes an error in a submit job event:

```
3272648796,2,3230,SubmitJob,Administrator,"Invalid start time or
   end time.",urup,urup,urup,-,-,-
```

A dash in the usage log entry means the field is not used. The error log organizes the entry fields into the following information groups:

- Fields 1 through 9 contain general information:
  - Field 1 contains the log file time stamp. The time is in seconds since 00:00:00, Jan. 1, 1901, GMT.
  - Field 2 contains the error severity level, an integer between 1 and 4. The numeric values in Table 3-4 indicate the level.

Table 3-4 Error severity levels and the corresponding values

Error severity level	Value
Information	1
Warning	2
Severe	3
Fatal	4

  - Field 3 contains the Error ID code.
  - Field 4 contains the Service name, indicating the subsystem where the error occurred such as the Factory, Encyclopedia, View, or Request service.
  - Field 5 indicates the Encyclopedia volume user.
  - Field 6 contains the error message.
  - Field 7 contains the Encyclopedia volume name.

- Field 8 contains the iServer cluster name.
- Field 9 contains the iServer node name.
- Depending on the error, fields 10 through 12 can contain information such as a file name and ID number.

  Table 3-5 summarizes some of the information available in fields 10 through 12 for an error log entry at Standard level.

Table 3-5 Information that is available for error log entries at the Standard level

Type of error	Operation data available
Encyclopedia volume user activity	Fields 10 through 12 can contain error parameters such as the following items: ■ Object name ■ ID number
Volume online or offline	Fields 10 and 11 contain the following data: ■ Volume name ■ Operation type either online or offline
iServer node start or stop	Field 10 contains the iServer name.
Service enable or disable	Fields 10 and 11 contain the following data: ■ Server name ■ List of services
Archive service error	Fields 10 through 12 contain error parameters.
Encyclopedia volume job purging Field 4 is Job Purge	Fields 10 through 12 contain error parameters.
Encyclopedia volume health monitoring Field 4 is Encyclopedia Health Monitor	Fields 10 through 12 contain error parameters.

Table 3-6 lists the general categories of iServer error messages.

Table 3-6 Categories of iServer error messages

Error ID range	Error description
0001 - 1000	System errors such as Out of memory or Low thread count
1001 - 3000	iServer errors such as Corrupt encyclopedia or Transient storage full Within this error category, the following sub-categories exist: ■ 1001 - 2000 Actuate internal datastore ■ 2001 - 3000 Actuate internal

(continues)

Table 3-6 Categories of iServer error messages (continued)

Error ID range	Error description
3001 - 6000	User errors such as Permission denied or ROX not found Within this error category, the following sub-categories exist: ■ 3001 - 4000 Encyclopedia engine ■ 4001 - 5000 Report engine ■ 5001 - 6000 View engine
6001 - 12000	■ 6001 - 7000 SOAP engine ■ 7001 - 8000 Process management daemon ■ 8001 - 9000 Cluster engine ■ 10001 - 11000 Server configuration ■ 11001 - 12000 XML parsing
12001 - 13000	Viewing server errors
13000 - 14000	AcMail exceptions
100001 - 100600	Actuate Information service
100601 - 100699	Actuate Caching service
100700 - 150000	Shared by Actuate Information service and Actuate Caching service

Configuring usage and error logging file settings

The administrator configures usage and error logging files in System—Properties—Advanced—Usage and Error Logging—Usage And Error Log File Settings, shown in Figure 3-11.

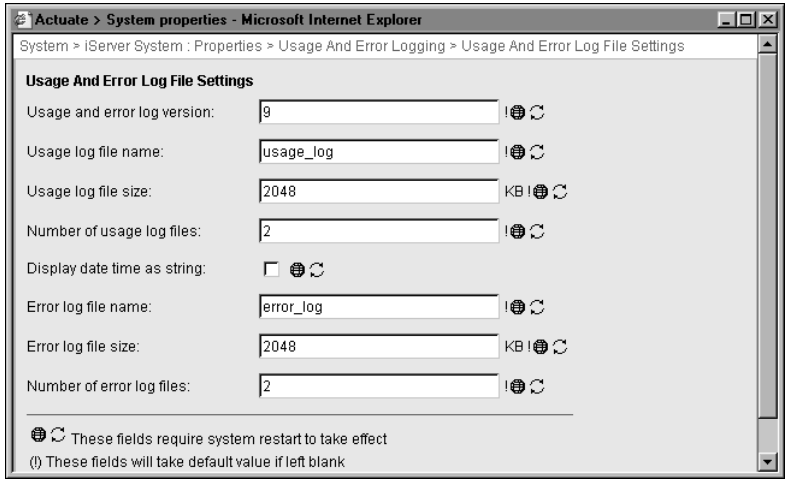

Figure 3-11 Specifying settings for usage and error log files

Table 3-7 describes the properties that appear on System—Properties—Advanced—Usage and Error Logging—Usage And Error Log File Settings.

Table 3-7 Usage and error log file parameters

Property name	Parameter name	Description
Usage and error log version	UsageAndError LogVersion	Use this setting for backward compatibility.
Usage log file name	UsageLogFile Name	Base name for the usage log file, which iServer sends to the usage logging application. The sample usage logging application places the log file in the log subdirectory that contains the usage log records for the volume. The directory for a usage log file is not configurable.
Usage log file size	UsageLogFileSize	If iServer uses multiple log files, this value is the maximum size of each log file.
Number of usage log files	NumberOfUsageLogFiles	Maximum number of usage log files. The usage logging application uses this value to create log file names, such as usage_log.csv and usage_log_1.csv.
Display date time as string	DateTimeAsString	Format of the date and time field for usage and error log entries. The format is either a string in the format mm/dd/yyyy or an unsigned long that specifies the number of seconds since January 1, 1970. The default value is false. The default value uses the unsigned long format.
Error log file name	ErrorLogFileName	Base name for the error log file. The sample error logging application places the log file in the log subdirectory that contains the error log records for the volume. The directory for a error log file is not configurable.
Error log file size	ErrorLogFileSize	Maximum size of an error log file. If iServer uses multiple log files, this is the maximum size of each log file. The error logging application uses this value. A custom error logging application can use a different value.
Number of error log files	NumberOfErrorLogFiles	Maximum number of error log files.

Chapter

4

Configuring e-mail notification

This chapter contains the following topics:

- About e-mail notification
- Adding an SMTP server to the iServer environment
- Specifying the Message Distribution service for e-mail notification
- Setting up Microsoft Exchange e-mail
- Setting up sendmail e-mail notification
- Configuring the notification list size and To: line
- Handling e-mail notification errors
- Customizing the e-mail message

About e-mail notification

The administrator can configure iServer to send e-mail notification to users and groups about completed jobs. After configuring iServer to send e-mail notifications, users trigger notification by setting an option in the schedule of a job.

By default, iServer uses Simple Mail Transfer Protocol (SMTP) to send the e-mail. iServer must connect to an SMTP server that the administrator configures using Configuration Console. To perform e-mail notification, iServer must have the View service enabled. On UNIX and Linux, iServer also supports sendmail.

To set up SMTP e-mail notification in the Advanced view, the administrator enables SMTP and specifies a maximum message size.

Adding an SMTP server to the iServer environment

When using multiple SMTP mail servers, iServer can balance e-mail loads to improve performance, as described in the following procedure.

How to add an SMTP server to the iServer environment

1. In the Advanced view of Configuration Console, choose System.
2. In System—Status, choose Properties, as shown in Figure 4-1.

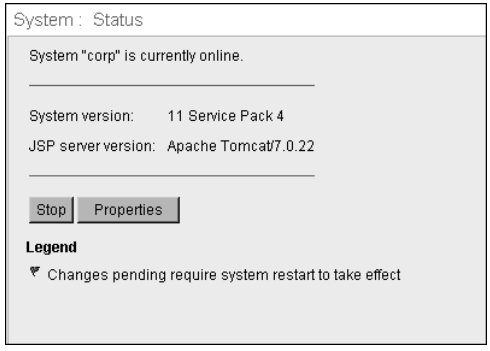

Figure 4-1 Choosing Properties on System—Status

3. In Properties, choose Notification.
4. In Notification, perform the following tasks:
    1. Enable SMTP.
    2. Accept the maximum message size, 5120, or specify a different limit for the size of the message and attachment in kilobytes.

3 Choose Add SMTP Server, as shown in Figure 4-2.

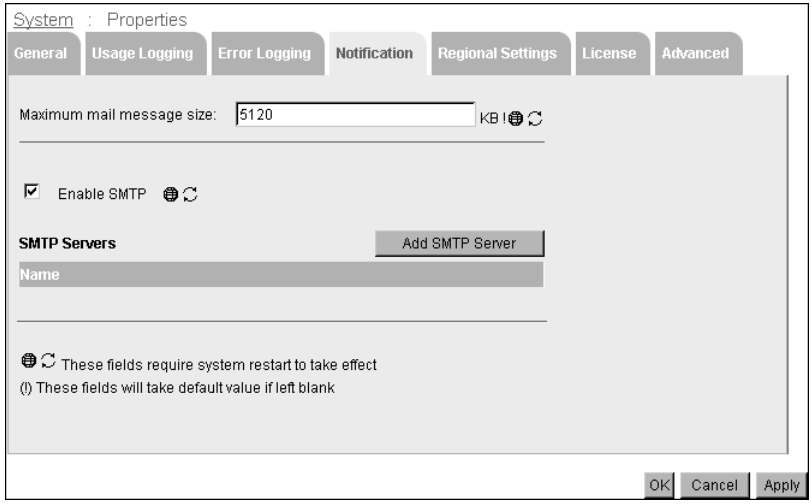

Figure 4-2 Adding an SMTP server to the iServer environment

New SMTP Server appears, as shown in Figure 4-3.

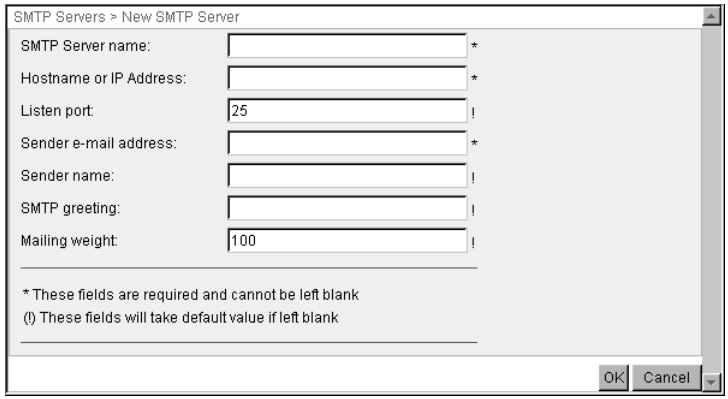

Figure 4-3 Setting properties of the SMTP e-mail server

5 In New SMTP Server, specify the properties SMTP Server Name through SMTP greeting, as shown in Figure 4-4. For more information about configuring SMTP properties, see "Configuring e-mail notification settings" in Chapter 1, "Performing basic configuration tasks."

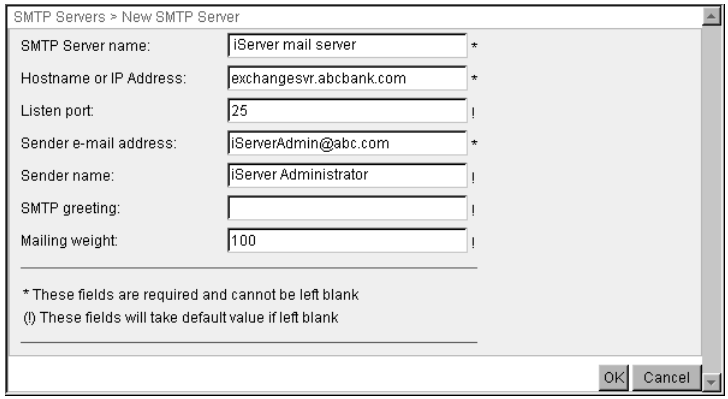

Figure 4-4 Specifying properties of a new SMTP server

6 In Mailing weight, specify the relative weight to use to determine message routing, then choose OK. If the value is zero, iServer uses the SMTP server only after receiving errors from all other SMTP servers in the SMTP server list. Otherwise, it uses load-balancing.

The display name of the new SMTP server that you specified in the first step appears in Notification.

Choose OK.

7 Restart iServer System.

Table 4-1 lists the property names that appear in Configuration Console with the corresponding parameter names in AC_SERVER_HOME/etc /acmetadescription.xml, indicating default settings, ranges, and when a property change takes effect.

Table 4-1 SMTP e-mail notification parameters

Property name	Parameter name	Default	Range	Takes effect
Hostname or IP address	SMTPHostName			Immediate
Listen port	SMTPPort	25	1 - 65535	Immediate
Sender display name	SenderName			Immediate
Sender e-mail address	SenderAddress			Immediate
SMTP greeting	Greeting			Immediate
SMTP mailing weight	MailingWeight	100	0 - 1000000	Immediate

Using SMTP server load balancing

iServer uses load-balancing to send e-mail notifications through SMTP mail servers based on the availability of processing resources. For example, the administrator configures load balancing by setting the Mailing weight parameter values of the first mail server to 40, the second to 30, and the third to 30. The settings establish a relative load-balancing ratio of 4:3:3 among the three mail servers. For every 100 messages iServer distributes, it sends 40 to the first mail server, 30 to the next, and 30 to the last.

When a cluster node receives an error sending an e-mail notice to one mail server, and succeeds in sending the notice to another mail server, the notice counts as part of the load-balancing quota for the mail server that failed. The notice also counts towards the quota of the mail server that succeeds, unless its quota for that round is already exhausted.

Using multiple SMTP servers in a cluster

In a cluster, iServer distributes e-mail notification requests only among the nodes that have the View service enabled. Nodes must have access to the mail servers to send e-mail notices. The example in Figure 4-5 configures two SMTP servers.

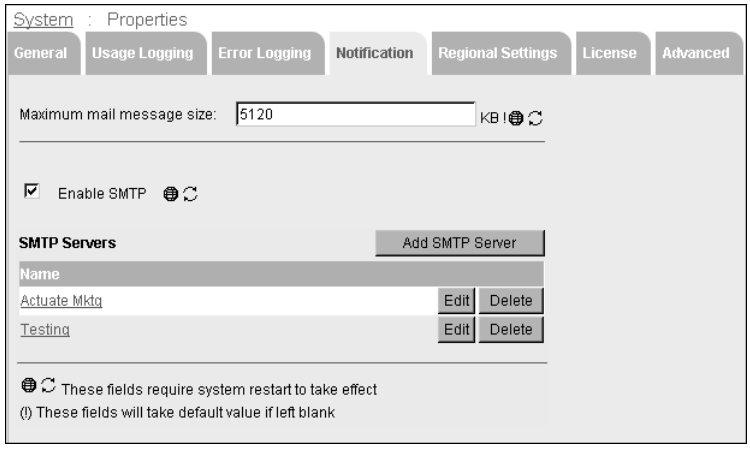

Figure 4-5 Viewing e-mail notification properties

To add SMTP servers to the iServer environment, the administrator modifies the SMTP server setup.

How to modify the SMTP server setup

1. In Configuration Console, in System—Status, choose Properties.

   System—Properties appears.

2. Choose Notification.

System—Properties—Notification appears.

3. To modify the SMTP server setup, perform the following tasks.
   1. Specify a Maximum message size for messages that iServer sends.
   2. Select Enable SMTP to use SMTP e-mail servers for e-mail notification, if necessary. Deselect this option to disable the use of SMTP e-mail servers for e-mail notification.
   3. Choose Add SMTP Server to add an SMTP e-mail server to the list that iServer uses for e-mail notification.
   4. In the list of SMTP e-mail servers, choose Delete to delete an SMTP e-mail server that iServer uses for notification.
   5. In the list of SMTP e-mail servers, choose Edit to edit the parameters for an SMTP e-mail server that iServer uses for notification.
4. Choose OK.

Specifying the Message Distribution service for e-mail notification

The administrator can use the Simple view or the Advanced view to configure iServer to send the e-mail notice to an Information Console user about a completed job. The e-mail message can contain a URL that includes a hyperlink to a design or document and the location of the Message Distribution service (MDS) for connecting to an iServer where the document resides. In the following example, the value of serverURL specifies the MDS:

```
http://sales:8900/iportal/newrequest
   /index.aspx?__requestType=scheduled &__executableName=
   /forecast.rox%3B1&serverURL=http://end2243:8000&volume=volume1
```

By default, iServer uses any node in the cluster that is online and has the Message Distribution service enabled.

In the example, the serverURL parameter and value are:

```
serverURL=http://end2243:8000
```

Include the http:// prefix when you specify the serverURL.

Setting up Microsoft Exchange e-mail

iServer sends Microsoft Exchange e-mail notices from an e-mail account. You must install Windows messaging on the iServer machine, and iServer must use a

Microsoft Exchange profile. Use Mail in the Control Panel folder to create a profile.

Start Actuate iServer 11 service using the iServer e-mail account. For example, the e-mail account is ActuateMail in the PostOff domain, and the iServer user account on the local machine is ActuServer. If you install iServer using the local machine's user account, ActuServer, change the Actuate iServer 11 service's logon account from ActuServer to Postoff\ActuateMail.

About the e-mail account

To use Microsoft Exchange with iServer e-mail notification, you must configure both iServer and Exchange Server if they are in different domains. You must set up an iServer account with the following properties:

- The account running the Actuate iServer 11 service must also be an account on the Microsoft Exchange Server.
- The account running the Actuate iServer 11 service and the account on the Microsoft Exchange Server must use the same password.

For example, iServer is on a machine in the Marketing domain and is running under the ActuServer account. The Exchange Server is on a machine in the PostOff domain. In this scenario, the machine running the Exchange Server must have an account named ActuServer and the password must be the same on both machines, as shown in Figure 4-6.

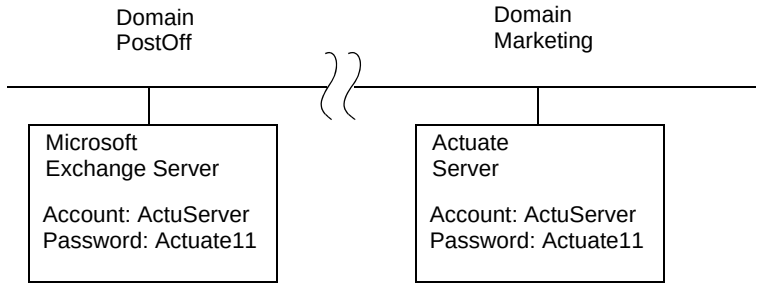

Figure 4-6 Setting up iServer e-mail for Microsoft Exchange

Registering the e-mail account

After setting up the iServer e-mail account and the Exchange profile, register the iServer e-mail account. You must register the e-mail account with each node, and each node must have the View service enabled.

How to register the iServer e-mail account

1. Using the account that runs iServer, run the mailinst program, located in the AC_SERVER_HOME\bin directory.

Actuate iServer E-mail Registration appears, requesting a profile name and a password, as shown in Figure 4-7.

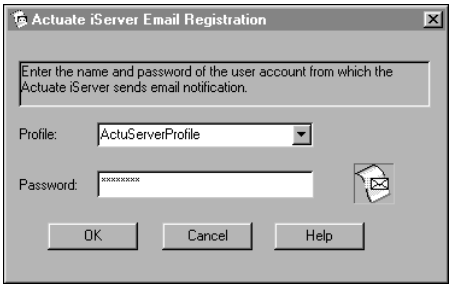

Figure 4-7 Registering the iServer e-mail account

2 Provide the profile information.

- In Profile, select a Microsoft Exchange profile name.
- In Password, type the password.

Choose OK.

Mailinst registers the e-mail account with the iServer machine.

3 Restart iServer.

After iServer restarts, it uses the e-mail profile to send e-mail notification.

After you register the account and restart iServer, it sends e-mail messages using standard tools.

To test the e-mail configuration, send a message to a user from the iServer account.

Setting up sendmail e-mail notification

A sendmail e-mail notification originates from the iServer user account. To send an iServer e-mail notice from a UNIX or Linux system, ensure that the account that runs the iServer can access the sendmail program from each node. Enable the View service on each node. The sendmail program is in /usr/lib/sendmail.

To test the e-mail configuration, send an e-mail message to a user from the iServer e-mail account.

How to test e-mail notification

1 Log in to the account that runs iServer.

2 Send the e-mail message using the following command:

```
/usr/lib/sendmail mail-address < message.txt
```

where

- message.txt contains the test message.
- mail-address is the user's account name as registered with iServer.

3 If the message arrives, the account setup is correct. If the message does not arrive, perform one or more of the following tasks:

- Check the e-mail address for typographical errors.
- Log in to an account other than the account that handles iServer e-mail and try sending e-mail to the user.
- Compare the user's account name that is registered with your e-mail program to the account name registered in the Encyclopedia volume. These two account names must match exactly.

4 If the e-mail does not arrive after you complete the tasks in step 3, and you continue to have problems sending e-mail from the iServer account, contact Actuate Support for assistance.

Configuring the notification list size and To: line

In System—Properties—Advanced—Notification, you can specify how iServer notifies users and administrators of events by setting the following properties, as shown in Figure 4-8.

- Maximum number of recipients per e-mail message
  Parameter name: MaxMailRecipients

  Specifies the maximum number of e-mail addresses to which iServer can address a single e-mail message. If the number of e-mail recipients exceeds the value of this parameter, iServer divides the list into smaller lists and sends the same e-mail message to each of the smaller lists.

  The maximum number of e-mail recipients for an iServer e-mail message cannot exceed the maximum number of e-mail recipients limit for the e-mail server. If the e-mail server receives a request that contains more e-mail recipients than the server permits, it does not send the e-mail message. The e-mail server sends a failure notice to iServer. iServer does not attempt to resend the message.

  The maximum number of e-mail recipients is a system attribute. The default value is 10,000 e-mail addresses. The minimum value is 100. The maximum value is 100,000.

- Use dummy line in place of empty To: line
  Parameter name: UseDummyToLine

Indicates whether to use the value of Dummy To: line in an e-mail notice that iServer sends if both the To: and Cc: values are empty. Applicable to SMTP and sendmail configurations.

- Dummy To: line
  Parameter name: DummyToLine

  Text to use in the To: line of an e-mail notice from iServer if Use dummy line in place of empty To: line is selected and both the To: and Cc: values in the e-mail notice from iServer are empty.

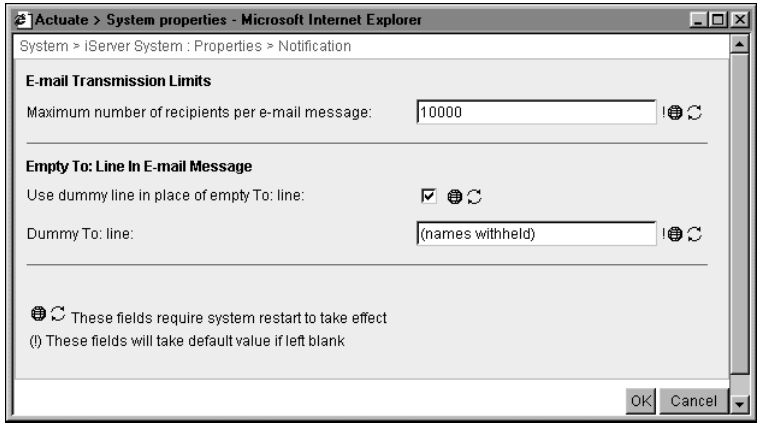

Figure 4-8 Specifying advanced notification property values

Handling e-mail notification errors

If an error occurs when iServer sends an e-mail message, iServer writes a message to the diagnostic log. Under most conditions, iServer also writes a message to the error log. Except under certain conditions, iServer tries to resend the e-mail.

Handling SMTP e-mail errors

When configured to use SMTP, iServer categorizes errors that occur when sending e-mail messages as either retryable or non-retryable. iServer supports resending e-mail messages when a retryable e-mail error occurs.

If a cluster node receives an error when sending an e-mail message to an SMTP server, it sends the e-mail message using the next configured SMTP server. If the node that sends the e-mail message receives non-retryable errors when sending the e-mail message to all SMTP servers, it writes a message to the error log.

If the node that sends the e-mail message receives an error from all the SMTP servers, and at least one of the errors is a retryable error, the node waits for the

retry interval of one second and tries to send the message to the SMTP server that returns a retryable error. If the SMTP server returns an error again, the node writes a message to the error log.

If iServer receives one of the following types of errors, it attempts to resend the e-mail message:

- Cannot connect to SMTP server
- Network connection fails
- Wait for response times out while sending message

For each error iServer receives, it writes a message to the diagnostic log.

If iServer receives one of the following types of errors, it does not attempt to resend the e-mail message:

- Unrecognized recipient
- Message too large
- Message contains too many recipients

Handling sendmail e-mail errors

Using sendmail, iServer makes one attempt to send an e-mail message. If iServer receives an error, it writes a message to the diagnostic log.

About MAPI and sendmail error handling

When iServer uses Microsoft Exchange MAPI on a Windows system and UNIX sendmail on a UNIX system, iServer distributes e-mail notice requests among the nodes that have the View service enabled. A node uses its iServer e-mail account. If the cluster node receives an error, the node writes a message to the diagnostic log.

Logging e-mail notification errors

By default, iServer logs e-mail notification error messages.

- If an error occurs while an Encyclopedia volume is sending an e-mail notice request to a cluster node, the volume writes an error message to its diagnostic log.
- If a cluster node receiving a request to send an e-mail notice encounters an error, the node writes an error message to its diagnostic log. For example, when the combined size of the message text and the attachment exceeds the maximum message size, iServer does not send the e-mail. The diagnostic log in AC_DATA_HOME/server/log describes failed e-mail notification attempts.

iServer localizes all errors that it writes to the log to the default locale for the machine.

Customizing the e-mail message

The administrator can customize the e-mail message that iServer sends to notify users about successful and failed jobs. For each Encyclopedia volume, the default location for the e-mail message template is AC_SERVER_HOME/etc. In a cluster, unless otherwise configured, each View process uses the notification template in the local etc directory when processing e-mail notification.

Sending e-mail notification in a cluster

iServer distributes e-mail notification requests among nodes in a cluster that have the View service enabled. In System—Properties—Advanced—Cluster Operation—Administrative, specify administrative e-mail account information for a cluster node administrator, as shown in Figure 4-9.

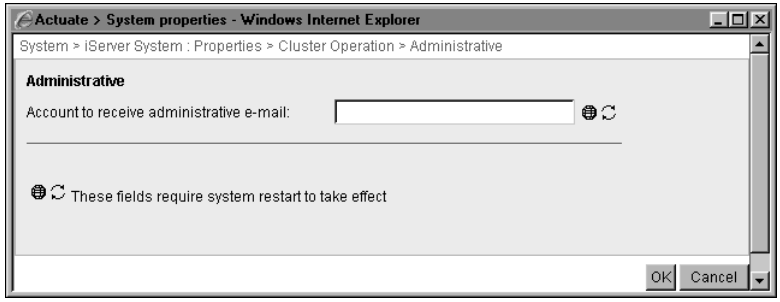

Figure 4-9 Specifying e-mail account information for cluster node administrator

You can configure iServer to send e-mail notices from multiple nodes using a single template.

Sending e-mail notices from nodes using a template

iServer uses an e-mail notification template to create the job completion notification e-mail. In a cluster, unless otherwise configured, each View process uses the notification template in the local AC_SERVER_HOME/etc directory.

If iServer sends e-mail notices from multiple nodes, the nodes use a single template in a partition that all nodes can access. The administrator creates the partition, copies acnotification.xml to the directory the partition specifies, then specifies the partition on Volumes—Properties—General in E-mail notification template partition, as shown in Figure 4-10.

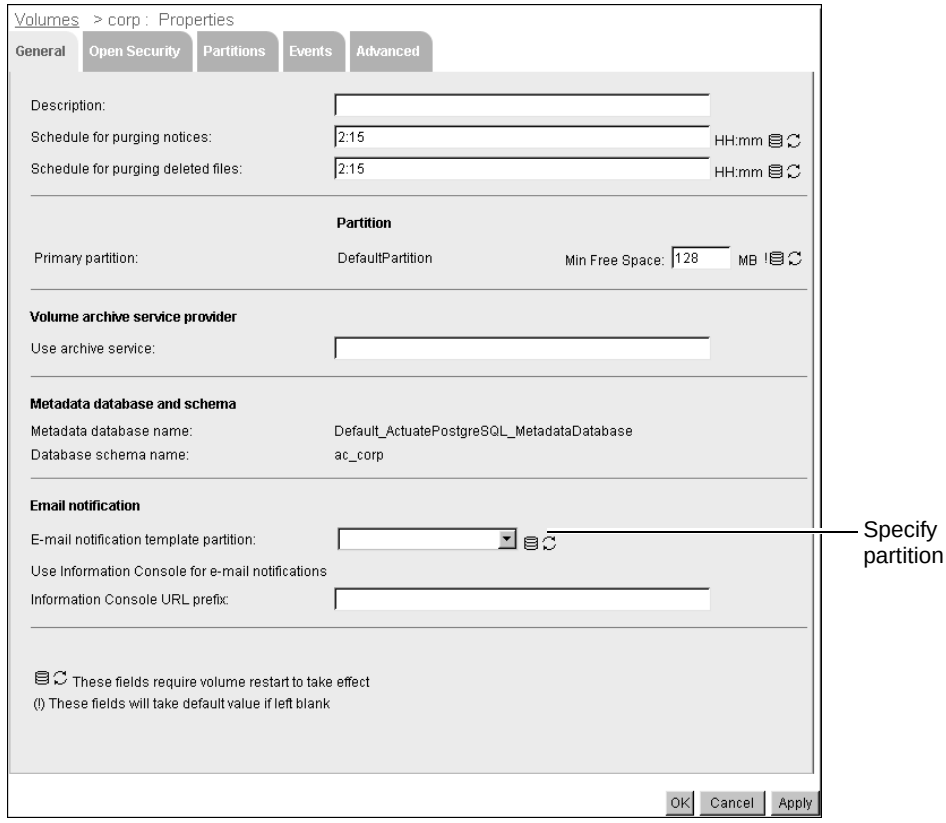

Figure 4-10 Specifying a partition for e-mail notification template file

If iServer cannot find acnotification.xml, or the template file contains invalid formatting, iServer cannot send e-mail notices.

Using the e-mail message template

The e-mail message template, acnotification.xml, is an XML file that uses UTF-8 encoding. iServer constructs the e-mail about job completion notices based on the template. To customize the e-mail that iServer sends, modify the following default template:

```
<?xml version="1.0" encoding="UTF-8"?>
<notificationTemplate version="1.0">
   <successMessage>
      <subject>Actuate iServer Notification</subject>
      <body>
         <
-- Body Text Begin -->
           Actuate iServer - Report <insert variable="jobType"/>
```

```
          complete
          For Information Console:
          Report: <insert variable="reportLink"/>
If the URL above does not work, copy the entire link and paste it
   into the address bar of your web browser, then press Enter or
   Return.
      Completed at: <insert variable="jobCompletion"></insert>
Note: If the job submitter requested that you receive the report
   as an attachment to this email, but the report is not attached,
   then you probably do not have the privileges required to view
   the entire report.  Please contact your system administrator.
        <
-- Body Text End -->
      </body>
   </successMessage>
   <failureMessage>
      <subject>Actuate iServer Notification</subject>
      <body>
         <
-- Body Text Begin -->
          Actuate iServer - Report <insert
             variable="jobType"/>
          failed.
          For Information Console:
          Report: <insert variable="reportLink"/>
If the URL above does not work, copy the entire link and paste it
   into the address bar of your web browser, then press Enter or
   Return.
      Completed at: <insert variable="jobCompletion"></insert>
Warning/Error:
<insert variable="jobDetailedStatus"/>
        <
-- Body Text End -->
      </body>
   </failureMessage>
</notificationTemplate>
```

Working with e-mail template elements

The following list describes the e-mail template elements and text encoding information:

- body

  Element that specifies the content of the e-mail body. The format of the body of the e-mail body can be in any format supported by the e-mail client. The body content can include insert elements.

  Use CDATA sections to embed e-mail body content that is exempt from XML parsing rules. For example, if you specify the e-mail body content in HTML or

any other markup language, the insert elements must be outside of CDATA sections.

- failureMessage
  Parent element of the subject and body for an e-mail notice for a failed job.
- insert element
  Optional element that inserts job and document information in subject and body message content. The value of the variable attribute specifies the information iServer inserts.

  The element must appear outside any CDATA section used within the content of body or subject elements.
- notificationTemplate
  Required root element of the e-mail notification template.
- subject
  Element that specifies the content of the subject line. The subject content is plain text. The tag value or content can include insert elements to display values related to the job or document.

  Use CDATA sections to embed e-mail body content that is exempt from XML parsing rules. The insert elements must be outside of CDATA sections.
- successMessage
  Parent element of the subject and body elements for an e-mail notice for a successful job.

The following list describes the e-mail template attributes:

- email-content-type
  Optional body element attribute that specifies the content type of the body content.

  Use this attribute when constructing the e-mail message, because the e-mail protocol requires that you specify the content type.

  The value for this attribute is either text/plain and text/html.
- variable
  Required insert element attribute. Specifies the information to insert in the e-mail subject or body.
- version
  Required notification template attribute. Specifies the version number of the notification template file.

Using variable attributes

Table 4-2 describes the valid values of variables for insertion in e-mail notices.

Table 4-2 Variable values for e-mail notices

Variable	Value
jobName	Job name.
jobSubmitter	Job submitter's user name.
jobStatus	Status of job: Completed or Failed.
jobDetailedStatus	Detailed status of the job from the job status page.
jobType	Type of job: Execution or Printing.
jobHeadline	The job's headline.
jobCompletion	Date and time of job completion.
reportDocumentName	Document name. Available for a successful job.
reportDocumentVersionName	Document version name. Available for a successful job.
reportDocumentVersionNumber	Document version number. Available for a successful job.
reportLink	Hyperlink to the document in the Encyclopedia volume for a successful job. For a failed job, the link accesses the job status page.

The following example uses the insert element's reportLink variable to display the URL to the document in the e-mail notice that iServer sends:

```
Report: <insert variable="reportLink" />
```

Using HTML in the e-mail template

To use HTML in the successMessage or failureMessage elements, set the message's body email-content-type attribute to text/html.

```
<body email-content-type="text/html">
```

Enclose the HTML in CDATA sections to exclude the HTML from XML parsing rules. The insert elements must be outside CDATA sections. The following example shows a successMessage element with HTML formatting:

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

```
   <subject>
      Report Delivery Notification:
      <insert variable="jobHeadline"/>
   </subject>
   <body email-content-type="text/html">
      <![CDATA[
         <html>
         <body>
         <h2>
      ]]>
      <insert variable="jobHeadline"/>
      <![CDATA[
         </h2>
      ]]>
      Version <insert variable=
         "reportDocumentVersionNumber"/>
         of report <insert variable="reportDocumentName"/>
       is now available online.
      <![CDATA[
         <a href="
      ]]>
         <insert variable="reportLink"/>
      <![CDATA[
         ">Go to Report</a>
         <p>
         <table border="2pt;">
         <tr><td>Report Submitter: </td><td>
      ]]>
      <insert variable="jobSubmitter"/>
      <![CDATA[
         </td></tr>
         <tr><td>Report Generation Date: </td>
         <td>
      ]]>
      <insert variable="jobCompletion"/>
      <![CDATA[
         </td>
         </tr>
         </table>
         <br><br>
      </body>
      </html>
      ]]>
   </body>
</successMessage>
```

Chapter

5

Working with services

This chapter contains the following topics:

- About BIRT iServer and PostgreSQL services
- Using iServer services
- Configuring service properties

About BIRT iServer and PostgreSQL services

The BIRT iServer and the PostgreSQL services must run before other iServer services and processes can start. During iServer installation, accepting the default settings starts the BIRT iServer service automatically when the computer reboots. Accepting the default settings installs the PostgreSQL database. The iServer service appears in the Windows Control Panel, as shown in Figure 5-1.

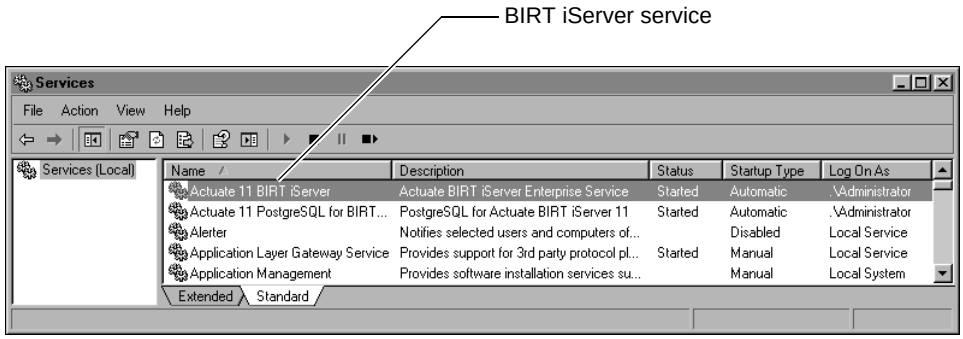

Figure 5-1 Viewing the iServer 11 service

The PostgreSQL for BIRT iServer service also appears in the Windows Control Panel, as shown in Figure 5-2.

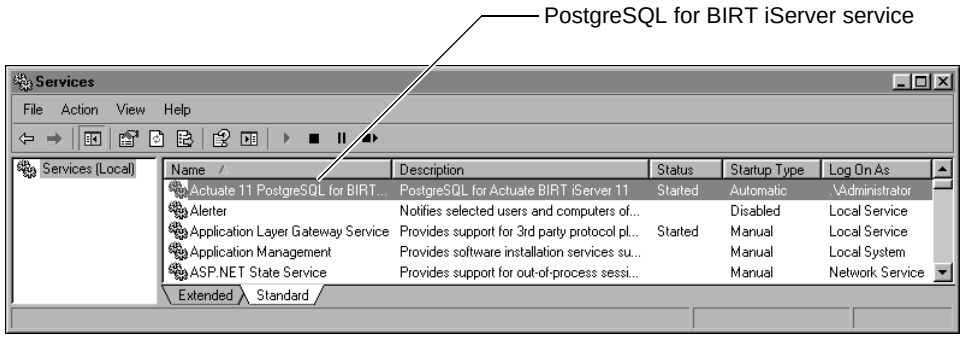

Figure 5-2 Viewing the PostgreSQL for BIRT iServer service

Using iServer services

Understanding iServer services and processes is important for taking advantage of the many configuration possibilities. The administrator can configure iServer as a stand-alone server or as a node in a cluster of iServers. Figure 5-3 shows a stand-alone configuration that runs iServer services on a single machine. Services

publish access methods to the iServer processes through the simple object access protocol (SOAP) interface.

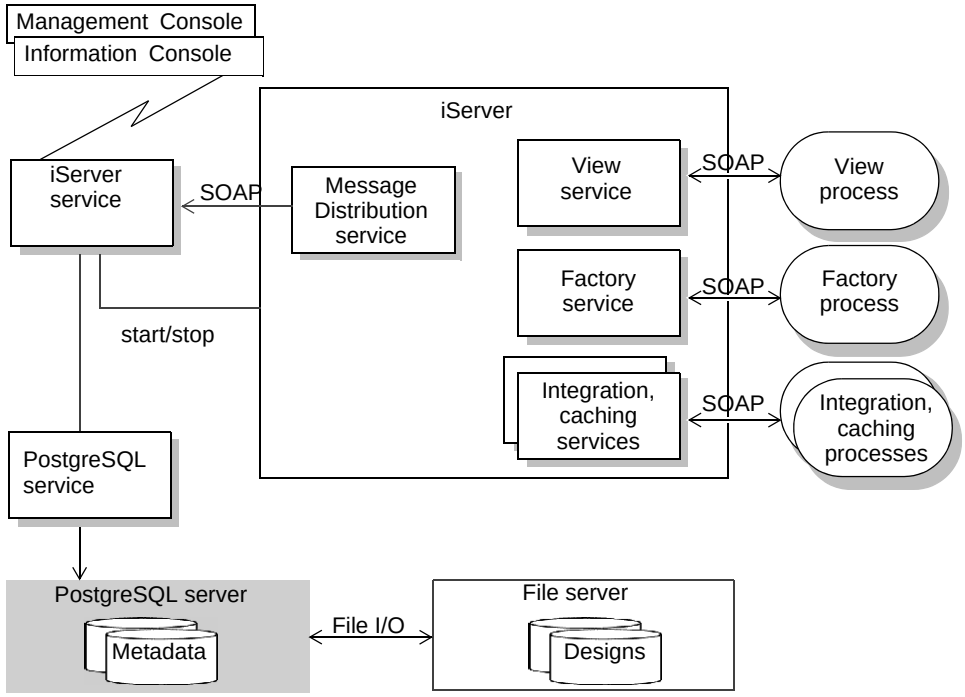

Figure 5-3 iServer architecture

Applications access processes using the Actuate Information Delivery API (IDAPI). IDAPI applications can communicate with cluster nodes and with iServer in a stand-alone configuration. A custom application that uses IDAPI to administer an iServer node sends messages to the cluster through the Message Distribution service.

The following list describes services running inside iServer:

- Message Distribution
  Participates in load balancing by dispatching generation and print requests to other cluster nodes. In a stand-alone iServer or a cluster, the Message Distribution service dispatches SOAP requests that come to iServer from clients, users, and applications.
- View
  Facilitates viewing documents in DHTML format, converting output to formats such as Excel and PDF, and handling requests to download files from an Encyclopedia volume. The Java view process, which handles BIRT designs and spreadsheets, runs inside the View process.

- Factory

  Executes requests to generate queries and documents and to print output on an iServer printer. The Java factory process, which handles BIRT designs and spreadsheets, runs inside the Factory process.

- Integration and Caching

  Coordinates the running of an information object (IOB) file that uses data from multiple data sources. Controls the Actuate Caching process that manages the information object cache and enables caching data from an information object IOB.

To control how iServer uses Factory processes, the administrator creates and manages resource groups using Configuration Console. Resource groups allocate Factory processes to handle prioritized requests for documents.

Configuring service properties

The administrator configures services in the Advanced view of Configuration Console.

How to access service properties

Access service properties settings by performing the following tasks:

1. Log in to Configuration Console and choose Advanced view. From the side menu, choose Server Configuration Templates.
2. On Server Configuration Templates, choose a template name, as shown in Figure 5-4.

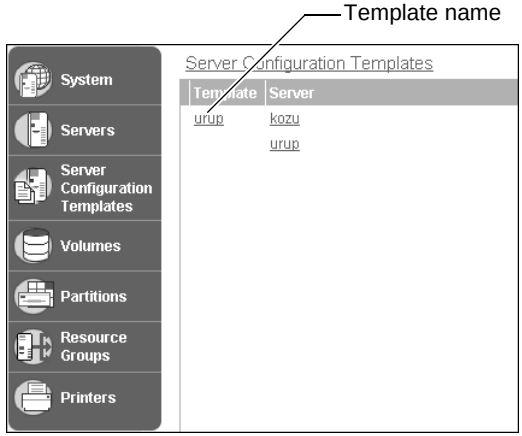

Figure 5-4 Choosing a template name

3 On Server Configuration Templates—Settings, service property folders appear, as shown in Figure 5-5.

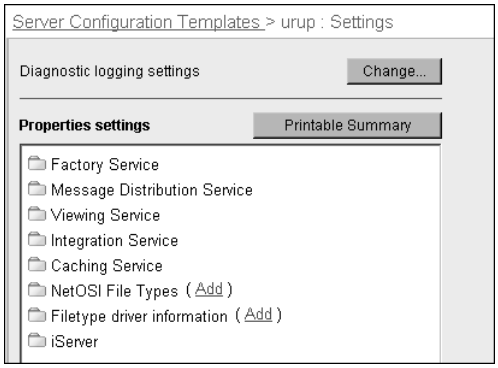

Figure 5-5 Accessing service property folders

Chapter

6

Configuring the View service

This chapter contains the following topics:

- Configuring Viewing service settings
- Configuring the Viewing service for BIRT documents and spreadsheets
- Configuring the Viewing service for e.reports

Configuring Viewing service settings

The Viewing service provides the basic framework to render and display a document. Viewing starts when a user runs a design to create a document or selects an existing document in either Information Console or Management Console. The Viewing service controls viewing a page or the table of contents, and searching a document.

A user navigates to Server Configuration Templates—Settings by choosing Server Configuration Templates from the side menu in the Advanced view of Configuration Console. Then, on Server Configuration Templates, the user chooses a template name, as shown in Figure 6-1.

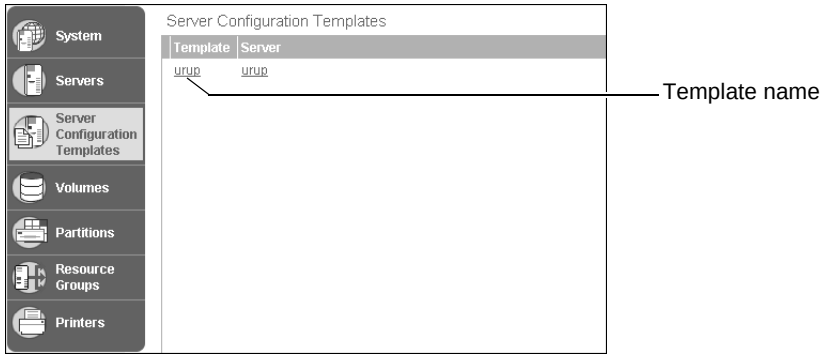

Figure 6-1 Choosing the template name

In Server Configuration Templates—Settings, Viewing service properties that the administrator configures include properties for Actuate Basic, and BIRT designs and documents, as shown in Figure 6-2.

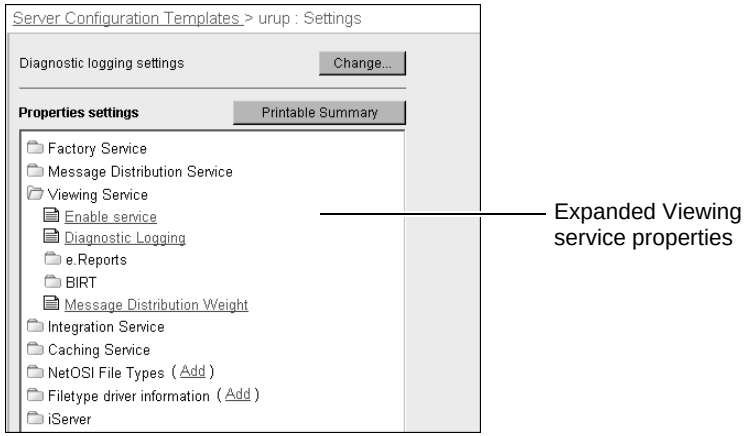

Figure 6-2 Viewing service properties in Settings

About setting Viewing service properties in iServer Release 11

The properties that appeared in Servers—Properties—Viewing service in previous iServer releases appear in Server Configuration Templates—Settings in Release 11. In Release 11, set Viewing Service properties in Server Configuration Templates—Settings as follows:

- To set viewing weight, choose Viewing Service—Message Distribution Weight.
- To set load configuration properties, choose Viewing Service—e.Reports—Process Management—Processes.
- To set Excel spreadsheet generation properties, choose Viewing Service—e.Reports—Report Rendering—Excel Generation.
- To set properties for images in PDF documents, choose Viewing Service—e.Reports—Report Rendering—PDF Generation—Image and Chart Display.
- To set e.Report executable cache settings, choose Viewing Service—e.Reports—Report Content Caches—Report Executable Cache.
- To set e.Report document cache settings, choose Viewing Service—e.Reports—Report Content Caches—Report Document Cache.

Enabling the Viewing service

The administrator can enable or disable the Viewing service from Server Configuration Templates—Settings, or by setting the EnableViewingService parameter in AC_DATA_HOME/config/11SP<service pack number>/acserverconfig.xml to true or false.

How to enable the Factory service

1. Expand Viewing Service and choose Enable service, as shown in Figure 6-2.
2. For Enable viewing service, accept the default value, which is selected, as shown in Figure 6-3.

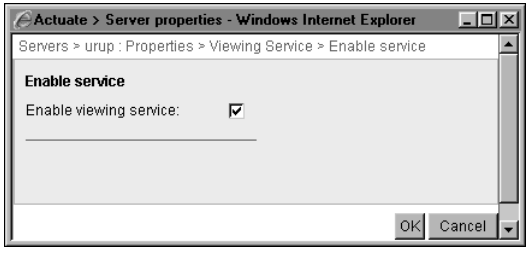

Figure 6-3 Enabling or disabling the Viewing service

About diagnostic logging

The administrator can configure diagnostic logging by expanding Viewing Service, and choosing Diagnostic Logging. For more information, see Chapter 3, "Using diagnostic, usage, and error logging."

Configuring the message distribution weight for a node

To improve performance in an iServer cluster, the administrator can configure the viewing weight for each node to control how iServer distributes the load across the cluster. By default, the cluster master attempts to balance the load by routing requests to the Viewing service of the node that is the least busy.

Setting the viewing weight of a node to a higher number than other nodes routes more requests to that node. Setting the viewing weight to a lower number tends to conserve node resources.

How to configure the message distribution weight for a node

1. In Server Configuration Templates—Settings, expand Viewing Service and choose Message Distribution Weight.
2. In Weight of this server for load balancing viewing requests, accept the default, 100, as shown in Figure 6-4. Alternatively, specify a different value.

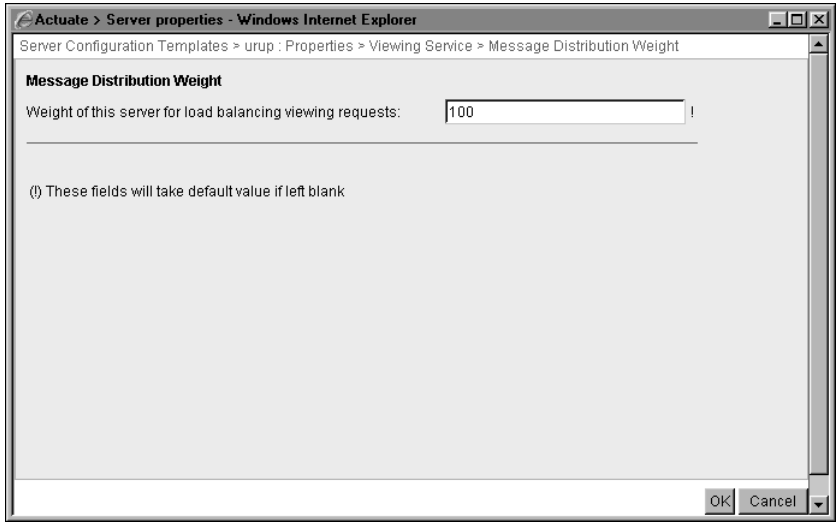

Figure 6-4 Specifying load balancing viewing requests weight

3. Choose OK.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the

Configuration Console property, when a property change takes effect, and the default value:

```
Name="Weight of this server for load balancing viewing requests"
DisplayName="Viewing weight"
TakesEffect="Immediate"
DefaultValue="100"
```

Configuring the Viewing service for BIRT documents and spreadsheets

The iServer architecture incorporates the Java View service running within the Viewing service. The Java View service handles user requests to view BIRT documents and spreadsheets. Figure 6-5 shows the list of Java View service configuration categories. This section describes how to configure properties in these categories, except BIRT JDBC Connection Pool, which is discussed later in this book.

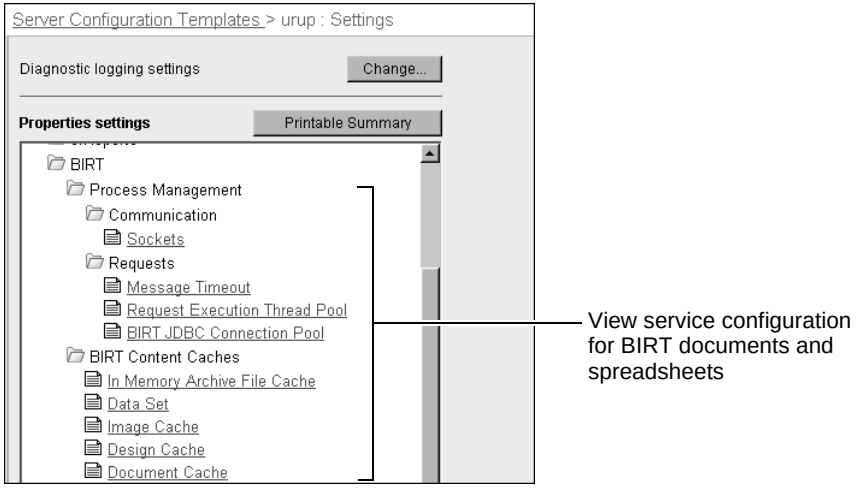

Figure 6-5 Viewing BIRT properties for the Viewing service

Configuring Java process communication

The Viewing service engages in Java process communication when a user views a BIRT document or spreadsheet. In an environment that restricts port usage, the administrator can specify and change the base port for the Viewing service and maximum range of other ports used for SOAP communication.

How to configure Java process communication

1. Expand Viewing Service, BIRT, Process Management, and Communication, then choose Sockets, as shown in Figure 6-5.
2. In Base port number for processes, accept the default base port, 21000, as shown in Figure 6-6. Alternatively, change the base port to a value in the range 1025 through 65535.

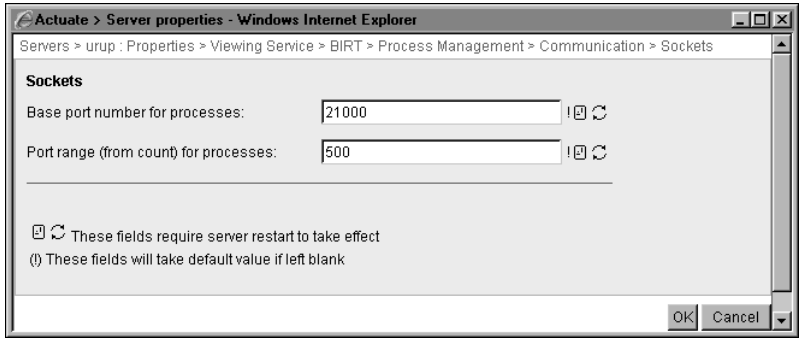

Figure 6-6 Specifying Java View service settings

3. In Port range (from count) for processes, accept the default value, 500, or change the value to a number in the range 0 through 64510.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and ranges:

```
Name="SocketBaseForJavaProcesses"
DisplayName="Base port number for processes"
TakesEffect="ServerRestart"
DefaultValue="21000"
Range="1025 65535"

Name="SocketCountForJavaProcesses"
DisplayName="Port range (from count) for processes"
TakesEffect="ServerRestart"
DefaultValue="500"
Range="0 64510"
```

Configuring Message Timeout

The administrator can set a message time-out period. If the Viewing service does not respond in time to an on-demand request for a BIRT document or spreadsheet, iServer rejects the request. Base the message time-out period on the expected response time of iServer for on-demand generation and loading of the temporary document.

Timeout for generation of gadgets in BIRT 360 specifies the number of seconds that iServer has to generate a gadget. If this time limit is reached, iServer cancels the gadget generation task.

How to configure Message Timeout

1 In Timeout for generation of gadgets in BIRT 360, accept the default value, 300, or change the value to a different number. iServer uses the default, 300, when you set the message time-out to blank, as shown in Figure 6-7. Choose OK.

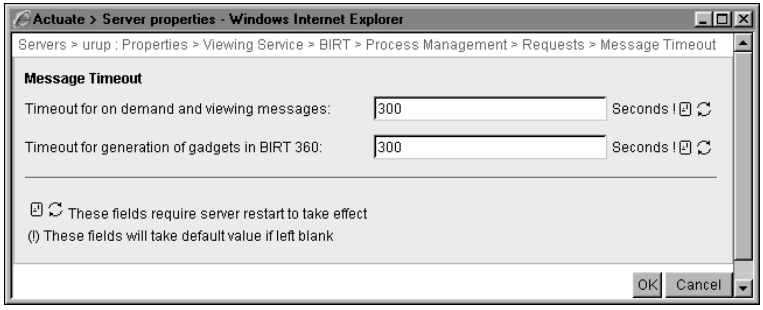

Figure 6-7 Setting time-out periods

2 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and ranges:

```
Name="OnDemandServerViewMessageTimeout"
DisplayName="Timeout for on demand and viewing messages"
TakesEffect="ServerRestart"
DefaultValue="300"
UnitOrFormat="Seconds"

Name="GadgetGenerationTimeOut"
DisplayName="Timeout for generation of gadgets in BIRT 360"
UnitOrFormat="Seconds"
TakesEffect="ServerRestart"
DefaultValue="300"
```

Configuring the thread pool

The administrator can limit the number of threads and size of the queue each on demand server uses for performing the following tasks:

- Generating temporary BIRT documents and spreadsheets
- Processing requests to view temporary and persistent BIRT documents and spreadsheets

To limit the number of threads, the administrator sets Maximum number of threads in each on demand server. The thread batch size is 5. iServer opens five new threads as needed until the total number reaches the limit. For example, setting the limit to 16 causes iServer to open the new threads in batches, 5, 5, 5 and 1. Increasing the value of this property increases CPU and memory consumption.

To set the size of the queue, the administrator sets Task queue size in each on demand server.

How to configure the thread pool

1 Expand Viewing Service, BIRT, Process Management, and Requests, as shown in Figure 6-5, then choose Request Execution Thread Pool.

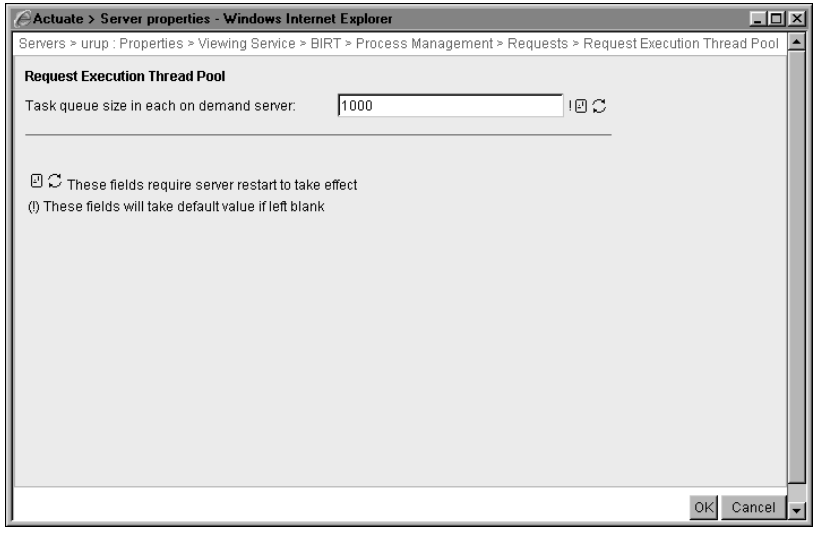

Figure 6-8 Configuring the thread pool

2 In Task queue size in each on demand server, accept the default, 1000, or type another value.

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when property changes take effect, and default value:

```
Name="OnDemandServerQueueSize"
DisplayName="Task queue size in each on demand server"
TakesEffect="ServerRestart"
DefaultValue="1000"
```

Configuring BIRT document and design caching

iServer caches documents and designs to respond quickly to viewing requests. The Viewing service maintains the following types of caches for BIRT documents and designs:

- In-memory archive
- Result set buffer
- Image cache
- Design cache
- Document cache
- Page count cache

The Encyclopedia volume stores persistent documents and designs until removed by the user. Temporary documents are not stored in the Encyclopedia volume. The administrator can configure caching to improve viewing performance for a particular environment.

Configuring the BIRT document in-memory archive

The administrator can configure caching to improve performance of viewing small, temporary and persistent BIRT documents. iServer caches BIRT documents of a size equal to or smaller than the value appearing in Maximum memory size of each BIRT report archive. When a user requests a document of a size larger than this value, iServer temporarily stores the document on the hard drive.

The Total memory size used to cache BIRT document and datamart files value determines how much memory iServer can use to cache documents. Increasing this value caches more BIRT documents in memory, consuming more memory. Decreasing this value temporarily stores more BIRT documents on the hard drive.

To optimize performance, the administrator tries to prevent storing most documents on the hard drive by making configuration decisions based on the size of the average document and the size of the cache.

The administrator can also set the time-out period for temporary and persistent document caches. When the time elapses, iServer clears the cache.

How to configure the BIRT document in-memory archive

1. Expand Viewing Service, BIRT, and BIRT Content Caches, as shown in Figure 6-5, and choose In Memory Archive File Cache.
2. In Total memory size used to cache BIRT document and datamart files, accept the default value, 153600 KB, or type another value, as shown in Figure 6-9. To disable memory-based caching, type a negative value or 0.

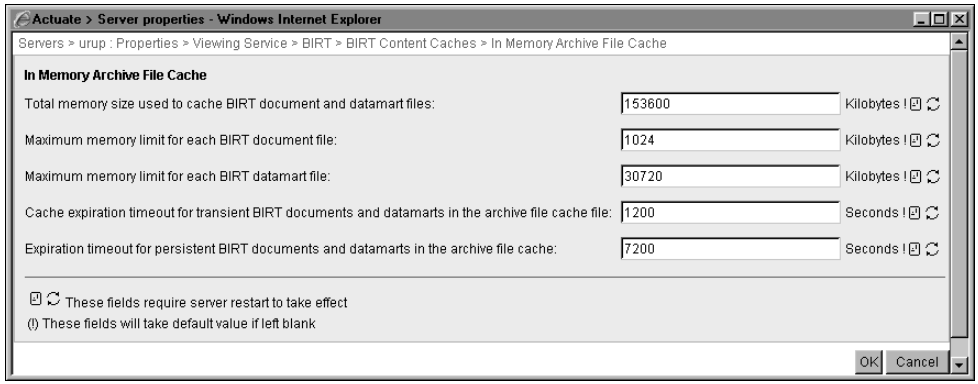

Figure 6-9 Configuring the BIRT document in-memory archive

3 In Maximum memory limit for each BIRT document file, accept the default value, 1024 KB. Alternatively, type a value that represents the size of average document you want iServer to cache. To disable memory-based caching, type a negative value or zero.

4 In Maximum memory limit for each BIRT datamart file, accept the default value, 30720 KB. Alternatively, type a value that represents the size of average datamart you want iServer to cache. To disable memory-based caching, type a negative value or zero

5 In Cache expiration timeout for transient BIRT documents and datamarts in the archive file cache file, accept the default, 1200 seconds, or type another value.

6 In Expiration timeout for persistent BIRT documents and datamarts in the archive file cache, accept the default, 7200, or type another value.

Choose OK.

7 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="TotalArchiveMemory"
DisplayName="Total memory size used to cache BIRT document and
   datamart files"
TakesEffect="ServerRestart"
UnitOrFormat="Kilobytes"
DefaultValue="153600"

Name="MaxMemoryPerArchive"
DisplayName="Maximum memory limit for each BIRT document file"
```

```
UnitOrFormat="Kilobytes"
TakesEffect="ServerRestart"
DefaultValue="1024"

Name="MaxMemoryPerDatamartArchive"
DisplayName="Maximum memory limit for each BIRT datamart file"
UnitOrFormat="Kilobytes"
TakesEffect="ServerRestart"
DefaultValue="30720"

Name="TransientArchiveFileCacheTimeout"
DisplayName="Cache expiration timeout for transient BIRT documents
   and datamarts in the archive file cache file"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="1200"

Name="PersistentArchiveFileCacheTimeout"
DisplayName="Expiration timeout for persistent BIRT documents and
   datamarts in the archive file cache"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="7200"
```

Configuring the BIRT result set buffer

The Java View service handles on-demand document generation requests and multiple concurrent requests, one at a time. Configuring the size of the buffer that stores the result sets for a data set in a BIRT document can improve the response to on-demand requests to generate BIRT documents. An administrator typically changes the size of this buffer from the default to another value under the following conditions:

- The data sets of most BIRT designs are larger than the default buffer size.
- The JVM start arguments include specification of a sufficient heap size to handle the new value.

Increasing the size of the result set buffer for a BIRT data object query increases the memory used to store the final result set, and decreases the disk space used. If the result set is larger than this value, iServer writes the data to disk.

How to configure the BIRT result set buffer

1. Expand Viewing Service, BIRT, and BIRT Content Caches, as shown in Figure 6-5, and choose Data Set.
2. In Maximum buffer size for BIRT Data Object query result set in BIRT 360, accept the default, 8 MB, as shown in Figure 6-10. Alternatively, type a different value.

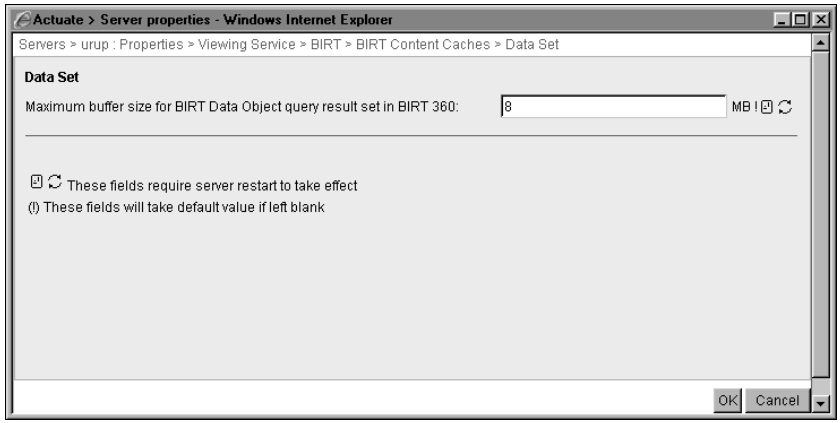

Figure 6-10 Configuring the buffer size for query result set

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, the default value, and range:

```
Name="MaxBIRTDataResultsetBufferSize"
DisplayName="Maximum buffer size for BIRT Data Object query result
   set in BIRT 360"
UnitOrFormat="MB"
TakesEffect="ServerRestart"
DefaultValue="8"
Range="1 256"
```

Configuring the BIRT image cache

The administrator can configure the cache of images in persistent BIRT documents by setting Cache timeout for images and charts from BIRT designs, documents and datamarts. Increasing this value increases the size of the memory buffers for caching images. Graphic-intensive documents load faster, but iServer uses more memory.

How to configure the BIRT image cache

1 Expand Viewing Service, BIRT, and BIRT Content Caches, as shown in Figure 6-5, and choose Image Cache.

2 In Cache timeout for images and charts from BIRT designs, documents and datamarts, accept the default, 86400 seconds, which is one day, as shown in Figure 6-11. Alternatively, type another value greater than 0. A value of 0 or less causes iServer to use a hard-coded value of 5.

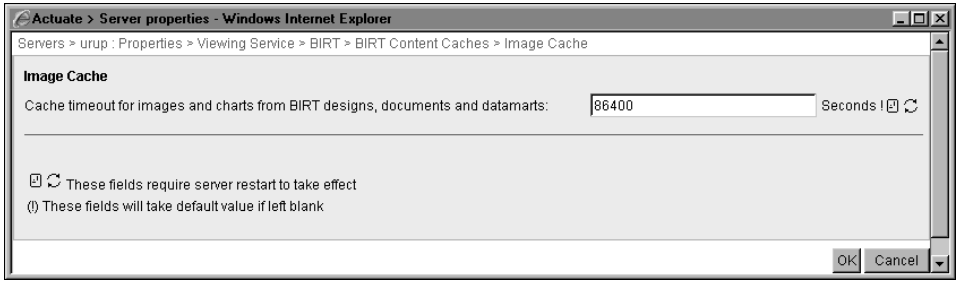

Figure 6-11 Configuring the BIRT report image cache

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, and the default value:

```
Name="BIRTImageCacheTimeout"
DisplayName="Cache timeout for images and charts from BIRT
   designs, documents and datamarts"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="86400"
```

Configuring the BIRT design cache

By default, iServer caches a BIRT design, including access privileges. Caching benefits users who access the design concurrently. Users who request access to the same design share the cached design if they have the required privileges. Performance can improve because iServer does not have to repeatedly load the design. Generally, the fewer number of designs iServer needs to load, the better the response time.

By configuring Cache time-out for BIRT designs, the administrator can control how long the design remains in cache. A cached design persists in memory until time-out occurs. Use this time-out setting to control memory usage.

By configuring Maximum number of BIRT designs, the administrator controls the number of designs in cache. When the cache reaches this limit, design caching stops. Use this setting to improve load performance.

How to configure the BIRT design cache

1 Expand Viewing Service, BIRT, and BIRT Content Caches, as shown in Figure 6-5, and choose Design Cache.

2 In Cache timeout for BIRT designs, accept the default, 1800 seconds or 30 minutes, as shown in Figure 6-12. Alternatively, type another value.

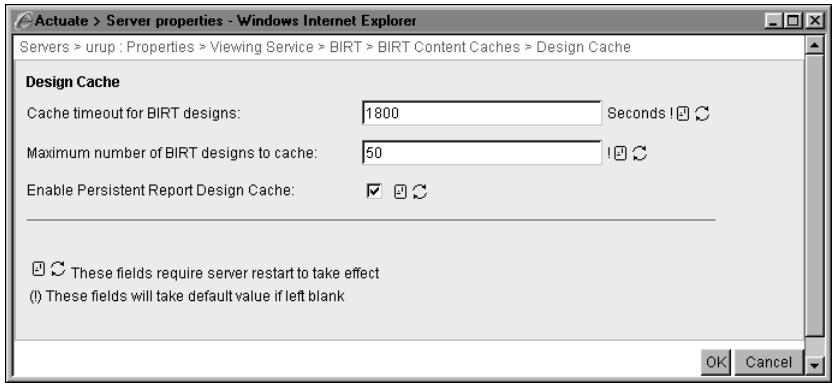

Figure 6-12 Configuring the BIRT design cache

3 In Maximum number of BIRT designs to cache, accept the default, 50, or type another value that limits the number of designs in the cache.

4 For Enable Persistent Report Design Cache, accept the default value of selected. Alternatively, disable the cache by deselecting this option.

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="BIRTReportDesignCacheTimeout"
DisplayName="Cache timeout for BIRT designs"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="1800"

Name="BIRTReportDesignCacheTotalNumberOfEntries"
DisplayName="Maximum number of BIRT designs to cache"
TakesEffect="ServerRestart"
DefaultValue="50"

Name="EnablePersistentDesignCache"
DisplayName="Enable Persistent Report Design Cache"
TakesEffect="ServerRestart"
DefaultValue="true"
```

Configuring the BIRT document cache

By default, iServer caches a BIRT document, including access privileges. Caching benefits users who access the document concurrently. Users who request access to the same document share the cached document if they have the required privileges. Performance can improve because iServer does not have to repeatedly load the document. Generally, the fewer number of documents iServer needs to load, the better the response time. iServer caches BIRT documents in the BIRT document in-memory archive cache. To access the cache, iServer creates a handle. If you enable the BIRT document cache, iServer caches this handle in the BIRT document cache. If you do not enable the BIRT document cache, iServer creates a new handle every time a user chooses to view a document. Enabling the BIRT document cache results in a faster response time but uses more memory, because iServer maintains the BIRT document cache in memory.

How to configure the BIRT document cache

1. Expand Viewing Service, BIRT, and BIRT Content Caches, as shown in Figure 6-5, and choose Document Cache.
2. For Enable caching of BIRT document and datamart handles, accept the default value, selected, as shown in Figure 6-13. Alternatively, deselect the option.

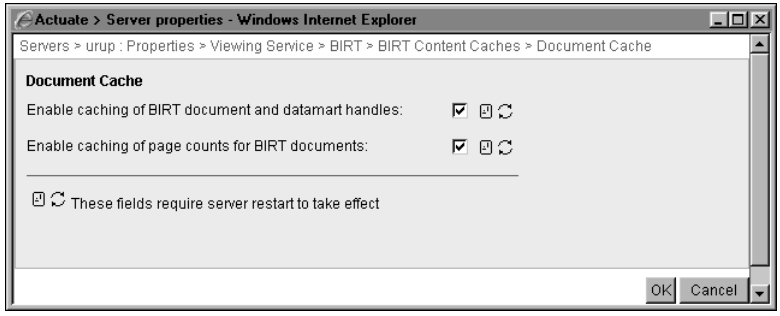

Figure 6-13 Enabling or disabling the BIRT document cache

3. For Enable caching of page counts for BIRT documents, accept the default value, selected, as shown in Figure 6-13. Alternatively, deselect the option.

   Choose OK.
4. Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when a property change takes effect, and the default values:

```
Name="BIRTReportDocumentCacheEnabled"
DisplayName="Enable caching of BIRT document and datamart handles"
```

```
TakesEffect="ServerRestart"
DefaultValue="True"

Name="BIRTReportPageCountCacheEnabled"
DisplayName="Enable caching of page counts for BIRT documents"
TakesEffect="ServerRestart"
DefaultValue="True"
```

Configuring spreadsheet caching

A view cache replacement policy puts spreadsheet data into the cache. The administrator can configure spreadsheet caching to improve viewing performance by modifying the essconfig.xml file in AC_SERVER_HOME \reportengines\engines\ess\essconfig.xml. This file contains the following parameters:

- ViewCacheSize
  The size of the view file cache in megabytes. Too high a value causes fewer documents to be swapped when the cache is full, improving performance but consuming more of the server machine's disk space. The default value is 5000.
- Headless
  True includes the Java graphics environment instead of using the native graphics environment. The default value is False.

Configuring the Viewing service for e.reports

The administrator can configure the Viewing service for e.reports by setting properties shown in Figure 6-14. You can configure the Viewing service for special user requirements, such as the requirement for high quality images in PDF output. The administrator can also tune iServer to improve performance, for example, by running additional View processes during peak times of usage. The following topics describe how to configure the Viewing service for e.reports:

- Configuring the maximum number of View processes
- Configuring the maximum worker threads
- Managing long viewing requests
- Configuring the generation of DHTML output
- Configuring Excel data generated by an e.report
- Configuring PDF conversion
- Configuring render profiles
- Configuring Viewing service caching

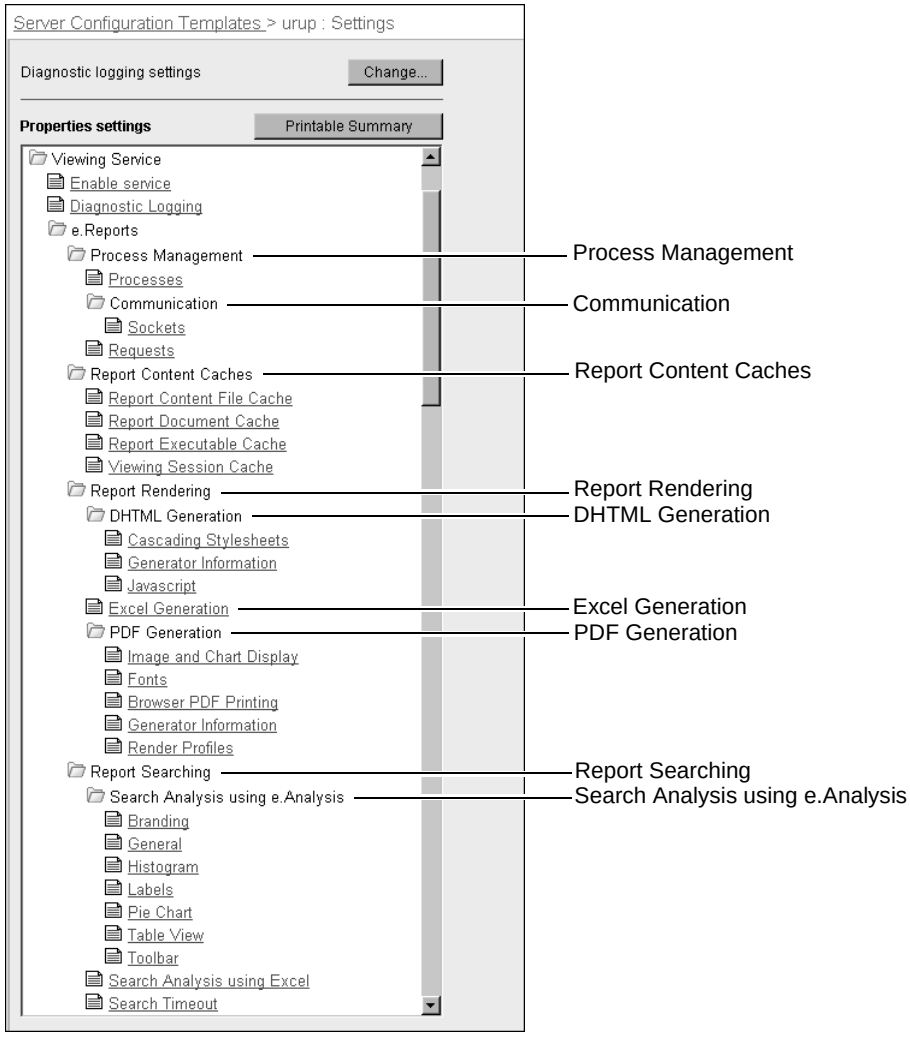

Figure 6-14 Viewing service properties for e.reports

Configuring process management for the Viewing service

In Server Configuration Templates—Settings—Viewing Service—e.Reports—Process Management—Processes, as shown in Figure 6-14, the administrator can set properties for:

- Load management
- Performance

Configuring the maximum number of View processes

By default, the iServer configuration starts one View process. On a single CPU machine, accept the default value of 1 for maximum number of View processes.

On a multiple-CPU machine, set one View process to run for every two CPUs. Monitor CPU usage on the machine using standard operating system administration tools. Configure more View processes to provide additional throughput on iServer during peak processing, as required.

To eliminate the startup time of the View process when iServer receives a viewing request, the administrator can configure iServer to run View processes before receiving any viewing requests. In View process load configuration, change Minimum number of View processes from the default, 0, to another number.

By restricting View process queue size, the administrator can conserve the resources of a node. In Viewing service—e.Reports—Process Management—Requests, change Maximum queue size per process for requests to define the size of the View process queue. If the number of requests for the Viewing service exceeds the queue size, iServer rejects some requests.

Configuring the maximum worker threads

When a user logs in to the Encyclopedia volume and submits a request to view a document, the request creates a new thread in the Viewing service. The thread responds to the request. The administrator can manage Viewing service traffic by raising the maximum number of worker threads.

Maximum number of worker threads per process limits the number of threads that can concurrently handle viewing requests at maximum speed. This value is not a hard limit. The Viewing service continues to handle new viewing requests after reaching the upper limit, but does so at a slower pace, and can reject some requests. To configure a Viewing service to handle a large number of requests and prevent or reduce the rejection of requests, raise the value of maximum number of worker threads per process. Actuate recommends using the default value, 4.

How to configure processes

Changing the maximum number of View processes affects the viewing of Actuate Basic documents only.

1. Expand Viewing Service, e.Reports, Process Management, as shown in Figure 6-14, then choose Processes.
2. In Processes, change Max number of processes from the default, 1, to 2 for a 4-CPU machine, as shown in Figure 6-15.

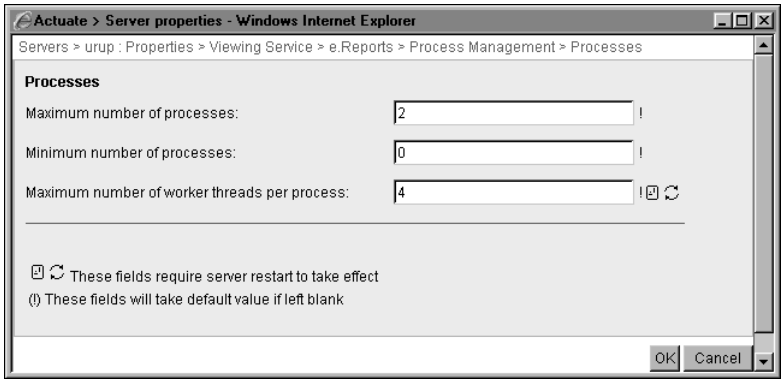

Figure 6-15 Changing the maximum View processes

3 In Minimum number of processes, to prevent delays at start-up time, accept the default, 0, which configures iServer to start a View process only after receiving a viewing request. To improve run-time availability, type a number greater than 0, which configures iServer to start View processes as the volume comes online.

4 For Maximum number of worker threads per process, follow the recommendation of Actuate and accept the default value, 4.

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and ranges:

```
Name="MaxProcesses"
DisplayName="Maximum number of processes"
TakesEffect="Immediate"
DefaultValue="1"
Range="0 128"

Name="MinProcesses"
DisplayName="Minimum number of processes"
TakesEffect="Immediate"
DefaultValue="0"
Range="0 128"

Name="MaxThreadsPerProcess"
DisplayName="Maximum number of worker threads per process"
TakesEffect="ServerRestart"
DefaultValue="4"
```

Configuring communication for e.reports

When working with Actuate Basic documents in an environment that restricts port usage, the administrator can specify and change the base port for the Viewing service and maximum range of other ports used for SOAP communication.

How to configure communication for e.reports

1. Expand Viewing Service, e.Reports, Process Management, and Communication, as shown in Figure 6-14, then choose Sockets.
2. In Base port number for processes, accept the default base port, 18500, as shown in Figure 6-16. Alternatively, change the base port to a value in the range 1025 through 65535.

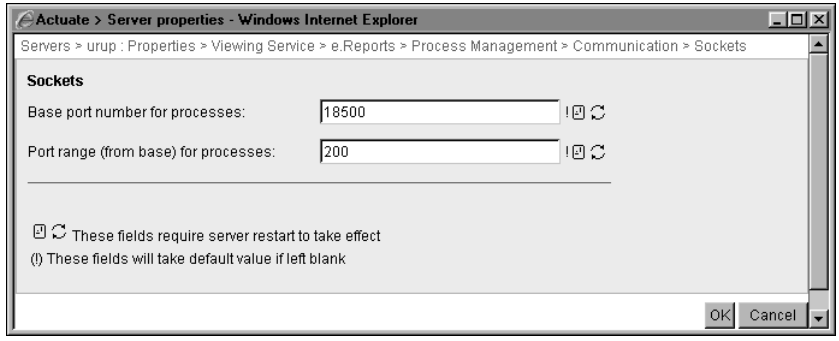

Figure 6-16 Specifying port settings for e.report communication

3. In Port range (from base) for processes, accept the default value, 200, or change the value to a number in the range 0 through 64510.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and ranges:

```
Name="SocketBaseForProcesses"
DisplayName="Base port number for processes"
TakesEffect="ServerRestart"
DefaultValue="18500"
Range="1025 65535"

Name="SocketCountForProcesses"
DisplayName="Port range (from base) for processes"
TakesEffect="ServerRestart"
DefaultValue="200"
Range="0 64510"
```

Configuring request management properties for the Viewing service

By choosing Server Configuration Templates—Settings—Viewing service—Process Management, as shown in Figure 6-14, the administrator can set properties for managing requests for large documents.

Managing long viewing requests

A request from a user to view a large document can block other users from viewing smaller documents. To prevent long viewing requests from blocking short requests, the administrator can configure iServer to handle these requests differently using the following settings:

- Maximum queue size per process for requests
  Defines the size of the View process queue. If the number of requests for Viewing service exceeds the queue size, iServer rejects some requests
- Maximum number of worker threads per process for processing long requests
  Requests for many long documents can occupy all the available threads. An incoming request to view other documents must wait.
- Maximum number of long requests per process
  This setting limits the number of long requests that iServer can actively process and keep in a View process queue. This value must meet the following criteria:
  - Equal to or greater than maximum number of worker threads per process for processing long requests
  - Equal to or less than maximum queue size per process for requests
- Minimum number of pages to qualify as a long request
  This setting specifies the minimum number of pages that iServer uses to determine a long viewing request.
- Minimum report size to qualify as a long request
  This setting specifies the minimum document size, in megabytes, that iServer uses to determine a long viewing request.

iServer considers a viewing request long if the following conditions apply:

- The requested document meets or exceeds minimum report size to qualify as a long request.
- The requested number of pages exceeds minimum number of pages to qualify as a long request.
- Maximum number of worker threads per process for processing long requests is greater than zero.

How to manage long viewing requests

1. Expand Viewing Service, e.Reports, Process Management, as shown in Figure 6-14, and choose Requests.
2. In Maximum queue size per process for requests, accept the default value, 128, or alternatively, specify a different value, as shown in Figure 6-17.
3. Set Maximum number of worker threads per process for processing long requests to 0 to handle all viewing requests the same way. Set the option to 1 or greater to specify the maximum number of threads for handling long viewing requests. iServer rejects a new long request if it is already handling the maximum number of long requests. iServer gives preference to requests to view shorter documents.

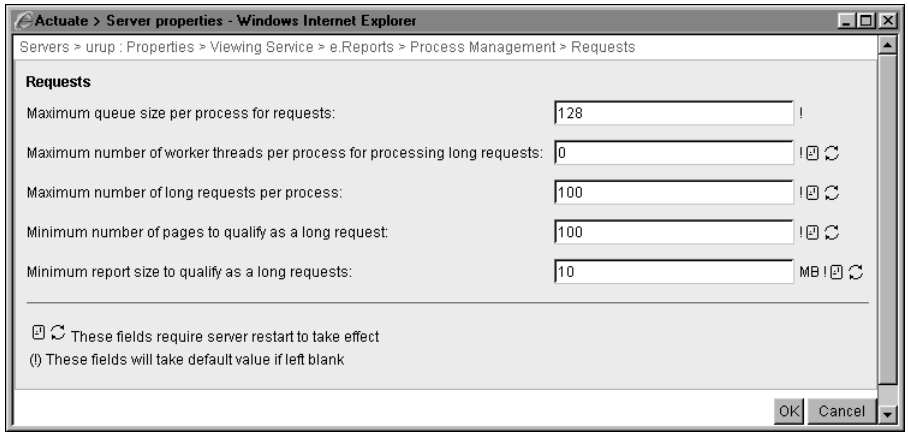

Figure 6-17 Configuring request management properties

4. In Maximum number of long requests per process, accept the default value, 100, unless you expect iServer to receive more than 100 long requests concurrently. If more than 100 long requests ask for service concurrently, to prevent rejection of any requests, type a value equal to, or greater than, the expected number of concurrent service requests.
5. In Minimum number of pages to qualify as a long request, accept the default, 100, or type another value.
6. In Minimum report size to qualify as a long request, accept the default, 10MB, or type another value.

   Choose OK.
7. Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the

Configuration Console properties, when property changes take effect, and default values:

```
Name="MaxActiveLongReqPerProcess"
DisplayName="Maximum number of worker threads per process for
   processing long requests"
TakesEffect="ServerRestart"
DefaultValue="0"

Name="MaxConcurrentRequests"
DisplayName="Maximum queue size per process for requests"
TakesEffect="Immediate"
DefaultValue="128"

Name="MaxActiveLongReqPerProcess"
DisplayName="Maximum number of long requests per process"
TakesEffect="ServerRestart"
DefaultValue="100"

Name="MinPagesForLongRequest"
DisplayName="Minimum number of pages to qualify as a long request"
TakesEffect="ServerRestart"
DefaultValue="100"

Name="MinReportSizeForLongRequest"
DisplayName="Minimum report size to qualify as a long request"
TakesEffect="ServerRestart"
UnitOrFormat="MB"
DefaultValue="10"
```

Configuring the generation of DHTML output

The administrator can set properties for DHTML generation by choosing Server Configuration Templates—Settings—Viewing service—DHTML Generation, as shown in Figure 6-14.

The DHTML output of an e.report document can link to a cascading style sheet and an external JavaScript file. The output can also contain metadata. The following topics describe how to configure the location of these files and configure the metadata:

- Configuring the location of cascading style sheets
- Changing metadata in an e.report DHTML document
- Changing the location of JavaScript files

Configuring the location of cascading style sheets

The administrator can change the location of cascading style sheets (CSS) linked to e.report DHTML documents and viewed in Information Console and Management Console.

How to change the CSS file location

1. Expand Viewing Service, e.Reports, expand Viewing Service, e.Reports, Report Rendering, and DHTML Generation, as shown in Figure 6-14, then choose Cascading Stylesheets.
2. In CSS file location relative to iPortal and Management Console, accept the default, as shown in Figure 6-18. For example, by default, the CSS files for viewing a document in Management Console reside in the following directory:

   ```
   AC_SERVER_HOME/servletcontainer/mgmtconsole/css
   ```

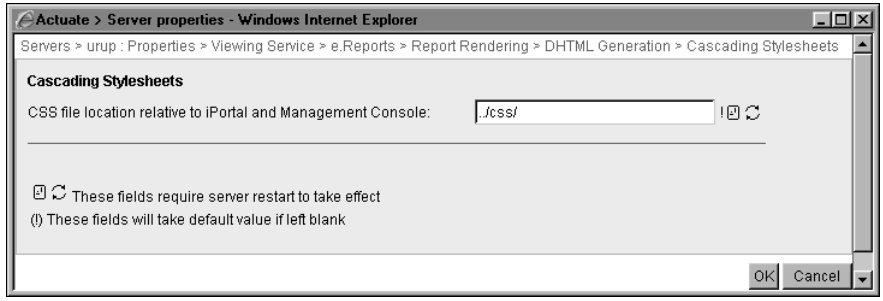

Figure 6-18 Setting the CSS file location

   Choose OK.

3. Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when a property change takes effect, and the default values:

```
Name="CSSFileLocationJSPRC"
DisplayName="CSS file location relative to iPortal and Management
   Console"
TakesEffect="ServerRestart"
DefaultValue="../css/"
```

Changing metadata in an e.report DHTML document

The administrator can set the name of the application that generates an e.report DHTML document in the metadata.

How to change the metadata of an e.report DHTML document

1 Expand Viewing Service, e.Reports, Report Rendering, and DHTML Generation, as shown in Figure 6-14, then choose Generator Information.

2 In DHTML content generator information, accept the default application name, Actuate, as shown in Figure 6-19, or type a new application name. To remove metadata from DHTML output, remove the value, leaving DHTML content generator information blank.

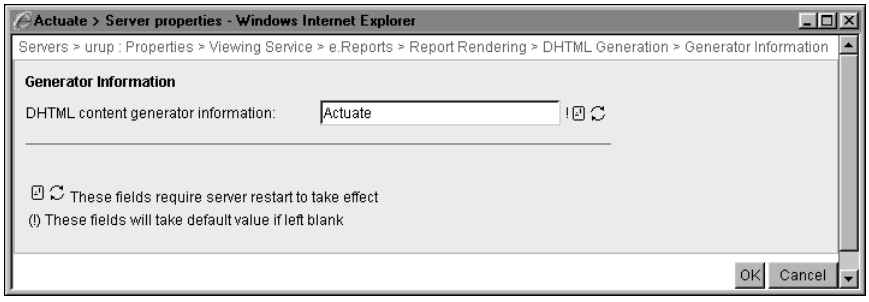

Figure 6-19 Setting the DHTML content generator information

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, and the default value:

```
Name="DHTMLGeneratorInfo"
DisplayName="DHTML content generator information"
TakesEffect="ServerRestart"
DefaultValue="Actuate"
```

Changing the location of JavaScript files

The administrator can change the location of JavaScript files used by Actuate Basic DHTML documents and viewed in Information Console and Management Console. The administrator can also change the location of JavaScript files used by an RSAPI application.

How to change the JavaScript file location

1 Expand Viewing Service, e.Reports, Report Rendering, and DHTML Generation, as shown in Figure 6-14, then choose Javascript.

2 In Javascript file location required for viewing through iPortal/Active Portal /Management Console, accept the default, as shown in Figure 6-20. For

example, by default, the JavaScript files reside in the following directory for viewing a document in Management Console:

```
AC_SERVER_HOME/servletcontainer/mgmtconsole/js
```

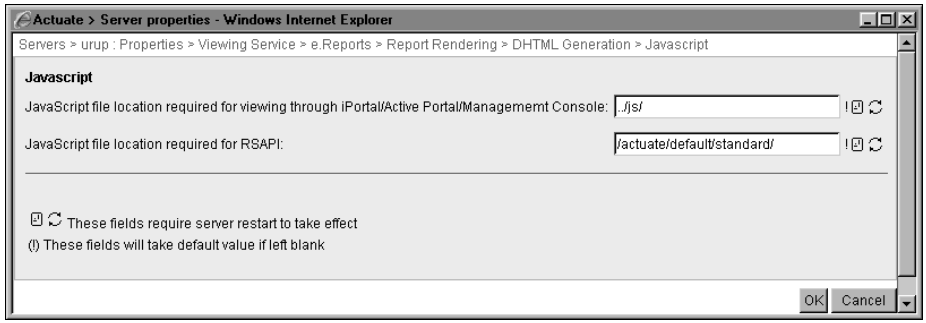

Figure 6-20 Setting the JavaScript file location

3 In JavaScript file location required for RSAPI, accept the default, /actuate /default/standard/. Alternatively, specify a different value.

Choose OK.

4 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="JSFileLocationJSPRC"
DisplayName="JavaScript file location required for viewing through
   iPortal/Active Portal/Managememt Console"
TakesEffect="ServerRestart"
DefaultValue="../js/"

Name="JSFileLocationRC"
DisplayName="JavaScript file location required for RSAPI"
TakesEffect="ServerRestart"
DefaultValue="/actuate/default/standard/"
```

Configuring Excel data generated by an e.report

The administrator can set properties for converting an e.report document to an Excel document by choosing Server Configuration Templates—Settings—Viewing Service—e.Reports—Report Rendering—Excel Generation, as shown in Figure 6-14.

The administrator can improve Viewing service performance by preventing iServer from converting an entire e.report document to Excel data when the

document is too large. The administrator can also solve layout problems that can occur in the Excel data.

Limiting the number of Excel pages to generate

The administrator can limit the number of Excel pages to generate from an e.report design or document by setting Max Excel pages. When converting the output, iServer checks this variable to determine the maximum number of pages to convert. If the output contains more pages than the maximum number, iServer does not convert the excess pages.

The typical range of values for this property is 20 through 2000. Too high a value slows the conversion process. A 0 value results in an empty Excel file. For example, to download a 200-page document, set Max Excel pages to 200 to download the entire document. Typically, the administrator changes the default value when converting more than 50 pages of a document to Excel format.

Solving layout problems in Excel data

To convert an e.report document to Excel, iServer traverses the document to determine the size of the Excel grid and the location of controls. A complete traversal of the document typically degrades iServer performance. By default, iServer traverses only the first 10 pages of a document, then converts the document based on the 10 pages of grid information. In most cases, iServer performs an acceptable conversion based on the incomplete information. Occasionally, there are layout problems or missing controls in Excel data beyond page 10. To solve these problems, select Visit all report pages to identify Excel grid information.

How to limit the number of Excel pages to generate, and choose whether to set iServer to traverse the entire document for Excel grid information

1. Expand Viewing Service, e.Reports, and Report Rendering, as shown in Figure 6-14, then choose Excel Generation.
2. In Maximum number of report pages that will be rendered as Excel, to limit the conversion of e.report documents to 50 or fewer Excel pages, accept the default, 50, as shown in Figure 6-21. To impose a different limit, type another value greater than 0.

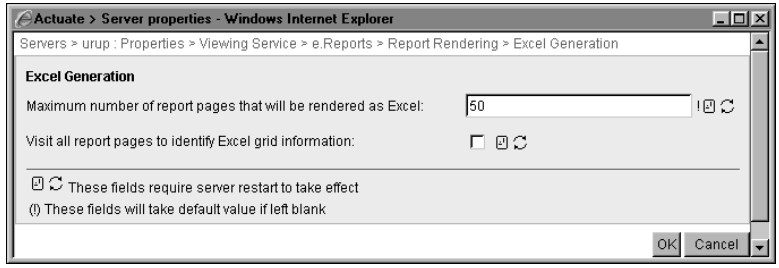

Figure 6-21 Configuring Excel generation properties

3 For Visit all report pages to identify Excel grid information, accept the default value of blank. Alternatively, select this option.

Choose OK.

4 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when a property change takes effect, and the default values:

```
Name="MaxPagesConvertibleToExcel"
DisplayName="Maximum number of report pages that will be rendered
   as Excel"
TakesEffect="ServerRestart"
DefaultValue="50"


Name="VisitAllPagesForExcelGridInfo"
DisplayName="Visit all report pages to identify Excel grid
   information"
TakesEffect="ServerRestart"
DefaultValue="False"
```

Configuring PDF conversion

The administrator can manage the conversion of Actuate Basic designs to PDF format to change how iServer generates the PDF. Configuring PDF generation can improve Viewing service quality and performance. Rendering charts in Actuate Basic designs also affects viewing, particularly when viewing output in PDF format. The following topics describe how to configure PDF generation and chart rendering:

- Setting the quality of JPG images in PDF output
- Configuring the format of charts
- Configuring font encoding
- Selecting the prompt for a paper tray
- Configuring PDF generator information

Configuring the format of charts

For PDF output, the administrator can configure the file format of an Actuate Basic chart to PNG or Rgbzip. The default format is PNG. Rgbzip is a bitmap format of higher quality than PNG images. Selecting Rgbzip increases the file size of the PDF.

Setting the quality of JPG images in PDF output

The Viewing service embeds static images in the PDF output of an Actuate Basic design to JPG or bitmap format. The default is bitmap. The administrator can set the file format to JPG. Converting an image to JPG format can take up to 15 seconds. The administrator can also increase or decrease the resolution of JPG images in the PDF file. Increasing the quality of images increases resource requirements and generates a larger file.

How to set the properties controlling the quality of PDF output

1. Expand Viewing service, e.Reports, Report Rendering, and PDF Generation, as shown in Figure 6-14, then choose Image and Chart Display.
2. In Image quality, accept the default value, 100 percent, or type a new value, as shown in Figure 6-22.

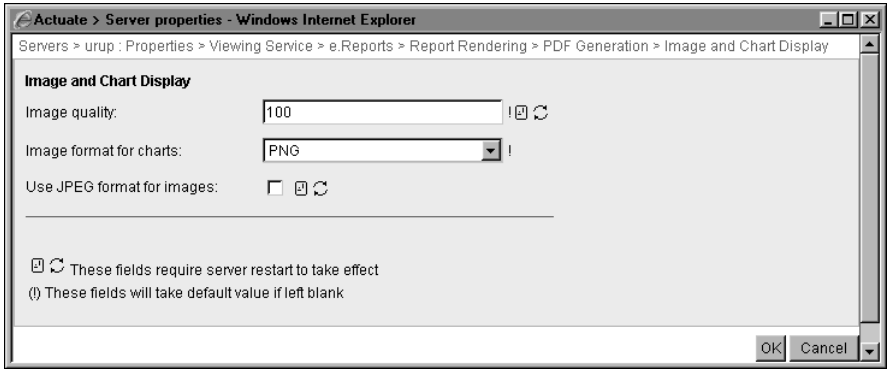

Figure 6-22 Selecting JPG format for static images in PDFs

3. In Image format for charts, accept the default, PNG. Alternatively, select RgbZip.
4. Select Use JPEG format for images. Alternatively, deselect the option.

   Choose OK.
5. Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and range:

```
Name="PDFQuality"
DisplayName="Image quality"
TakesEffect="ServerRestart"
DefaultValue="100"
Name="PDFChartFormat"
DisplayName="Image format for charts"
```

```
TakesEffect="Immediate"
DefaultValue="PNG"
Range="PNG RgbZip"

Name="PDFUseJPEGForImage"
DisplayName="Use JPEG format for images"
TakesEffect="ServerRestart"
DefaultValue="False"
```

Configuring font encoding

By selecting Use font encoding, the administrator can configure iServer to use a cjk.conf file to determine the character encoding in the output of an Actuate Basic design. The cjk.conf file is located in AC_SERVER_HOME/etc. If the cjk.conf file does not list the font used in the design, iServer scans the design for a text string to determine encoding. By default, Use font encoding is not selected and iServer determines the encoding by scanning a design for such text strings.

An Actuate Basic design embeds font information for every font that the designer uses in controls unless the designer chooses not to embed the font. The designer can specify the use of TrueType fonts in an external file. The administrator must set the full path name to the directory containing the TrueType fonts. If iServer cannot find the font information in the specified directory, PDF generation fails.

iServer supports substituting an OpenType font for a TrueType font. To reduce PDF file size, iServer embeds only a subset of the font when generating PDF output.

How to configure font encoding in the PDF output

1. Expand Viewing Service, e.Reports, Report Rendering, and PDF Generation, as shown in Figure 6-14, then choose Fonts.
2. For Use font encoding, as shown in Figure 6-23, accept the default, deselected. Alternatively, select this option.

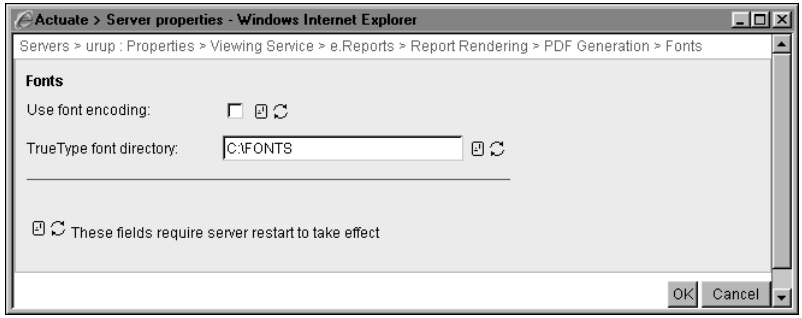

Figure 6-23 Specifying font information for PDF generation

3 In TrueType font directory, type the full path to the font directory. For example, type C:\FONTS, as shown in Figure 6-23.

Choose OK.

4 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and a default value:

```
Name="PDFUseFontEncoding"
DisplayName="Use font encoding"
TakesEffect="ServerRestart"
DefaultValue="False"

Name="PDFFontDirectory"
DisplayName="TrueType font directory"
TakesEffect="ServerRestart"
```

Selecting the prompt for a paper tray

The PDF format supports printer tray selection. By default, the print dialog option is preset to select a paper tray that matches the page size of the Actuate Basic document. The administrator can configure the Viewing service to deselect this option. Deselecting the option prints the PDF using the default printer tray.

How to configure the prompt for a paper tray

1 Expand Viewing service, expand PDF Generation, as shown in Figure 6-14, then choose Browser PDF Printing.

2 Deselect Preset print dialog option to pick paper tray by page size, as shown in Figure 6-24. Alternatively, accept the default, selected.

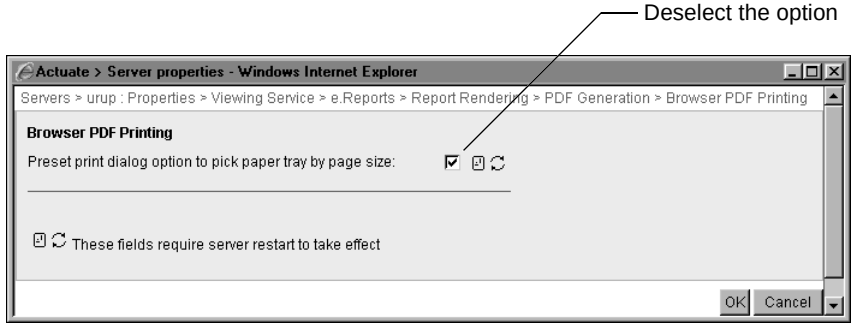

Figure 6-24 Configuring the Viewing service to select the default printer tray

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, and the default value:

```
Name="PDFPickPrinterTrayByPageSize"
DisplayName="Preset print dialog option to pick paper tray by page
   size"
TakesEffect="ServerRestart"
DefaultValue="True"/>
```

Configuring PDF generator information

The administrator can configure the metadata in PDF output to contain the name of the author and the application that produced the PDF.

How to configure PDF generator information

1. Expand Viewing Service, e.Reports, Report Rendering, and PDF Generation, as shown in Figure 6-14, then choose Generator Information.
2. In Creator of the original document (the report), type the name of the author, as shown in Figure 6-25.

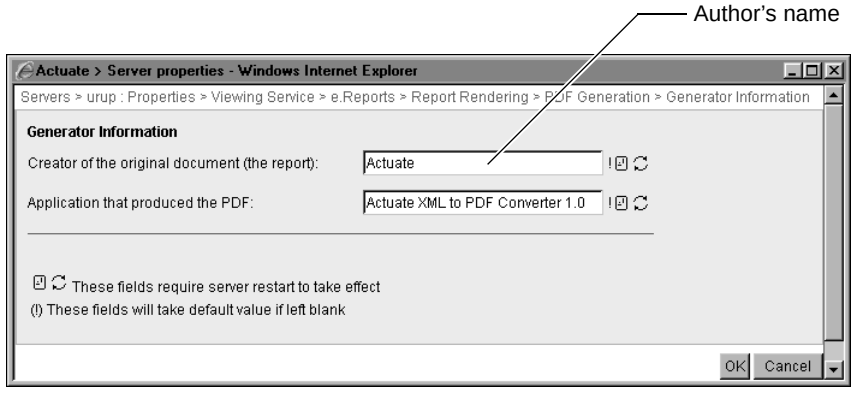

Figure 6-25 Configuring PDF generator information

3. In Application that produced the PDF, accept the default, Actuate XML to PDF Converter, or type the name of another application.

   Choose OK.
4. Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the

Configuration Console properties, when property changes take effect, and the default values:

```
Name="PDFCreator"
DisplayName="Creator of the original document (the report)"
TakesEffect="ServerRestart"
DefaultValue="Actuate"

Name="PDFProducer"
DisplayName="Application that produced the PDF"
TakesEffect="ServerRestart"
DefaultValue="Actuate XML to PDF Converter 1.0"
```

Configuring chart rendering

If you set Server Configuration Templates—Settings—Viewing service—PDF Generation—Image quality to 300, set the chart server maximum heap size to 1024 megabytes (MB), as described in this section.

The chart server renders an Actuate Basic chart for output. Although the chart server is not part of the Viewing service architecture, configuration of rendering in the chart engine affects viewing.

The chart server uses a Java Virtual Machine (JVM) to render a chart. The administrator can set up a parameter string that iServer uses to start the JVM. The string can include standard start options for Java Runtime Environment (JRE), such as the amount of heap to use and the option to include the Java graphics environment, as shown in the following example:

```
-Xmx128M -Xms64M -Djava.awt.headless=true
```

The option -Xmx128M in the string sets the maximum heap size to the default, 128MB. The option -Xms64M sets the minimum heap size to the default, 64MB. The option -Djava.awt.headless=true specifies including the Java graphics environment.

The administrator changes the maximum and minimum heap sizes to improve chart generation performance. Generating PDF output or printing many large charts generally justifies increasing the maximum heap size. Configured using the optimum heap size, the chart server generates charts quickly and does not compete with iServer for memory.

The chart server initializes using the minimum heap size. For example, chart generation for the first several charts likely slows when the minimum heap size is decreased from 64 MB to 16 MB.

How to configure chart rendering

1. In Server Configuration Templates—Settings, expand iServer, then choose Chart Server for e.Reports, as shown in Figure 6-26.

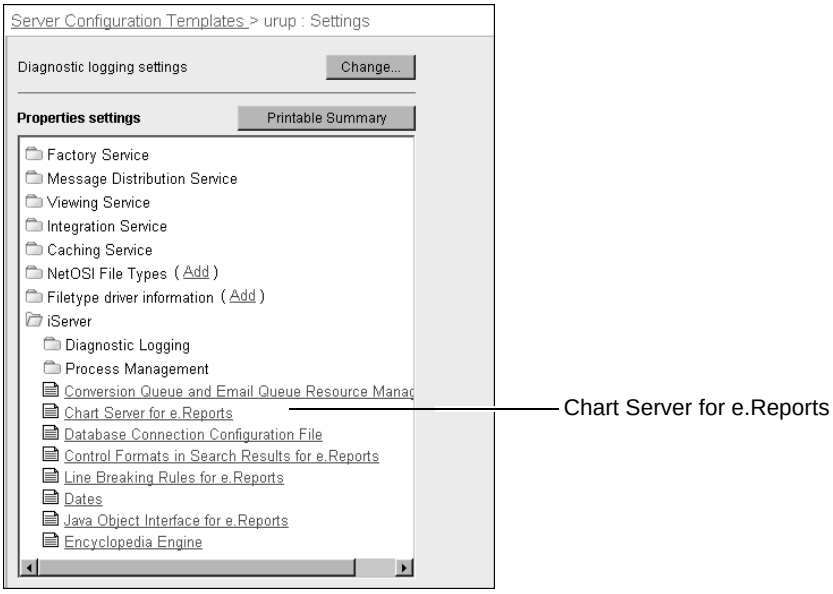

Figure 6-26 Choosing Chart Server for e.Reports

2 In Chart JVM parameter string, accept the default, blank, as shown in Figure 6-27. The chart server uses default start options. Alternatively, type standard start options for the Java Runtime Environment (JRE).

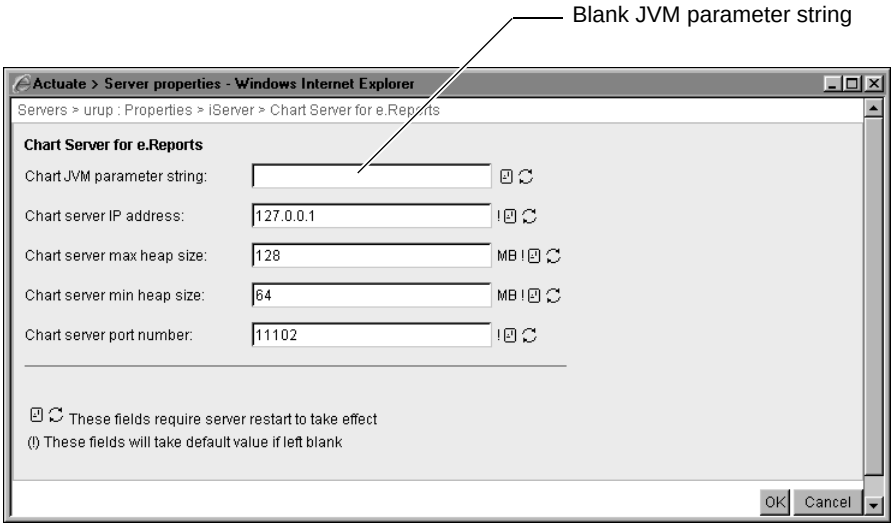

Figure 6-27 Specifying chart server properties

3 In Chart server IP address, accept the default Chart server IP address. Changing this value is not recommended.

4 In Change server max heap size, accept the default, 128, or type a new value for the maximum heap size. For example, type 256, as shown in Figure 6-27.

5 In Change server min heap size, accept the default 64, or type a new value for the minimum heap size.

6 In Chart server port number, accept the default, 11102. Alternatively, if another application uses the port, type a different port number. Do not type commonly used port numbers, such as 80 or 8080.

   Choose OK.

7 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="ChartJVMParameterString"
DisplayName="Chart JVM parameter string"
TakesEffect="ServerRestart"
DefaultValue=""

Name="ChartServerIPAddress"
DisplayName="Chart server IP address"
TakesEffect="ServerRestart"
DefaultValue="127.0.0.1"

Name="ChartServerMaxHeapSize"
DisplayName="Chart server max heap size"
UnitOrFormat="MB"
TakesEffect="ServerRestart"
DefaultValue="128"

Name="ChartServerMinHeapSize"
DisplayName="Chart server min heap size"
UnitOrFormat="MB"
TakesEffect="ServerRestart"
DefaultValue="64"

Name="ChartServerPort"
DisplayName="Chart server port number"
TakesEffect="ServerRestart"
DefaultValue="11102"
```

Configuring render profiles

The administrator configures iServer to use a render profile when converting a certain type of Actuate Basic design output to an output format. In e.Report

Designer Professional, the user must set the Actuate Basic design to use the PDF Writer, compile, and upload the design to iServer.

The administrator configures rendering primarily for exporting documents to PDF. PDF rendering capabilities include setting metadata, such as the title, author, and keywords. The PDF output can include a table of contents based on the standard Actuate Basic table of contents. The administrator can also configure rendering to manage performance by enabling or disabling multithreading during rendering, to control the resolution of displayed charts, and to produce display and print output at different resolutions.

Using a render profile typically produces a smaller PDF and conserves significant processing power, but uses more memory. When not using rendering, iServer supports more character sets and fonts. By default, rendering supports Windows ANSI (code page 1252).

iServer runs the job that generates output using a profile in AcRenderProfiles.xml in the following default directory:

```
AC_SERVER_HOME/etc
```

The administrator can configure the Factory service to relocate the directory. AcRenderProfiles.xml can contain multiple profiles. iServer finds the correct profile to use for rendering based on the PDF Writer settings in the design.

How to configure rendering

1. Expand Viewing Service, e.Reports, Report Rendering, and PDF Generation, as shown in Figure 6-14, then choose Render Profiles.
2. Select Enable rendering using profiles to use Render Profiles in Actuate Basic designs, as shown in Figure 6-28. Alternatively, accept the default value, deselected, to convert Actuate Basic design output to PDFs that use PostScript fonts, character encoding other than Windows ANSI, GIF images, TIFF images, or right-to-left languages.
3. In Render Profiles, accept the path to AcRenderProfiles.xml, or type a new path.

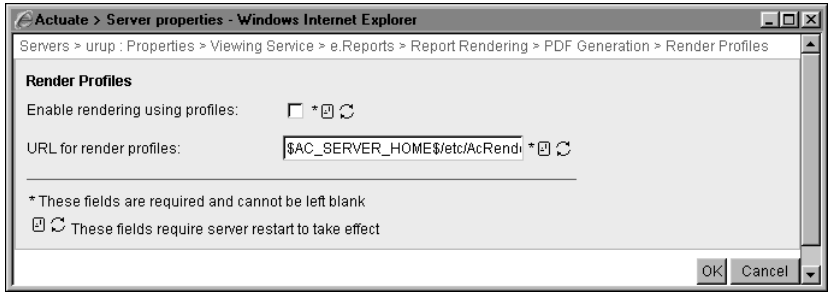

Figure 6-28 Configuring render profiles URL

Choose OK.

4 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes takes effect, and a default value:

```
Name="EnableRender"
DisplayName="Enable rendering using profiles"
TakesEffect="ServerRestart"
DefaultValue="False"

Name="RenderProfilesURL"
DisplayName="URL for render profiles"
TakesEffect="ServerRestart"
```

Configuring Viewing service caching

The Viewing service maintains several types of caches, including a file cache, a session cache, and caches of Actuate Basic compiled designs and documents. The following sections discuss why and how to configure the caches:

- Configuring the file cache of static objects
- Configuring the view session cache
- Configuring the Actuate Basic compiled design cache
- Configuring the extended viewing cache of documents

Configuring the file cache of static objects

iServer caches static objects, such as image files and applets, to respond quickly to user requests for DHTML documents. The viewing time includes the time to render, download, and display the document in a web browser. To configure the file cache, the administrator sets the time-out, maximum file cache entries, and maximum file cache size.

- Setting the file cache time-out

  The administrator can configure how much time elapses before the file cache times out. When the time expires, the Viewing service purges an object from its file cache. To improve Viewing service performance, find a time-out setting appropriate for the documents and user viewing habits. Setting the time-out too low purges cache objects too frequently and degrades performance. Setting the time-out too high displays outdated images in a document. Commonly used time-out periods range from 1 hour through 3 days, which is 3600 seconds through 259200 seconds. The default is 86400 seconds, 1 day.

- Setting the maximum file cache entries

  The administrator can configure the maximum file cache size for each View process. A cluster node can run several View processes simultaneously. The maximum cache size for all View processes on the node must be less than the amount of free disk space available on the machine. If the maximum file cache size is too small, the cache empties too often, degrading viewing performance. If the maximum file cache size is too large, the cache uses too much temporary disk space.

- Setting the maximum file cache size

  The administrator can configure the maximum number of static objects, such as an image, video clip, or applet in the cache. Each cache entry maps to a static object in a document. Actuate recommends setting the maximum number of file cache entries between 500 and 10000.

  Set the maximum file cache size equal to or higher than the maximum file cache entries; otherwise, when the size of the cache exceeds the limit, the Viewing service ignores the higher maximum number of entries setting and replaces cached objects with new ones.

Because the Viewing service cache is disk-based, increasing the time-out period, maximum file cache entries, and cache size generally increases disk space usage.

How to configure the file cache

1. Expand Viewing Service, e.Reports, and Report Content Caches, as shown in Figure 6-14, then choose Report Content File Cache.
2. In Cache timeout for search results, table of contents and image files, accept the default time-out value, 86400 seconds, as shown in Figure 6-29. Alternatively, type a new time-out value in seconds.

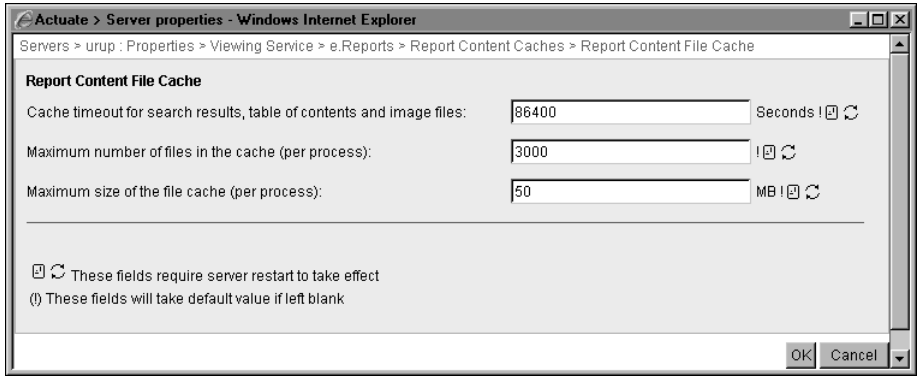

Figure 6-29 Configuring file caching

3. In Maximum number of files in the cache (per process), accept the default, or type a new number of entries. Consider increasing the maximum file cache entries if you increase the time-out.

4 In Maximum size of the file cache (per process), accept the default, or type a new value in megabytes. The memory must accommodate the entries in the cache.

   Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and a range:

```
Name="FileCacheTimeout"
DisplayName="Cache timeout for search results, table of contents
   and image files"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="86400"

Name="MaxFileCacheEntriesPerProcess"
DisplayName="Maximum number of files in the cache (per process)"
TakesEffect="ServerRestart"
DefaultValue="3000"
Range="5 20000"

Name="MaxFileCacheSizePerProcess"
DisplayName="Maximum size of the file cache (per process)"
TakesEffect="ServerRestart"
UnitOrFormat="MB"
DefaultValue="50"
```

Configuring the extended viewing cache of documents

Extending the viewing cache improves the performance of viewing large Actuate Basic documents, especially those using page-level security. Extending the memory-based viewing cache reduces virtual memory in the machine.

The administrator can enable and disable caching for all documents or only page-level-secure documents. By default, the data in the extended viewing cache times out after 1200 seconds, 20 minutes. The cache holds up to 15 entries. The administrator can change the time-out and entry capacity.

How to configure the ROI cache

1 Expand Viewing Service, e.Reports, and Report Content Caches, as shown in Figure 6-14, then choose Report Document Cache.

2 In Enable caching of report documents, accept the default, No Cache, as shown in Figure 6-30. This action disables the extended viewing cache. Alternatively, select Cache all requests. This action caches all documents. To

cache only documents using page-level security, select Cache only page-level security requests by choosing PLSOnly.

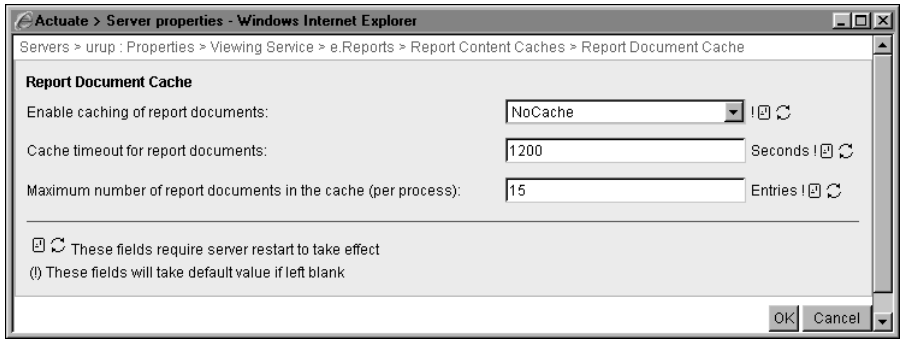

Figure 6-30 Configuring the ROI cache properties

3 In Cache timeout for report documents, accept the default value, 1200 seconds, or change the time-out to another value.

4 In Maximum number of report documents in the cache (per process), accept the default, 15, or change the cache size to another value.

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and a range:

```
Name="ExtendedViewingCache"
DisplayName="Enable caching of report documents"
TakesEffect="ServerRestart"
DefaultValue="NoCache"
Range="NoCache CacheAll PLSOnly"

Name="ExtendedViewingCacheTimeout"
DisplayName="Cache timeout for report documents"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="1200"

Name="MaxExtendedViewingCacheSizePerProcess"
DisplayName="Maximum number of report documents in the cache (per
   process)"
UnitOrFormat="Entries"
TakesEffect="ServerRestart"
DefaultValue="15"
```

Configuring the Actuate Basic compiled design cache

The administrator can configure the number of compiled designs to cache and set the time before the cache times out. When the cache times out, iServer clears the cache. In most cases, using the default values is recommended.

How to configure the Actuate Basic compiled design cache

1. Expand Viewing Service, e.Reports, and Report Content Caches, as shown in Figure 6-14, then choose Report Executable Cache.
2. In Maximum number of report executables in the cache (per process), accept the default, 128, as shown in Figure 6-31. Alternatively, specify a new value.

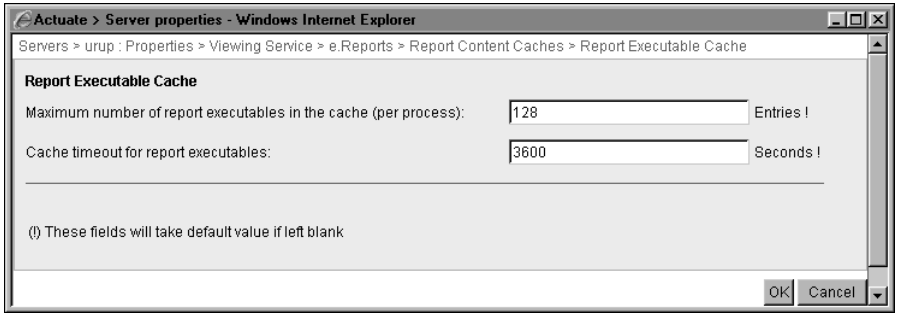

Figure 6-31 Configuring ROX cache properties

3. In Cache timeout for report executables, type the time, in seconds, before the cache times out and iServer clears the cache, or accept the default, 3600 seconds.

   Choose OK.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="MaxROXCacheSizePerProcess"
DisplayName="Maximum number of report executables in the cache
   (per process)"
UnitOrFormat="Entries"
TakesEffect="Immediate"
DefaultValue="128"

Name="MaxROXCacheSizeTimeout"
DisplayName="Cache timeout for report executables"
TakesEffect="Immediate"
UnitOrFormat="Seconds"
DefaultValue="3600"
```

Configuring the view session cache

The Viewing service stores information about a document, session, and user in the view session cache. Caching this information benefits users who view the same Actuate Basic document concurrently. The Viewing service loads the document only once when the first user submits a request to view the document. Other users, who subsequently request the document, view it from the cache. Generally, the fewer number of documents iServer needs to load, the better the response time. The Viewing service periodically checks permissions of users to view the documents. If the user no longer has permission to view the document, the Viewing service invalidates the cache.

To configure the view session cache, the administrator sets an access control list (ACL) cache time-out, the session cache size, and the session cache time-out.

- Setting the ACL cache time-out

  The value of ACL cache time-out determines how often the Viewing service checks the access permissions of a document that a user is viewing. A small ACL cache time-out value causes the View process to check and purge the session cache frequently. Frequent checks and purges can degrade viewing performance.

- Setting the session cache size

  The value of maximum number of cached sessions (per process) limits the number of view sessions in the view session cache. The fewer sessions in the cache, the less memory required. The number of view sessions is the number of users multiplied by the number of different Actuate Basic documents being viewed. When users multiplied by documents exceeds the default value, 4096, consider increasing the maximum cached sessions to improve viewing performance.

- Setting the session cache time-out

  The value of session cache time-out determines how often the Viewing service waits before invalidating the contents of a user's view session cache. After the session cache times out, the Viewing service purges the cache when a user accesses the cache data or when the Viewing service adds a new session to the cache. A small value can cause the View process to purge the cache frequently. Frequent purging can degrade viewing performance.

How to configure the session cache

1. Expand Viewing Service, e.Reports, and Report Content Caches, as shown in Figure 6-14, then choose Viewing Session Cache.
2. In Cache timeout for user ACL, accept the default time-out value, 3600 seconds, as shown in Figure 6-32. Alternatively, type a new time-out value in seconds.

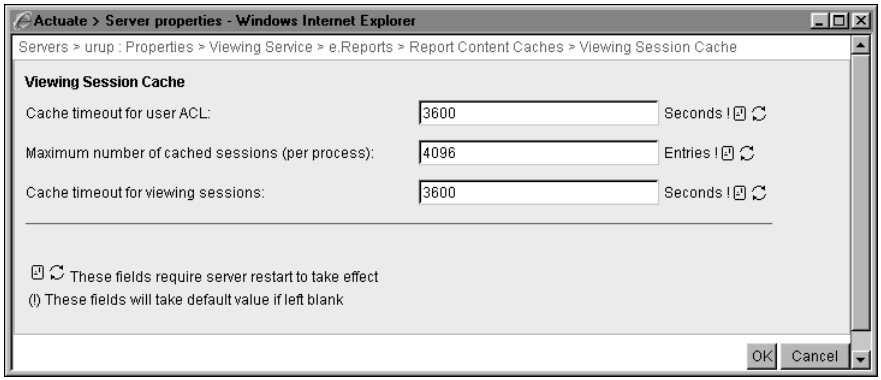

Figure 6-32 Specifying session cache properties

3 In Maximum number of cached sessions (per process), accept the default, or type a new number of sessions. Consider increasing Cache timeout for user ACL if you increase Maximum number of cached sessions (per process).

4 In Cache timeout for viewing sessions, accept the default, or type a new value in seconds. Consider increasing the time-out if you increase Maximum number of cached sessions (per process).

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="ACLTimeout"
DisplayName="Cache timeout for user ACL"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="3600"

Name="SessionCacheSizePerProcess"
DisplayName="Maximum number of cached sessions (per process)"
UnitOrFormat="Entries"
TakesEffect="ServerRestart"
DefaultValue="4096"

Name="SessionCacheTimeout"
DisplayName="Cache timeout for viewing sessions"
TakesEffect="ServerRestart"
UnitOrFormat="Seconds"
DefaultValue="3600"
```

Configuring report searching for the Viewing service

By choosing Server Configuration Templates—Settings—Viewing Service—e.Reports—Report Searching, as shown in Figure 6-14, the administrator can set properties for searching documents.

Configuring the Viewing service for e.Analysis documents

The administrator can configure the Viewing service to customize e.Analysis document viewing by choosing Server Configuration Templates—Settings—Viewing Service—e.Reports—Report Searching—Search Analysis using e.Analysis, as shown in Figure 6-14. A user must search DHTML documents to generate an e.Analysis document. The administrator can configure the Viewing service for searching DHTML documents.

Configuring e.Analysis branding

The administrator can specify the main title of the e.Analysis view, and the browser title bar that displays query output. Using the default value of blank for Main Title entitles the window by concatenating the names of the categories, or dimensions, of the columns and rows.

Microsoft Access requires the following default watermark to build a cube:

```
daedclejcpafef
```

How to configure e.Analysis branding

1. Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose Branding.
2. In Main title, accept the default, blank, as shown in Figure 6-33. This action entitles the document using names of the column and row categories. Alternatively, type a title up to four lines long, using backslash n, \n, to start a new line.

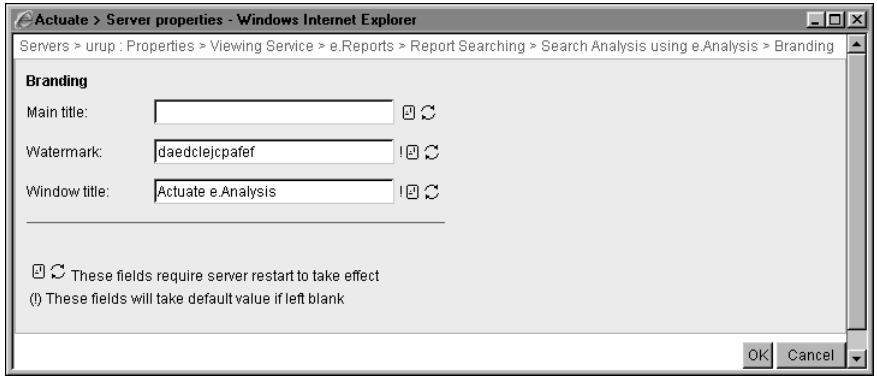

Figure 6-33 Configuring e.Analysis branding

3 In Watermark, accept the default value.

4 In Window title, accept the default, Actuate e.Analysis, or type another name for the title bar.

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="eAnalysisMainTitle"
DisplayName="Main title"
TakesEffect="ServerRestart"

Name="eAnalysisWatermark"
DisplayName="Watermark"
TakesEffect="ServerRestart"
DefaultValue="daedclejcpafef"

Name="eAnalysisWindowTitle"
DisplayName="Window title"
TakesEffect="ServerRestart"
DefaultValue="Actuate e.Analysis"
```

Configuring general e.Analysis viewing

The administrator can limit the size of the cube that e.Analysis accepts. e.Analysis rejects a user request for a larger cube.

The administrator also configures the number of decimal points in the data stored in Microsoft Access. The setting does not affect the formatting of the data in the document. Actuate recommends setting the number of decimal points to one greater than the largest number of decimal places in the data.

The administrator configures the application code base path for viewing either XMLAPI or RSAPI applications using Information Console or Management Console. XMLAPI applications require the following path:

```
../eanalysis
```

RSAPI applications require the following path:

```
/actuate/default/eanalysis
```

How to configure general e.Analysis viewing

1 Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose General.

2 In Data Cube Size, accept the default, 6, as shown in Figure 6-34. Alternatively, type another value.

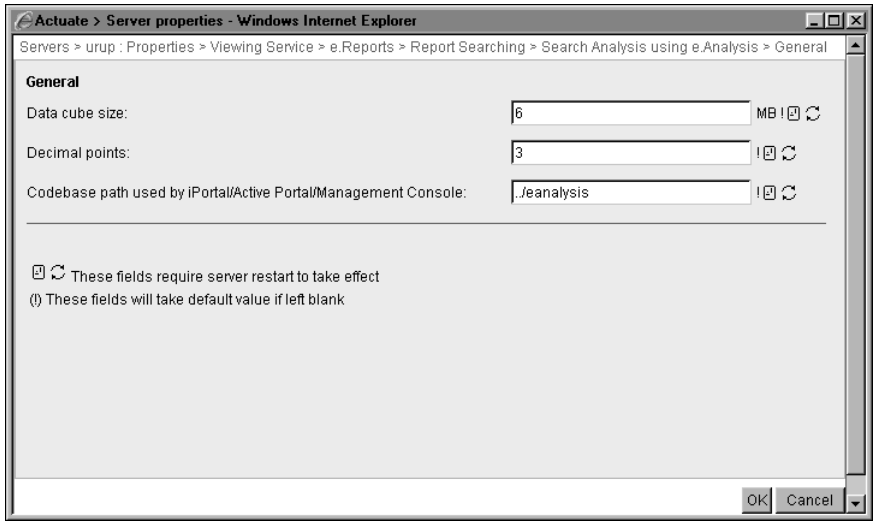

Figure 6-34 Configuring general e.Analysis viewing

3 In Decimal points, accept the default, 3, or type another value.

4 In Codebase path used by iPortal/Active Portal/Management Console, accept the following default path for XMLAPI code:

```
../eanalysis
```

Alternatively, type the following path for RSAPI code:

```
/actuate/default/eanalysis.
```

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="EnableViewingService"
DisplayName="Enable viewing service"
TakesEffect="Fixed"
DefaultValue="False"

Name="eAnalysisCubeSizeLimit"
DisplayName="Data cube size"
TakesEffect="ServerRestart"
UnitOrFormat="MB"
```

```
DefaultValue="6"
Name="eAnalysisDecimalPoint"
DisplayName="Decimal points"
TakesEffect="ServerRestart"
DefaultValue="3"

Name="eAnalysisPathJSPRC"
DisplayName="Codebase path used by iPortal/Active Portal/
   Management Console"
TakesEffect="ServerRestart"
DefaultValue="../eanalysis"
```

Configuring e.Analysis histograms

The administrator can configure the output of histograms, commonly called bar charts, for viewing in two or three dimensions. The administrator can also display data as a percentage, or not, and hide data in the histogram output.

How to configure e.Analysis histograms

1. Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose Histogram.
2. In Show 3D Histograms, accept the default, which is selected, as shown in Figure 6-35. This setting shows output in three dimensions. Alternatively, deselect the option. This setting displays the output in two dimensions.

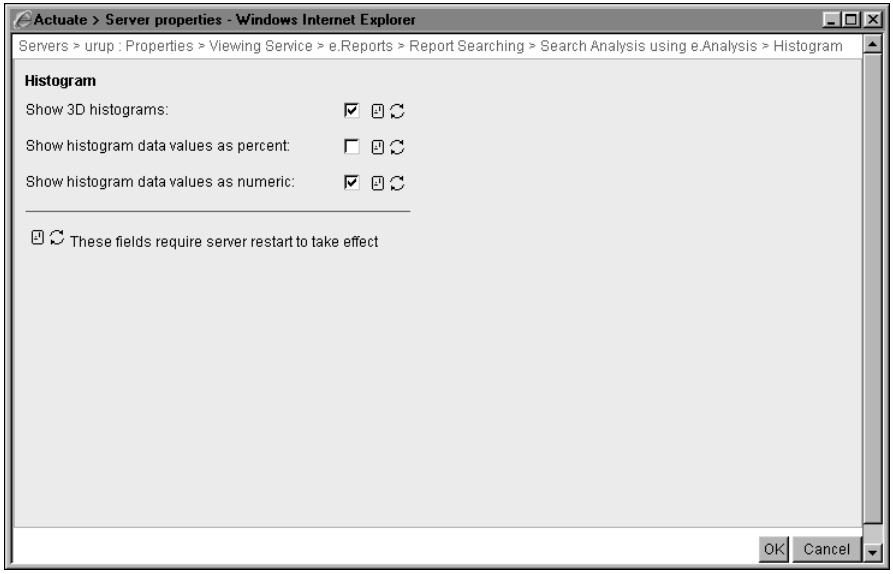

Figure 6-35 Configuring e.Analysis histograms

3 In Show histogram data values as percent, accept the default, which is not selected. Alternatively, to show values as percentages, select the option.

4 In Show histogram data values as numeric, accept the default, which is selected, and deselect Show histogram data values as percent, if necessary.

Selecting both Show histogram data as numeric and Show histogram data as percentage displays data as percentage.

To hide data, deselect Show histogram data values as percent and Show histogram data values as numeric.

Choose OK.

5 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="eAnalysisShowHistogramIn3D"
DisplayName="Show 3D histograms"
TakesEffect="ServerRestart"
DefaultValue="True"

Name="eAnalysisShowHistogramPercent"
DisplayName="Show histogram data values as percent"
TakesEffect="ServerRestart"
DefaultValue="False"

Name="eAnalysisShowHistogramValues"
DisplayName="Show histogram data values as numeric"
TakesEffect="ServerRestart"
DefaultValue="True"
```

Configuring e.Analysis labels

The administrator can configure the Viewing service to affect the output of e.Analysis. The output can contain a label, or not, on the category axis of a line graph. The administrator can also configure the output of a line graph to show data values or not. The data values can appear as percentages or not. By default, the line graph does not show data values.

How to configure e.Analysis labels

1 Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose Labels.

2 In Show line graph subcategory labels, accept the default, not selected, as shown in Figure 6-36. Alternatively, select the option to label the category axis.

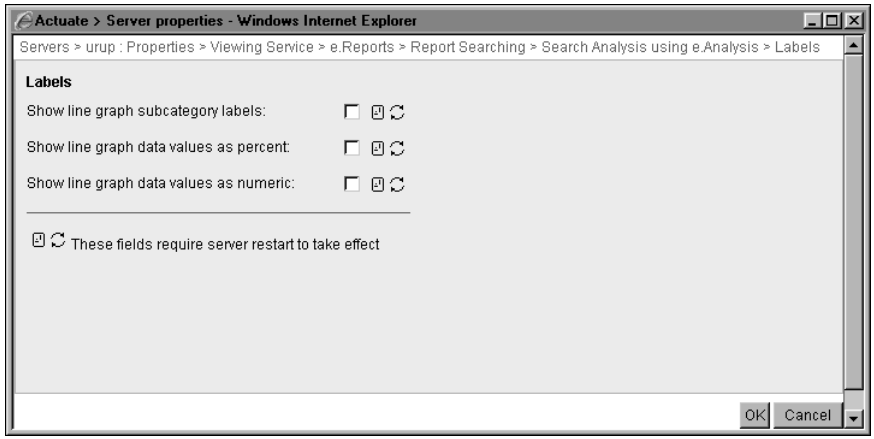

Figure 6-36 Configuring e.Analysis labels

3. In Show line graph data values as percent, accept the default, not selected, or select the option to show data values as percentages.
4. In Show line graph data values as numeric, accept the default, not selected, or select the option to show data values as numeric values.

   If Show line graph data values as percent and Show line graph data values as numeric are not selected, the output of the line graph does not include data values.

   Choose OK.
5. Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="eAnalysisShowLineLabels"
DisplayName="Show line graph subcategory labels"
TakesEffect="ServerRestart"
DefaultValue="False"/>

Name="eAnalysisShowLinePercent"
DisplayName="Show line graph data values as percent"
TakesEffect="ServerRestart"
DefaultValue="False"/>

Name="eAnalysisShowLineValues"
DisplayName="Show line graph data values as numeric"
TakesEffect="ServerRestart"
DefaultValue="False"
```

Configuring e.Analysis pie charts

The administrator can control the number of slices in pie chart output by grouping excess small slices. The administrator configures a minimum percentage of the total pie chart as a threshold below which slices do not appear independently in the output. e.Analysis combines slices falling below the threshold and displays the group as the Other slice. If all slices are less than the threshold, all slices are displayed.

The administrator can also configure the output of pie charts in two or three dimensions, display or hide category labels, display data as a percentage, or not, and hide data in the chart output.

How to configure e.Analysis pie charts

1. Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose Pie Chart.
2. In Combine pie chart subcategories that are less than, accept the default, 3, or type another value from 1 through 99, as shown in Figure 6-37.

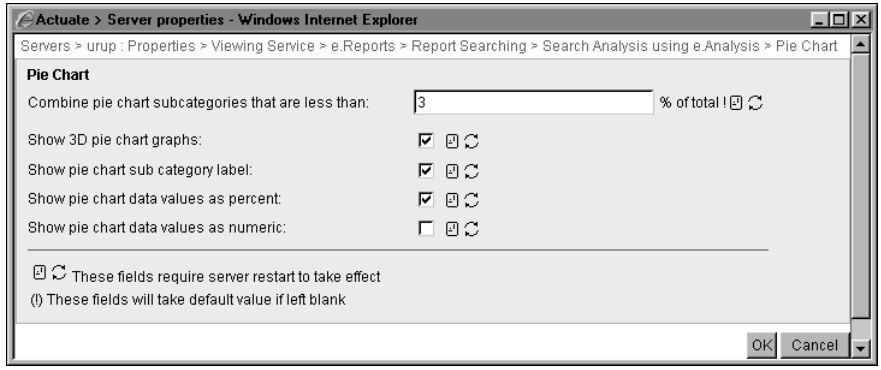

Figure 6-37 Configuring e.Analysis pie charts

3. In Show 3D pie chart graphs, accept the default, selected, to display the pie chart in three dimensions. Alternatively, deselect the option to display the pie chart in two dimensions.
4. In Show pie chart sub category label, accept the default, selected, to display category labels in pie charts. Alternatively, deselect the option to prevent the display of labels in pie charts.
5. In Show pie chart data values as percent, accept the default, selected, or deselect the option.
6. In Show pie chart data values as numeric, accept the default, not selected, or select the option.

   If neither Show pie chart data as percent nor Show pie chart data as numeric is selected, the output of the pie chart does not include data values.

Choose OK.

7 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, default values, and a range:

```
Name="eAnalysisPieChartCombineMinimum"
DisplayName="Combine pie chart subcategories that are less than"
TakesEffect="ServerRestart"
UnitOrFormat="% of total"
DefaultValue="3"
Range="0 99"

Name="eAnalysisShowPieChartIn3D"
DisplayName="Show 3D pie chart graphs"
TakesEffect="ServerRestart"
DefaultValue="True"

Name="eAnalysisShowPieChartLabels"
DisplayName="Show pie chart sub category label"
TakesEffect="ServerRestart"
DefaultValue="True"

Name="eAnalysisShowPieChartPercent"
DisplayName="Show pie chart data values as percent"
TakesEffect="ServerRestart"
DefaultValue="True"

Name="eAnalysisShowPieChartValues"
DisplayName="Show pie chart data values as numeric"
TakesEffect="ServerRestart"
DefaultValue="False"
```

Configuring the e.Analysis table view

The administrator can configure the following options to change e.Analysis table output:

- Auto-resize vertical axis
  Expands rows to fill the height of the data display area. If Auto-resize vertical axis is selected, e.Analysis resizes the rows automatically to accommodate the largest row label. If not selected, e.Analysis truncates long row label text using an ellipsis (...).

- Disable measures total
  Disables the totals, root nodes measures, of categories, which are dimensions.

- Horizontal axis color
  Using the red, green, and blue (RGB) values in the range 0 through 255, sets the color for the columns headers.
- Show hierarchy of column sub categories
  Not implemented.
- Show column totals
  Displays shaded rows of column totals.
- Show leading column totals
  Displays leading shaded column totals. If Show leading column totals is not selected, trailing shaded column totals display. Show column totals must be selected.
- Show hierarchy of row sub categories
  Not implemented.
- Show row totals
  Displays the shaded rows of totals.
- Show leading column totals
  Displays shaded totals above the child categories. If not selected, displays trailing row totals.
- Show drill controls
  Displays the expand and collapse buttons on labels.
- Show columns with no data
- Show rows with no data
- Show grid lines
  Displays grid lines in tables, histograms and line graphs.
- Show columns with all zeros
  Displays columns with zero data.
- Show rows with all zeros
  Displays rows with zero data.
- Sort dimension
  Sorts dimensions in ascending, descending, or false order.
- Vertical axis color
  Using the red, green, and blue (RGB) values in the range 0 through 255, sets the color for the row headers.

How to configure the e.Analysis table view

1 Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose Table View.

2 In Table View, select the option to enable the feature, as shown in Figure 6-38. Deselect the option to disable the feature.

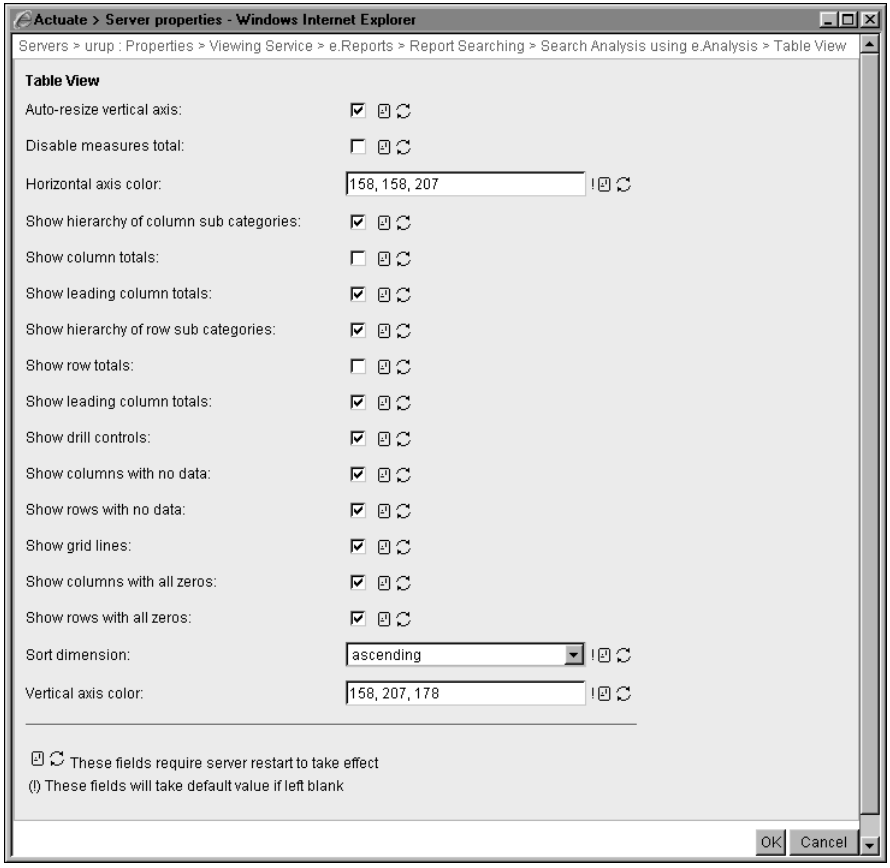

Figure 6-38 Configuring the e.Analysis table view

Choose OK.

3 Restart iServer.

The following snippets from the default acmetadescription.xml include the acmetadescription.xml parameter names, the corresponding display names of the Configuration Console properties, when property changes take effect, and default values:

```
Name="eAnalysisAutoResizeVerticalAxis"
DisplayName="Auto-resize vertical axis"
```

```
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisDisableMeasuresTotal"
DisplayName="Disable measures total"
TakesEffect="ServerRestart"
DefaultValue="False"/>

Name="eAnalysisHorizontalAxisColor"
DisplayName="Horizontal axis color"
TakesEffect="ServerRestart"
DefaultValue="158, 158, 207"/>

Name="eAnalysisShowColumnLevels"
DisplayName="Show hierarchy of column sub categories"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowColumnTotals"
DisplayName="Show column totals"
TakesEffect="ServerRestart"
DefaultValue="False"/>

DisplayName="Show leading column totals"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowRowLevels"
DisplayName="Show hierarchy of row sub categories"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowRowTotals"
DisplayName="Show row totals"
TakesEffect="ServerRestart"
DefaultValue="False"/>

Name="eAnalysisShowRowTotalLeading"
DisplayName="Show leading column totals"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowDrillControls"
DisplayName="Show drill controls"
TakesEffect="ServerRestart"
DefaultValue="True"/>
```

```
Name="eAnalysisShowEmptyColumns"
DisplayName="Show columns with no data"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowEmptyRows"
DisplayName="Show rows with no data"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowGridLines"
DisplayName="Show grid lines"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowZeroColumns"
DisplayName="Show columns with all zeros"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisShowZeroRows"
DisplayName="Show rows with all zeros"
TakesEffect="ServerRestart"
DefaultValue="True"/>

Name="eAnalysisSortDimension"
DisplayName="Sort dimension"
TakesEffect="ServerRestart"
Range="ascending descending false"
DefaultValue="ascending"/>

Name="eAnalysisVerticalAxisColor"
DisplayName="Vertical axis color"
TakesEffect="ServerRestart"
DefaultValue="158, 207, 178"/>
```

Configuring the e.Analysis toolbar

The administrator can configure the e.Analysis toolbar by showing or hiding items.

How to configure the e.Analysis toolbar

1. Expand Viewing Service, e.Reports, Report Searching, and Search Analysis using e.Analysis, as shown in Figure 6-14, then choose Toolbar.
2. In Toolbar, select the items to appear in the toolbar. Alternatively, accept the default, all items deselected, as shown in Figure 6-39.

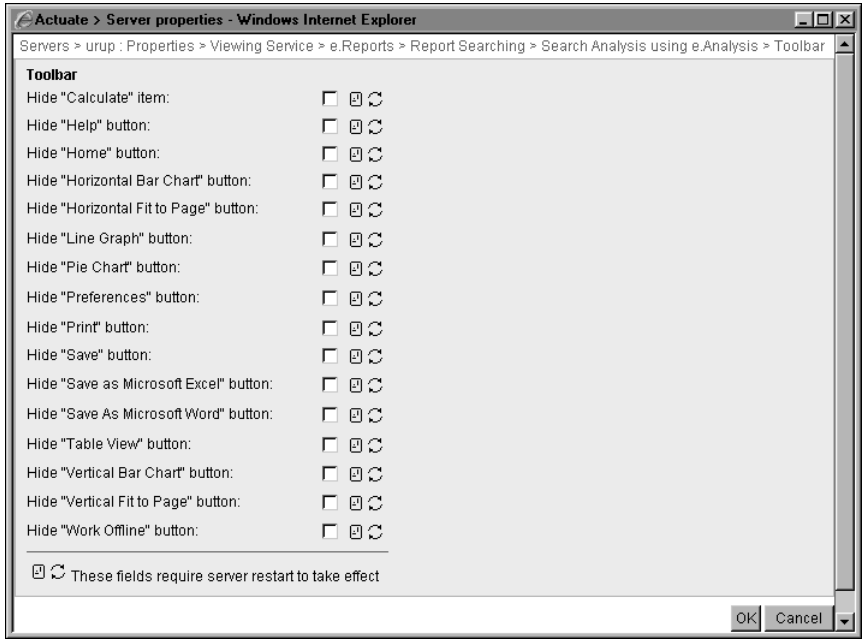

Figure 6-39 Configuring the e.Analysis toolbar

Choose OK.

3 Restart iServer.

Table 6-1 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings and when a property change takes effect.

Table 6-1 e.Analysis toolbar parameters

Property name	Parameter name	Default	Takes effect
Hide "Calculate" item	eAnalysisHideCalculate	False	Server Restart
Hide "Help" button	eAnalysisHideHelp	False	Server Restart
Hide "Home" button	eAnalysisHideHome	False	Server Restart
Hide "Horizontal Bar Chart" button	eAnalysisHideHorizontalBarChart	False	Server Restart
Hide "Horizontal Fit to Page" button	eAnalysisHideHorizontalFitTo Page	False	Server Restart

Table 6-1 e.Analysis toolbar parameters

Property name	Parameter name	Default	Takes effect
Hide "Line Graph" button	eAnalysisHideLineGraph	False	Server Restart
Hide "Pie Chart" button	eAnalysisHidePieChart	False	Server Restart
Hide "Preferences" button	eAnalysisHidePreferences	False	Server Restart
Hide "Print" button	eAnalysisHidePrint	False	Server Restart
Hide "Save" button	eAnalysisHideSave	False	Server Restart
Hide "Save as Microsoft Excel" button	eAnalysisHideSaveAsMSExcel	False	Server Restart
Hide "Save As Microsoft Word" button	eAnalysisHideSaveAsMSWord	False	Server Restart
Hide "Table View" button	eAnalysisHideTableView	False	Server Restart
Hide "Vertical Bar Chart" button	eAnalysisHideVerticalBarChart	False	Server Restart
Hide "Vertical Fit to Page" button	eAnalysisHideVerticalFitToPage	False	Server Restart
Hide "Work Offline" button	eAnalysisHideWorkOffline	False	Server Restart

Configuring the Viewing service for searching DHTML documents

Users analyze the results of a search of Actuate Basic documents in DHTML format using e.Analysis. Users can search only documents that the designer configures for searching by setting the Searchable property to SearchWithIndex or SearchNoIndex. The administrator can configure the Viewing service to manage the formatting, the font of search results, and the search time-out.

Formatting currency and numerical data in search results

The administrator can preserve the currency and numerical data formatting in search results by setting the following option:

Preserve search result format for Currency and Double controls

IEEE 754 defines the double data type, double.

Selecting the option ensures consistency in formatting of the search result data and the Actuate Basic document. Deselecting the option does not preserve formatting.

How to preserve currency and numerical data formatting in search results

1 In Server Configuration Templates—Settings, expand iServer and choose Control Formats in Search Results for e.Reports, as shown in Figure 6-40.

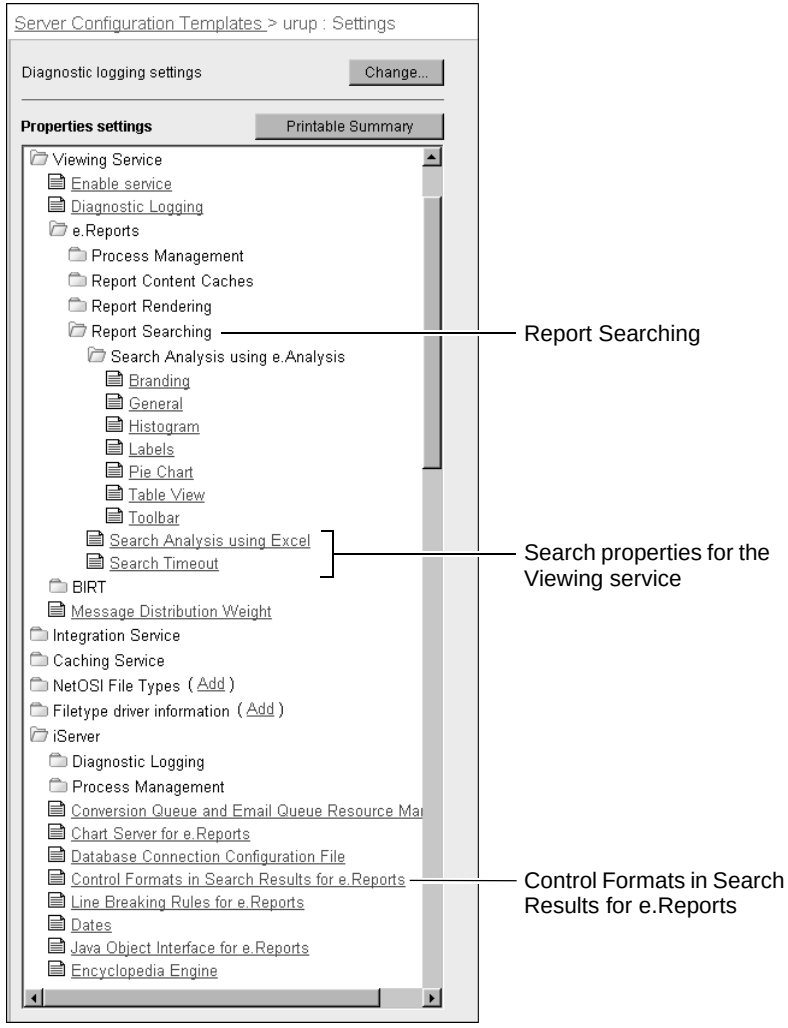

Figure 6-40 Choosing Control Formats in Search Results for e.Reports

2 In Control Formats in Search Results for e.Reports, accept the default to preserve the formatting of currency and doubles, as shown in Figure 6-41. Alternatively, deselect the option to display unformatted data.

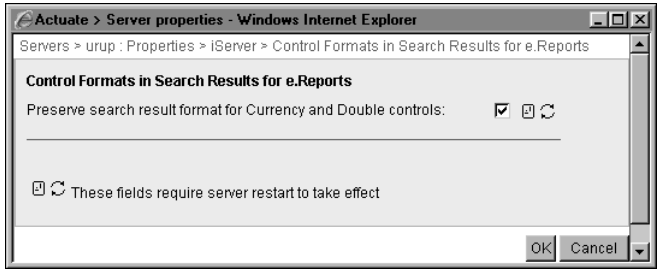

Figure 6-41 Configuring formatting of search results

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, and the default value:

```
Name="PreserveSearchResultFormat"
DisplayName="Preserve search result format for Currency and Double
   controls"
TakesEffect="ServerRestart"
DefaultValue="True"
```

Configuring search properties for the Viewing service

The administrator can access search properties for the Viewing service by expanding Viewing Service, e.Reports, and Report Searching, and choosing the search properties for the Viewing service, as shown in Figure 6-40.

Setting the font for displaying search results

The administrator can specify the font used when saving search results as a Microsoft Excel file by setting the following property:

Font used for search results exported to Excel

A user's system must have the font installed.

How to configure the font for displaying search results

1 Expand Viewing Service, e.Reports, and Report Searching, as shown in Figure 6-40, and choose Search Analysis using Excel.

2 In Font used for search results exported to Excel, type a valid font name for an installed font. For example, type Courier New, as shown in Figure 6-42. Alternatively, accept the default, blank. Arial is the default font.

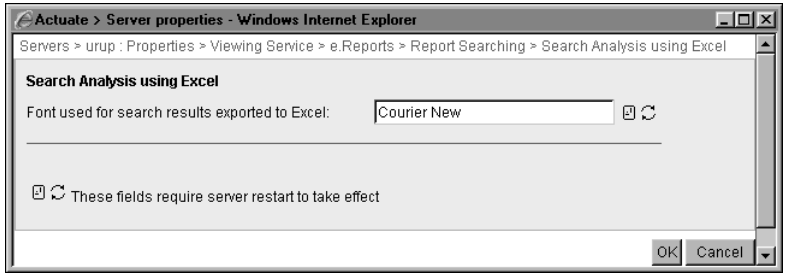

Figure 6-42 Specifying the display setting for search results

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, and the default value:

```
Name="FontUsedForSearchResultToExcel"
DisplayName="Font used for search results exported to Excel"
TakesEffect="ServerRestart"
DefaultValue=""
```

Managing the search time-out

The viewing process, by default, allows a search operation 180 seconds to complete a search of an Actuate Basic document. The administrator can configure the search time-out to change how long iServer spends searching. If searching times out too quickly, increase the search time-out. To search the entire Encyclopedia volume, set the search time-out to zero. When a search times out before finding a match, iServer issues a warning.

How to configure the search time-out

1 Expand Viewing Service, e.Reports, and Report Searching, as shown in Figure 6-40, and choose Search Timeout.

2 In Timeout for search query, accept the default, 180 seconds, as shown in Figure 6-43. Alternatively, type another value.

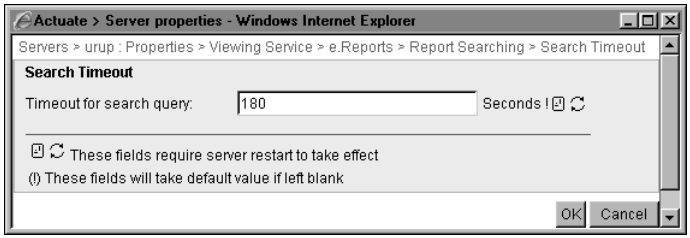

Figure 6-43 Specifying the search time-out

Choose OK.

3 Restart iServer.

The following snippet from the default acmetadescription.xml includes the acmetadescription.xml parameter name, the corresponding display name of the Configuration Console property, when a property change takes effect, and the default value:

```
Name="SearchTimeout"
DisplayName="Timeout for search query"
TakesEffect="ServerRestart"
UnitOrFormat='Seconds'
DefaultValue="180"
```

Chapter

7

Configuring the Factory service

This chapter contains the following topics:

- About the Factory service
- Configuring the Factory service for general use
- Configuring the Factory service for Actuate Basic documents
- Configuring the Factory service for BIRT documents and spreadsheets

About the Factory service

The Factory service provides the basic engine for running a design, and printing a document on the server-side. An asynchronous Factory generates scheduled documents or queries. A synchronous Factory generates temporary documents. To generate temporary documents, iServer must enable both the Factory and View services.

In Server Configuration Templates—Settings, Factory service properties that the administrator configures include properties for Actuate Basic, BIRT, and spreadsheet designs and documents, as shown in Figure 7-1.

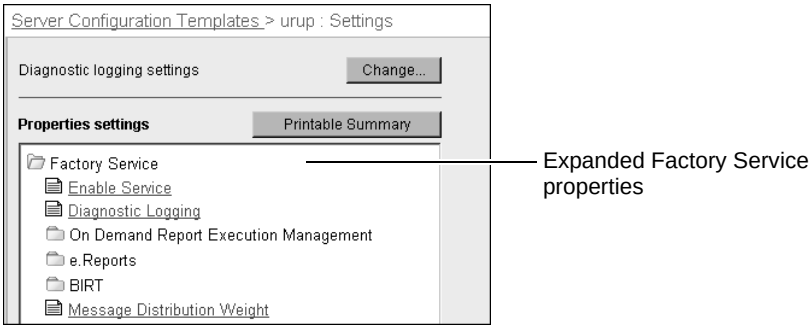

Figure 7-1 Configuring the Factory service in Settings

About setting Factory service properties in iServer Release 11

The properties that appeared in Servers—Properties—Factory Service in previous iServer releases appear in Server Configuration Templates—Settings in Release 11. In Release 11, set Factory service properties in Server Configuration Templates—Settings as follows:

- To set transient report storage properties, choose Factory Service—On Demand Report Execution Management—Transient Report Management.
- To set synchronous report execution properties, choose Factory Service—On Demand Report Execution Management—On Demand Execution Queue.
- To set cluster load balancing properties, choose Factory Service—Message Distribution Weight.
- To set e.Reports cache settings, choose Factory Service—e.Reports—Report Generation Caches—Report Executable Cache.
- To set Maximum number of factories for a resource group, choose Resource Groups, then choose <resource group>—Properties—Template Assignments.

Enabling the Factory service

The administrator can enable or disable the Factory service from Server Configuration Templates—Settings, or by setting the EnableGenerationService parameter in AC_DATA_HOME/config/11SP<service pack number>/acserverconfig.xml to true or false.

How to enable the Factory service

1. Expand Factory Service and choose Enable Service, as shown in Figure 7-1.
2. For Enable factory service, accept the default value, which is selected, as shown in Figure 7-2.

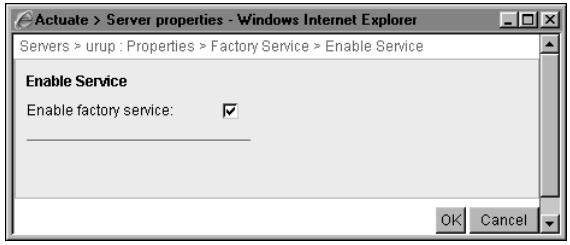

Figure 7-2 Enabling or disabling the Factory service

About diagnostic logging

The administrator can configure diagnostic logging by expanding Factory Service, and choosing Diagnostic Logging. For more information, see Chapter 3, "Using diagnostic, usage, and error logging."

Configuring the Factory service for general use

The following sections describe Factory service configuration tasks for general use:

- Configuring the transient document cache
- Configuring the number of entries in the transient document cache
- Configuring the Factory service for synchronous jobs

Figure 7-3 shows the selections the administrator makes in Server Configuration Templates—Settings to configure properties for general use.

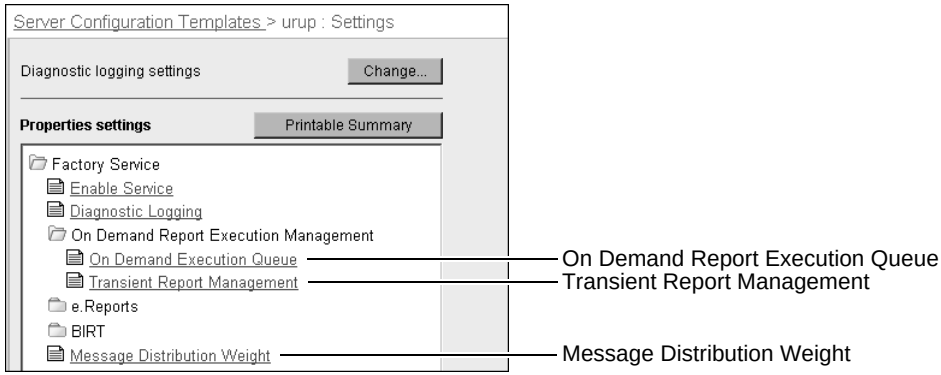

Figure 7-3 Configuring synchronous and transient report properties

Configuring the transient document cache

When a user requests the Factory service to generate a temporary document, iServer stores the document in an external, disk-based cache. By default, after 30 minutes, iServer deletes the file. The administrator can set the size of the cache for temporary documents, location of the cache, and cache time-out.

Setting the transient document cache location

By default, the Factory uses the following directory location for the cache:

`AC_DATA_HOME/server/tmp`

The administrator can specify a different path, locally or on the network. iServer must have access to the location. Actuate recommends keeping /tmp on a local disk drive. Accessing the /tmp directory over the network can impede iServer performance.

Setting the transient document cache size

The transient document cache is disk-based. By setting the cache size and time-out, the administrator controls how much disk space the Factory uses to cache temporary documents. When the cache reaches the limit, iServer sends a message to users indicating a storage problem. Increase the size of the cache if users receive this message. Decrease the size of the cache to conserve disk space. Setting the cache size to 0 permits only one document at a time in the cache.

Setting the transient document time-out

The administrator can control how long documents remain in the cache and on the disk by setting the transient document time-out. The document remains in the cache until the time expires, then iServer clears the cache.

When configuring the time-out, consider the time required for document generation and viewing by users. For example, to generate and view one-page

documents, try a time-out value of 10 minutes. To estimate the document generation time, add the values of MaxSyncJobRuntime and SyncJobQueueWait.

Configuring the number of entries in the transient document cache

The administrator can limit how many temporary documents the Factory caches when users generate documents. Maximum memory cache entries for transient reports is the upper limit on how many temporary documents the cache stores. When the cache reaches the limit, iServer sends a message to the user indicating a storage problem.

If users generate too many temporary documents too quickly, a message about failure to register the document appears. To correct the problem, increase the value of Maximum memory cache entries for transient reports. Specify a value larger than the typical maximum number of documents generated within the transient document time-out period. Increasing the time-out and Maximum memory cache entries for transient reports can increase iServer memory usage. When changing value of Maximum memory cache entries for transient reports, consider adjusting the Disk cache size for transient reports property.

How to configure the transient document cache

1. In Server Configuration Templates—Settings, expand Factory Service. Expand On Demand Report Execution Management, and choose Transient Report Management, as shown in Figure 7-3.
2. In Location of disk cache for transient reports, accept the default, or type another path, as shown in Figure 7-4.

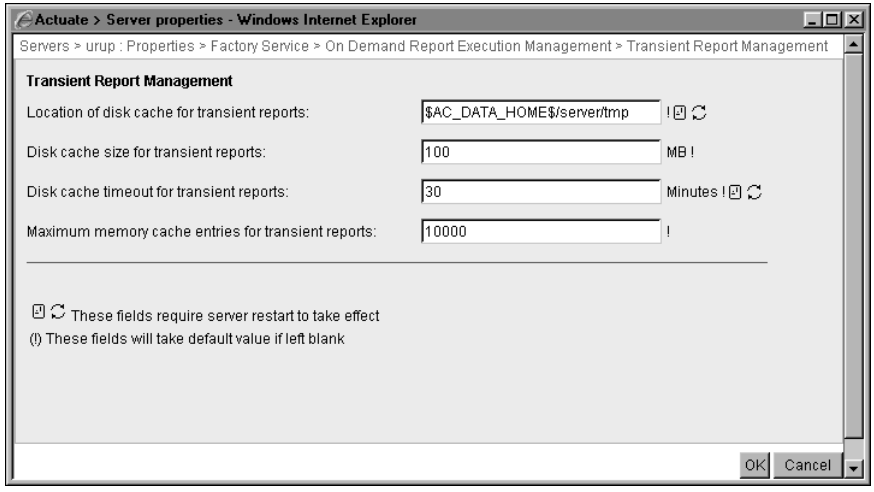

Figure 7-4 Changing the transient document cache size

3 In Disk cache size for transient reports, accept the default, 100 MB. Alternatively, type another value in megabytes.

4 In Disk cache timeout for transient reports, accept the default, 30 minutes. Alternatively, increase the value to give users more time to complete their work. Decrease the value to free disk space sooner.

5 In Maximum memory cache entries for transient reports, accept the default, 10000, as shown in Figure 7-4. Alternatively, type a value greater than 0 to change the number of entries permitted in the transient document cache. Setting the value to 0 prevents the generation of transient documents.

Choose OK.

6 Restart iServer if you change the cache location or time-out values.

Table 7-1 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, which indicate default settings, ranges, and when a property change takes effect.

Table 7-1 Transient document storage parameters

Property name	Parameter name	Default	Range	Takes effect
Disk cache size for transient reports	TransientReportCacheSize	100 MB	0 – 99999	Immediate
Location of disk cache for transient reports	TransientReportCache Location			Server Restart
Disk cache timeout for transient reports	TransientReportTimeOut	30 Minutes	1 – 1440	Server Restart
Maximum memory cache entries for transient reports	TransientStoreMaxCache Entries	10000		Immediate

Configuring the Factory service for synchronous jobs

A user best practice is to use a schedule to generate large documents, but not necessarily short ones. The administrator can configure synchronous job settings to prevent problems that can occur when users generate large documents unscheduled. The Factory generates unscheduled documents synchronously. A synchronous job is the Factory process that generates a temporary document.

Configuring the maximum run time

Max synchronous job runtime limits the time a Factory can take to generate a document. Such a limit can prevent the generation of a huge unscheduled document from dominating Factory resources and degrading the response time of iServer. If an organization has mostly short documents, decreasing the value of the maximum execution time for on-demand execution requests can improve performance. If an organization has a high ratio of system resources to users, increasing the value of the maximum execution time for on-demand execution requests permits users to generate large, unscheduled documents.

Setting the value of Maximum execution time for on demand execution requests too high can delay generation of small, unscheduled documents, which can cause requests to back up in the queue and time out, and users to experience delays. Setting the value to 0 prevents the generation of any unscheduled documents.

Configuring the queue size

Each synchronous resource group has a job queue. When Factories are busy, a request for a Factory service to generate a document waits in the queue. The administrator sets the number of requests that the queue can hold using Job queue size for synchronous reports.

If a user receives a message that the synchronous job queue is full, consider increasing the queue size or resources, such as CPU power. Setting the size of the queue too large can cause the accumulation of too many requests in the queue. A request can time out in the queue.

Configuring the queue time-out

Job queue time-out for transient reports is the period of time, in seconds, that a request for an unscheduled document remains in the queue. Requests wait in the queue for 600 seconds, or 10 minutes, by default, then the request times out and fails. The user who requested the document receives a message that the job expired in the queue. The administrator can change the queue time-out by setting Job queue time-out for transient reports.

How to configure the Factory service for synchronous jobs

1. In Server Configuration Templates—Settings, expand Factory Service. Expand On Demand Report Execution Management, and choose On Demand Execution Queue, as shown in Figure 7-3.
2. In Maximum execution time for on demand execution requests, accept the default, 300 seconds, or 5 minutes, as shown in Figure 7-5. Alternatively, type the maximum number of seconds a Factory can spend per document.

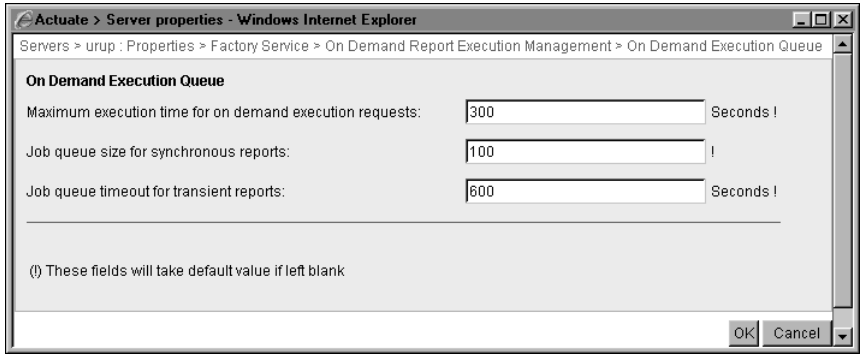

Figure 7-5 Configuring Max synchronous job runtime

3 In Job queue size for synchronous reports, accept the default, 100. Alternatively, type a value greater than 0 to increase or decrease the queue size. A value of 0 causes requests for documents to fail when a Factory is not available to generate the document.

4 In Job queue timeout for transient reports, accept the default, 600. Alternatively, type a value greater than 0 to increase or decrease the wait time. A value of 0 rejects requests to run a document unless a Factory is available immediately.

Table 7-2 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings, ranges, and when a property change takes effect.

Table 7-2 Synchronous document execution parameters

Property name	Parameter name	Default	Range	Takes effect
Maximum execution time for on demand execution requests	MaxSyncJobRuntime	300 Seconds	0 – 86400	Immediate
Job queue size for synchronous reports	SyncJobQueueSize	100	0 – 99999	Immediate
Job queue timeout for transient reports	SyncJobQueueWait	600 Seconds	0 – 999	Immediate

Configuring the message distribution weight of a node

By setting the Message Distribution Weight property, the administrator can assign priority to nodes for synchronous Factory service. By default, the number of Factories, compared to other nodes in the cluster, determines the weight of a node. The number of Factories in a node usually reflects the number of processors. The administrator can change the default weight of a node to take other factors, such as CPU speed, into consideration. The administrator can also

change the weights of nodes if the synchronous Factories of a node are too idle or too busy.

The message distribution weight of a node is relative to the message distribution weights of other nodes. A node having a lower weight than other nodes receives fewer requests for unscheduled documents than the others. The message distribution weight of a node applies to all synchronous resource groups defined for the node.

How to set the Message Distribution Weight property

1. In Server Configuration Templates—Settings, expand Factory Service, then choose Message Distribution Weight, as shown in Figure 7-3.
2. In Weight of this server for load balancing on demand execution requests, as shown in Figure 7-6, accept the default, 100. Alternatively, type a value relative to the weights of other nodes in the cluster. Type a multiple of 100 to increase the weight of a node. Type a value below 100 to decrease the weight of a node.

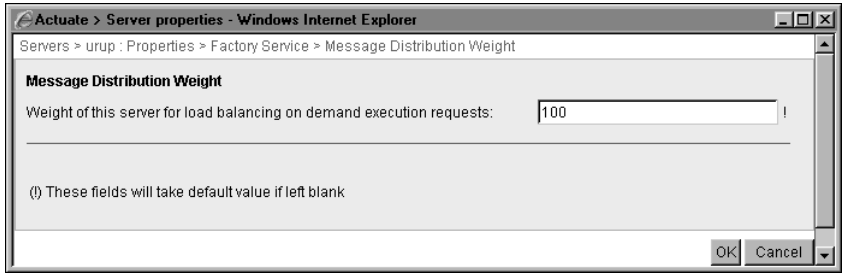

Figure 7-6 Setting the Message Distribution Weight property

Choose OK.

Table 7-3 lists the property name that appears in Configuration Console with the corresponding parameter name in acmetadescription.xml, indicating the default setting and when a property change takes effect.

Table 7-3 Message distribution weight parameter

Property name	Parameter name	Default	Takes effect
Weight of this server for load balancing on demand execution requests	SynchReportingWeight	100	Immediate

Configuring the Factory service for Actuate Basic documents

The administrator can configure the Factory service to cache Actuate Basic designs and to change the locations of Excel output and the Render profiles file. These topics are discussed in the following sections:

- Configuring the Actuate Basic design cache
- Configuring the Excel output directory
- Configuring the render profiles directory
- Shutting down and recycling Factories

Figure 7-7 shows the selections the administrator makes in Server Configuration Templates—Settings to configure properties for Actuate Basic documents.

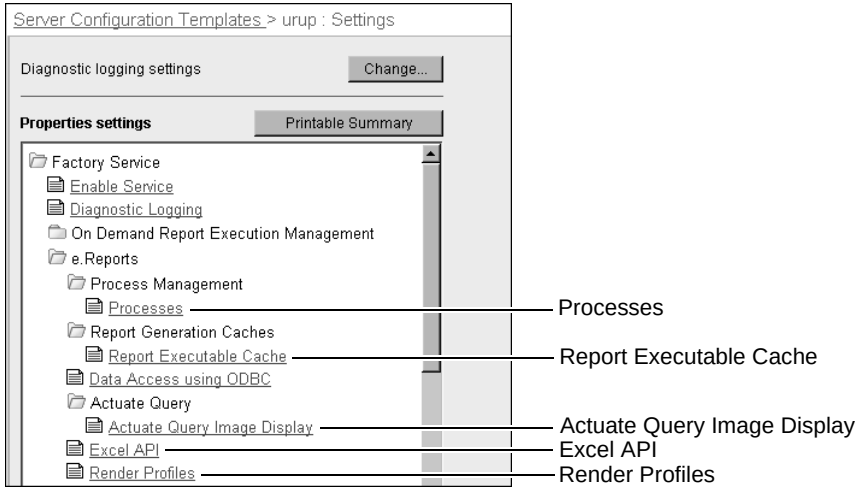

Figure 7-7 Configuring properties for Actuate Basic documents

Configuring the Actuate Basic design cache

Caching Actuate Basic compiled designs can improve the performance of the Factory service by making files available for sharing from the cache instead of reloading files for each request. The administrator can disable the Actuate Basic design cache to conserve memory, but this action can degrade performance. Actuate recommends configuring Maximum number of report executables that can be cached per factory process instead of disabling the cache.

To improve performance, the administrator can configure Maximum number of report executables that can be cached per factory process. Decreasing the value of this property reduces the number of files in the cache and conserves memory.

Typically, an organization that generates documents from very large Actuate Basic designs decreases the value of Maximum number of report executables that can be cached per factory process. Typically, an organization that shares many different Actuate Basic designs concurrently among users increases the value of this property.

How to configure the Actuate Basic design cache

1. In Server Configuration Templates—Settings, expand Factory Service. Expand e.Reports. Expand Report Generation Caches, and choose Report Executable Cache, as shown in Figure 7-7.
2. For Disable caching of report executables, accept the default value, deselected, as shown in Figure 7-8. Alternatively, to disable caching, select the option.

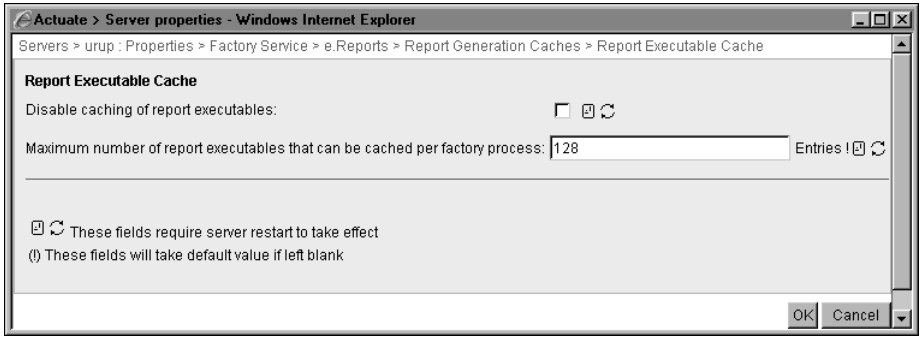

Figure 7-8 Configuring the report executable cache

3. In Maximum number of report executables that can be cached per factory process, accept the default, 128, or type a different value. Typically, values range from a few to 200, depending on the size and number of concurrent requests for documents.

   Choose OK.
4. Restart iServer.

Table 7-4 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings and when a property change takes effect.

Table 7-4 ROX cache parameters

Property name	Parameter name	Default	Takes effect
Disable caching of ROX	DisableProgramManager CacheDisable caching of ROX	False	Server Restart

(continues)

Table 7-4 ROX cache parameters (continued)

Property name	Parameter name	Default	Takes effect
Maximum number of report executables that can be cached per factory process	MaxROXCacheSizePerFactory	128 Entries	Server Restart

Configuring Actuate Query image display

The administrator can specify the following properties for generating Actuate Query output that includes an image:

- Height for the left-corner image
- Width for the left-corner image
- Path to the left-corner image
- Height for the right-corner image
- Width for the right-corner image
- Path to the right-corner image

How to configure the Actuate Query image display

1. In Server Configuration Templates—Settings, expand Factory Service, e.Reports, and Actuate Query, then choose Actuate Query Image Display, as shown in Figure 7-7.
2. For the properties shown in Figure 7-9, accept the default values, or alternatively, specify new values.

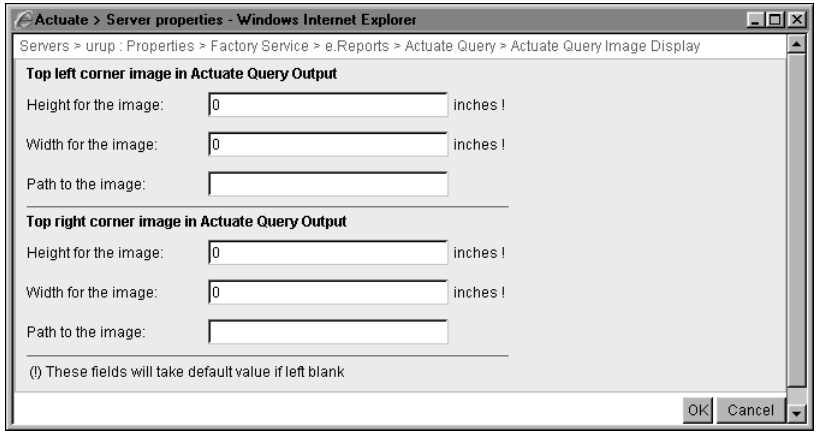

Figure 7-9 Specifying properties for an image in Actuate Query output

Configuring the Excel output directory

By default, the Factory saves the Excel output of an Actuate Basic document in a directory specified by the environment variable or registry entry AC_VIEWSERVER_EXCELOUTPUTDIR. The administrator can configure the Factory to save the Excel output of an Actuate Basic document to a different directory in AC_SERVER_HOME. The account that runs iServer must have write access to the directory. iServer adds $ServerDir as a prefix to the new directory name.

How to configure the Excel output directory

1. In Server Configuration Templates—Settings, expand Factory Service. Choose e.Reports. Choose Excel API, as shown in Figure 7-7.
2. In Excel API, shown in Figure 7-10, type a new path. Do not use a file separator.

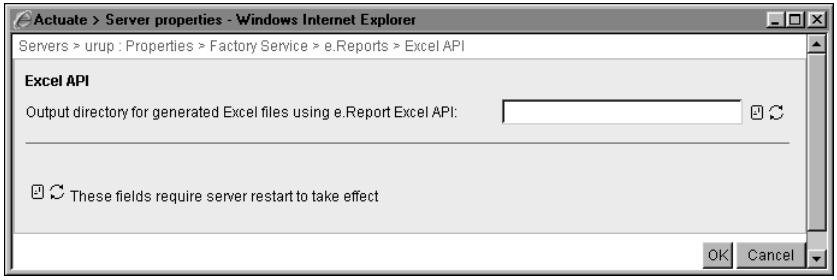

Figure 7-10 Specifying an Excel output directory

Choose OK.

3. Restart iServer.

Table 7-5 lists the property name that appears in Configuration Console with the corresponding parameter name in acmetadescription.xml, indicating when a property change takes effect.

Table 7-5 Excel generation parameter

Property name	Parameter name	Takes effect
Output directory for generated Excel files using e.Report Excel API	OutputDirForRuntimeExcel	Server Restart

Configuring the render profiles directory

The administrator configures rendering primarily for exporting documents to PDF. The administrator configures the Factory service to change the location of

the render profile. iServer runs the job that generates output using a profile in AcRenderProfiles.xml in the following default directory:

```
AC_SERVER_HOME/etc
```

The administrator can configure the Factory service to relocate the directory.

How to configure the render profiles directory

1. In Server Configuration Templates—Settings, expand Factory Service, expand e.Reports, and choose Render Profiles, as shown in Figure 7-7.
2. In Render Profiles URL, type a new path, as shown in Figure 7-11.

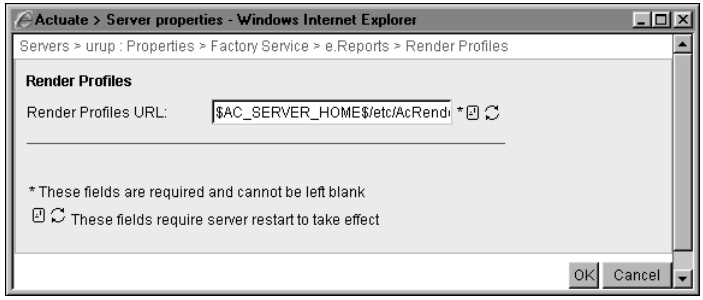

Figure 7-11 Specifying a render profiles URL

Choose OK.

3. Restart iServer.

Table 7-6 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating when a property change takes effect.

Table 7-6 Render profiles parameter

Property name	Parameter name	Takes effect
Render Profiles URL	RenderProfilesURL	Server Restart

Shutting down and recycling Factories

By default, inactive Factories do not shut down. A Factory runs until iServer stops. The administrator can configure Factories to shut down after a period of inactivity. Shutting down Factories frees system resources and can improve performance. Shutting down Factories too quickly, however, can cause a high turnover of Factories that degrades performance. For example, changing:

Time after which idle factory processes are shut down

from the default, 0, to 1 second, shuts down Factories too quickly. The Factories shut down after handling a request if another request does not arrive within 1 second.

The administrator can also recycle Factory processes by limiting the total number of requests that the Factories can handle. After a Factory handles a certain number of requests, iServer shuts down the Factory. Shutting down an existing Factory frees resources, and iServer creates a new Factory when a user requests Factory service. Setting Number of request executions after which the Factory process is recycled to 0 disables recycling. Enable recycling to free system resources, such as memory and handles, which Factories tend to accumulate and hold over time.

How to shut down and recycle Factories

1. In Server Configuration Templates—Settings, expand Factory Service. Expand e.Reports. Expand Process Management, and choose Processes, as shown in Figure 7-7.
2. In Time after which idle factory processes are shut down, accept the default, 0 seconds, as shown in Figure 7-12. Inactive Factories do not shut down. Alternatively, type a number of seconds of inactivity after which to shut down factories.

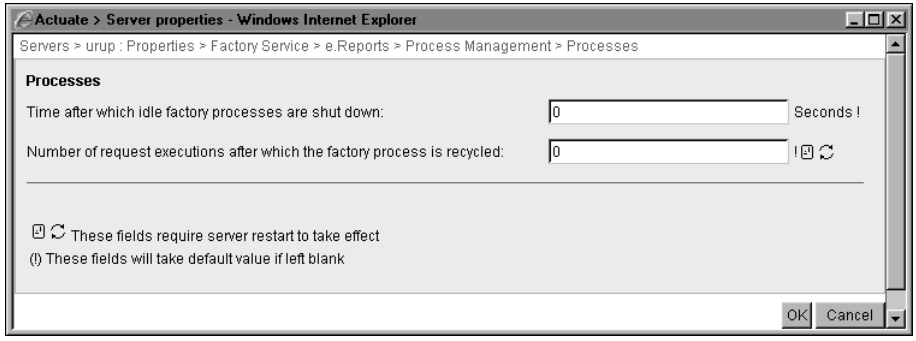

Figure 7-12 Configuring Factory shut-down and recycling

3. In Number of request executions after which the Factory process is recycled, accept the default 0, which disables recycling. Alternatively, type a value greater than 0 to enable recycling.

   Choose OK.
4. If you changed Number of request executions after which the factory process is recycled, restart iServer.

Table 7-7 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings and when a property change takes effect.

Table 7-7 Factory service parameters

Property name	Parameter name	Default	Takes effect
Time after which idle factory processes are shut down	FactoryIdleTimeout	0 Seconds	Immediate
Number of request executions after which the factory process is recycled	ProcessRecycleCount	0	Server Restart

Configuring the Factory service for BIRT documents and spreadsheets

When generating BIRT and spreadsheet documents, the administrator can configure the Factory service for managing resources and optimizing performance. The following sections discuss this topic:

- Configuring the Factory service base port and range of ports
- Recycling Java Factories
- Configuring BIRT caching
- Configuring the maximum rows in a BIRT chart

Figure 7-13 shows the selections the administrator makes in Server Configuration Templates—Settings to configure properties for BIRT documents and spreadsheets.

Configuring the Factory service base port and range of ports

The Factory service engages in Java process communication when a user generates a BIRT document or spreadsheet. In an environment that restricts port usage, the administrator can specify and change the base port for the Factory service and the maximum range of other ports used for SOAP communication.

How to configure the Factory service base port and range of ports

1 In Server Configuration Templates—Settings, expand Factory Service, then expand BIRT. Expand Process Management. Expand Communication, and choose Sockets, as shown in Figure 7-13.

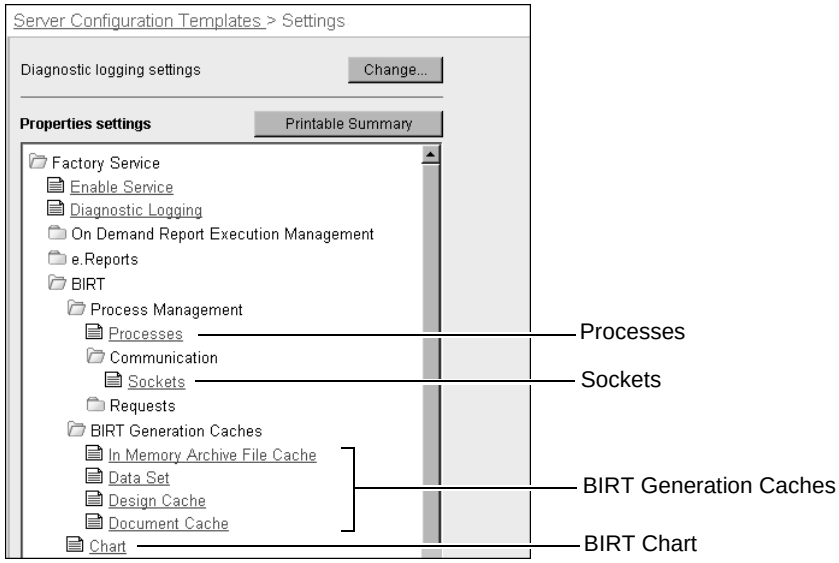

Figure 7-13 Configuring properties for BIRT documents and spreadsheets

2 In Base port number for BIRT factory processes, accept the default, 21500, as shown in Figure 7-14. Alternatively, type another port number.

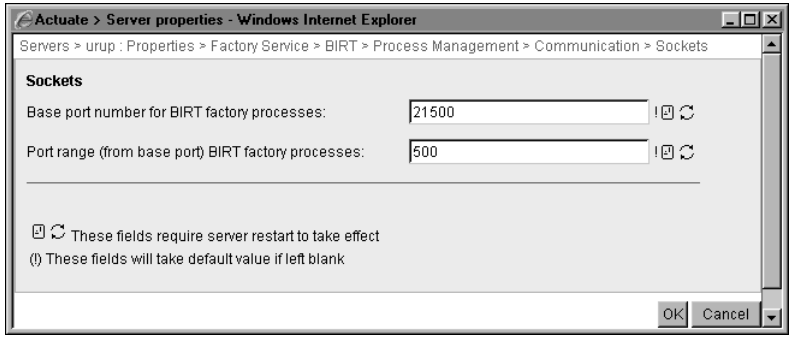

Figure 7-14 Specifying a base port for the Factory

3 In Port range (from base port) BIRT factory processes, accept the default 500, or type another maximum port number.

Choose OK.

4 Restart iServer.

Table 7-8 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings, ranges, and when a property change takes effect.

Table 7-8 Java process communication parameters

Property name	Parameter name	Default	Range	Takes effect
Base port number for BIRT factory processes	SocketBaseForJava Processes	21500	1025 – 65535	Server Restart
Port range (from base port) BIRT factory processes	SocketCountForJava Processes	500	0 – 64510	Server Restart

Recycling Java Factories

By setting Number of Requests Before Recycling Processes to a value greater than 0, the administrator can limit the number of requests for spreadsheet and BIRT documents that a Factory can handle. After the Factory reaches the limit for handling requests, the Factory shuts down, freeing resources. A new Factory emerges. Setting the value of Number of Requests Before Recycling Processes low restarts Factories more frequently than setting the value high. Actuate generally recommends the default value, 0, which disables recycling. In the event of a resource shortage that can occur over time, for example, due to connectivity problems, enable recycling.

How to enable or disable Java Factory recycling

1. In Server Configuration Templates—Settings, expand Factory Service, then expand BIRT. Expand Process Management, then choose Processes, as shown in Figure 7-13.
2. In Number of requests before recycling Java Factory processes, accept the default, 0, as shown in Figure 7-15. This action disables recycling. Alternatively, to enable recycling, type a non-zero value as high as the maximum number of requests a Java Factory handles concurrently.

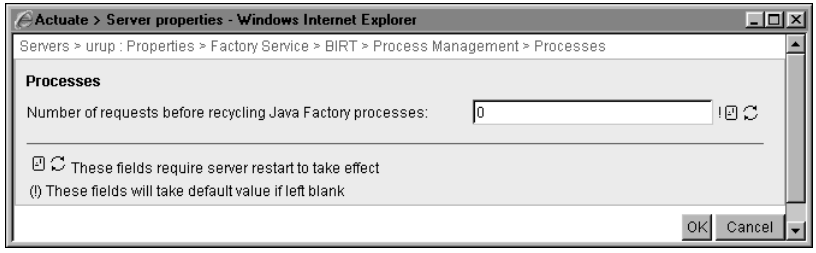

Figure 7-15 Enabling Java Factory recycling

Choose OK.

3. Restart iServer.

Table 7-9 lists the property name that appears in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating the default setting and when a property change takes effect.

Table 7-9 Java process management parameter

Property name	Parameter name	Default	Takes effect
Number of requests before recycling Java Factory processes	JavaProcessRecycleCount	0	Server Restart

Configuring BIRT caching

The administrator can configure the Factory service to change the BIRT data result set buffer, and the design and document caches by choosing Server Configuration Templates—Settings—Factory Service—Java Factory Service—Caching, as shown in Figure 7-13. These topics are discussed in the following sections:

- Configuring the data result set buffer
- Configuring the BIRT design cache
- Configuring the BIRT document cache

Configuring the data result set buffer

A memory-based buffer stores the data result set for a BIRT data object generation query. The Factory sorts, groups, and aggregates data in the buffer. By default this buffer is 128 MB. If the buffer is too small, iServer writes the data result sets to disk. Change this property to tune BIRT document generation performance on iServer under the following conditions:

- The data set of most documents is larger than the default size.
- You have configured sufficient JVM heap size to handle the size of the buffer.

Consider the number of concurrent requests for BIRT documents that the Java Factory can handle when configuring the buffer size. The Factory handles scheduled, asynchronous BIRT document generation requests one at a time.

How to configure the data result set buffer

1. In Server Configuration Templates—Settings, expand Factory Service, BIRT, and BIRT Generation Caches, as shown in Figure 7-13, then choose Data Set.
2. In Maximum result set buffer size for BIRT data object generation query, accept the default buffer size, 128 MB, as shown in Figure 7-16. Alternatively, type a larger buffer size to accommodate larger data result sets.

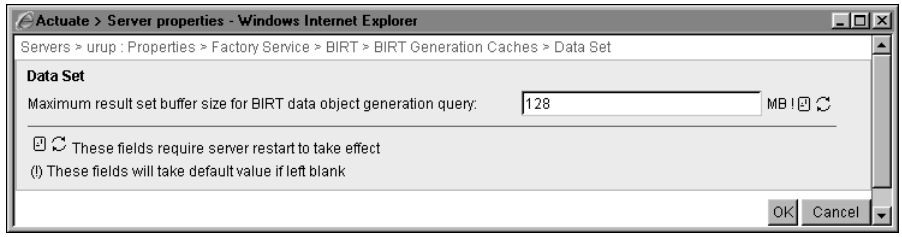

Figure 7-16 Configuring the BIRT report data set buffer

Choose OK.

3 In Server Configuration Templates—Settings, expand Factory Service, BIRT, and BIRT Generation Caches, as shown in Figure 7-13, then choose In Memory Archive File Cache.

4 In Maximum memory limit for each BIRT document file, accept the default, 8192 KB, or specify more space for larger documents, as shown in Figure 7-17.

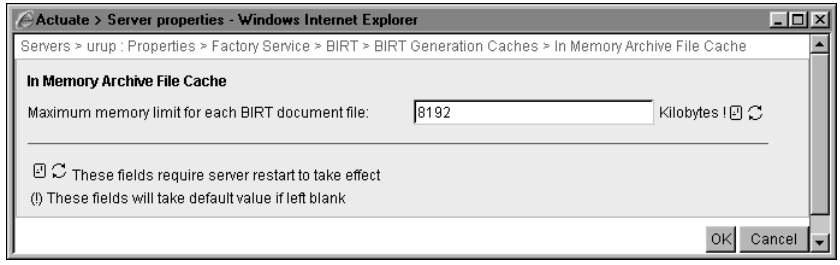

Figure 7-17 Setting maximum memory limit for a BIRT document

Choose OK.

5 Restart iServer.

Table 7-10 lists the property name that appears in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating the default setting, range, and when a property change takes effect.

Table 7-10 BIRT data set buffer parameter

Property name	Parameter name	Default	Range	Takes effect
Maximum result set buffer size for BIRT data object generation query	MaxBIRTDataResult setBufferSize	10 MB	1 – 256	Server Restart
Maximum memory limit for each BIRT document file	MaxMemoryPer Archive	8192 KB		Server Restart

Configuring the BIRT design cache

The administrator can configure how long the Factory keeps BIRT designs in the cache by setting a time-out value. When the time expires, the Factory clears the design from the cache. While a design remains in the cache, users who request the design share the cached file. iServer checks permissions of users to access the design. A cached design has a lifetime equal to the time-out value. Increasing the time-out value keeps designs in cache longer. The higher the time-out value, the more likely users are to access a stale design. The lower the time-out value, the sooner the Factory clears the cache.

The administrator can also configure the capacity of the cache by setting the maximum number of entries allowed in the cache. When the cache reaches capacity, the Factory stops caching designs.

How to configure the BIRT design cache

1. In Server Configuration Templates—Settings, expand Factory Service, BIRT, and BIRT Generation Caches, as shown in Figure 7-13, then choose Design Cache.
2. In Cache timeout for BIRT designs, accept the default, 1800 seconds, or 30 minutes, as shown in Figure 7-18. Alternatively, type a new value in seconds.

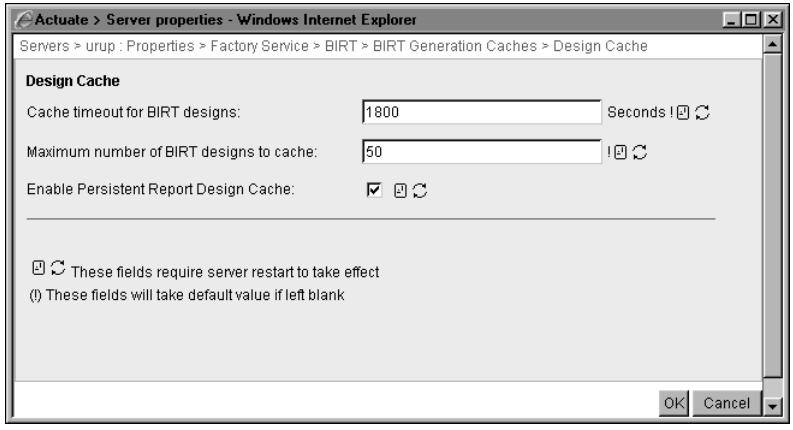

Figure 7-18 Configuring the time-out for the BIRT design cache

3. In Maximum number of BIRT designs to cache, accept the default, 50, or type a new value.
4. For Enable Persistent Report Design Cache, accept the default value of selected. Alternatively, disable the cache by deselecting this option.

   Choose OK.
5. Restart iServer.

Table 7-11 lists the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings and when a property change takes effect.

Table 7-11 BIRT report design cache parameters

Property name	Parameter name	Default	Takes effect
Cache timeout for BIRT designs	BIRTReportDesignCache Timeout	1800 Seconds	Server Restart
Maximum number of BIRT designs to cache	BIRTReportDesignCache TotalNumberOfEntries	50	Server Restart
Enable Persistent Report Design Cache	EnablePersistentDesignCache	True	Server Restart

Configuring the BIRT document cache

By default, iServer caches a BIRT document, including access privileges. Caching benefits users who access the document concurrently. Users who request access to the same document share the cached document if they have the required privileges. Performance can improve because iServer does not have to repeatedly load the document. Generally, the fewer number of documents iServer needs to load, the better the response time. iServer caches BIRT documents in the BIRT document in-memory archive cache. To access the cache, iServer creates a handle. If you enable the BIRT document cache, iServer caches this handle in the BIRT document cache. If you do not enable the BIRT document cache, iServer creates a new handle every time a user chooses to view a document. Enabling the BIRT document cache results in a faster response time but uses more memory, because iServer maintains the BIRT document cache in memory.

How to disable caching BIRT documents

1. In Server Configuration Templates—Settings, expand Factory Service, BIRT, and BIRT Generation Caches, as shown in Figure 7-13, then choose Document Cache.
2. In Document Cache, accept the default. By default, Enable caching of BIRT document and datamart handles is selected, as shown in Figure 7-19. Alternatively, deselect the option to disable the cache.

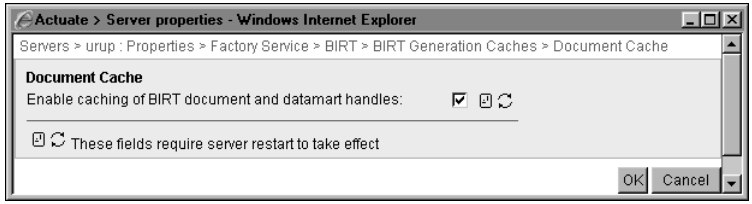

Figure 7-19 Enabling the BIRT document cache

Choose OK.

3 Restart iServer.

Table 7-12 lists the property name that appears in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating the default setting and when a property change takes effect.

Table 7-12 BIRT report document cache parameter

Property name	Parameter name	Default	Takes effect
Enable caching of BIRT document and datamart handles	BIRTReportDocumentCacheEnabled	True	Server Restart

Configuring the maximum rows in a BIRT chart

By default, BIRT charts display all data when rendered. The administrator can limit the number of rows displayed to prevent the rendering of huge charts from causing performance problems. Actuate recommends using the default setting, 0, for Maximum number of rows for a BIRT chart that displays all data when rendered.

How to limit the rows in a BIRT chart

1 In Server Configuration Templates—Settings, expand Factory Service and BIRT, then choose Chart, as shown in Figure 7-13.

2 In Maximum number of rows for generating a chart, accept the default, 0, as shown in Figure 7-20. Alternatively, type a positive integer that represents the maximum number of rows.

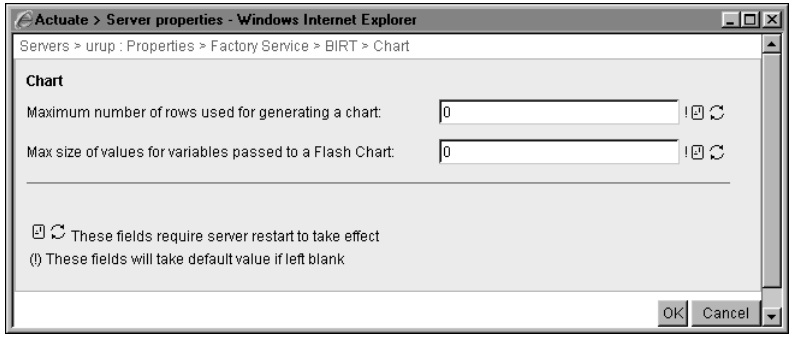

Figure 7-20 Configuring the maximum number of rows in a BIRT chart

3 In Max size of values for variables passed to a Flash Chart, accept the default, 0. Alternatively, type a non-zero value to limit the size of variables used in a Flash chart that appears in a BIRT document.

Choose OK.

4 Restart iServer.

Table 7-13 lists the property name that appears in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating the default setting and when a property change takes effect.

Table 7-13 Chart parameter

Property name	Parameter name	Default	Takes effect
Maximum number of rows for generating a chart	BIRTChartMaxRows	0	Server Restart
Max size of values for variables passed to a Flash Chart	BIRTChartMaxVariable Size	0	Server Restart

Chapter

8

Using resource groups

This chapter contains the following topics:

- Introducing resource groups
- Configuring a resource group
- Adding a resource group
- Using a resource group
- Stopping a resource group
- Deleting a resource group
- Using resource groups programmatically

Introducing resource groups

A resource group controls the Factory processes that iServer uses to run a design and distribute a document. A resource group allocates a set of Factory processes in a stand-alone iServer or in a cluster for executing jobs assigned to the resource group. You choose synchronous or asynchronous resource groups, depending on whether you schedule the design to run.

A design that runs unscheduled runs synchronously, as soon as possible in the foreground. iServer does not store the generated document in the Encyclopedia volume. You can view, navigate, and search the document. You must enable both the Factory and View services to run designs unscheduled.

A scheduled job runs asynchronously in the background. iServer stores job schedules and information about completed jobs in the Encyclopedia volume.

Table 8-1 maps the types of resource groups to user operations in Management Console.

Table 8-1 Types of resource groups mapped to user operations

Default resource group	Resource group type	User operations
Default Sync	Sync	Run an Actuate Basic design unscheduled.
Default Async	Async	Schedule an Actuate Basic design or Actuate Query to run right now, later, on a recurring basis, or when triggered by an event. Print an Actuate Basic document.
Info Object Web Services	Sync	Access Actuate information object data through a web service.
Default BIRT 360	View	Run a BIRT dashboard (.dashboard) or gadget (.gadget) design unscheduled and view the generated document.
Default BIRT Data Analyzer	View	Run a BIRT Cube View (.cubeview) or Data Object Store (.data) design unscheduled and view the generated document.
Default BIRT eSS Factory	Async	Schedule a BIRT Spreadsheet design (.sox) or template (.vtf, .vtx) to run right now, later, on a recurring basis, or when triggered by an event. Print a spreadsheet document.
Default BIRT eSS Online	View	Run a BIRT Spreadsheet design (.sox) or template (.vtf, .vtx) design unscheduled and view the generated document.

Table 8-1 Types of resource groups mapped to user operations

Default resource group	Resource group type	User operations
Default BIRT Factory	Async	Schedule a Java design (.rptdesign) to run right now, later, on a recurring basis, or when triggered by an event. Print a Java document.
Default BIRT Online	View	Run a BIRT design unscheduled and view the generated document.
Default BIRT Studio	View	Used when creating, modifying, and viewing documents using BIRT Studio.

You typically use resource groups for the following purposes:

- To control the load balancing of servers in a cluster
- To prioritize asynchronous jobs
- To specify which node in a cluster runs designs of a particular type
- To improve performance of a BIRT application, for example, by passing an argument to the application to decrease the heap size
- To quickly run a synchronous design using the Default Sync or Default BIRT Online resource group

  When users run a design unscheduled in an Encyclopedia volume that uses multiple resource groups, iServer selects a resource group based on load balancing and on the file type of the design. If iServer cannot find a resource group with an available Factory, the job fails. Using the Default Sync or Default BIRT Online resource group can ensure the availability of a Factory for running a design unscheduled.

Configuring a resource group

You can access resource group properties from the Advanced view by choosing Resource Groups, as shown in Figure 8-1.

By configuring resource group properties, you can control the following operations:

- Prevent iServer from running jobs sent to a resource group.
- Determine which Encyclopedia volumes use the Factories of a resource group.
- Set the maximum number of Factory processes that an iServer can run.
- Specify the file types that Factories allocated to a resource group can run.

- Prioritize jobs for an asynchronous resource group.
- Specify start arguments for Java Runtime Environment (JRE) for running Java designs.

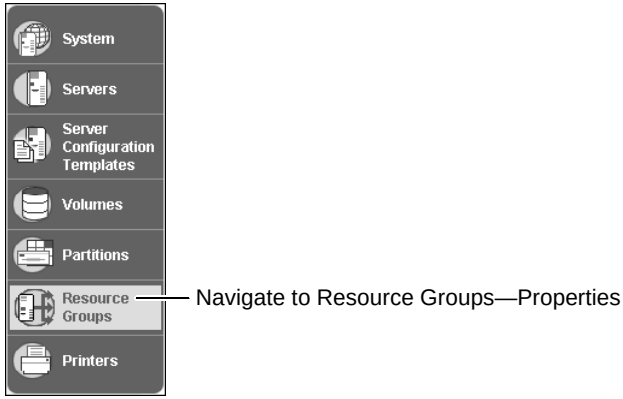

Figure 8-1 Navigating to resource group properties

Allocating Factories for a resource group

Valid values for the Max factory property range from 1 to twice the number of processors on your machine. When setting this property, consider the Factories allocated for other resource groups. Also, consider using FactoryIdleTimeout to make unused Factories available. Increasing the number of Factories impacts performance. Setting the value to 0 disables the resource group for the server. By default, iServer allocates one Factory process to each default resource group during installation.

Limiting Java Factories for a resource group

A Factory process assigned to a resource group stays idle until iServer routes a job to the resource group. If there are no idle Factory processes for the resource group, a new Factory process starts when a job arrives. Using the Min factories property, you can optimize performance by preventing iServer from starting excessive Java Factories to run BIRT designs and spreadsheets.

How to change properties of a resource group

1. From the Advanced view of Configuration Console, choose Resource Groups. Figure 8-2 shows the default resource groups that iServer creates at installation.
2. Point to the arrow next to the resource group name, for example Default Async, and choose Properties, as shown in Figure 8-3.

   Properties—General appears, as shown in Figure 8-4.

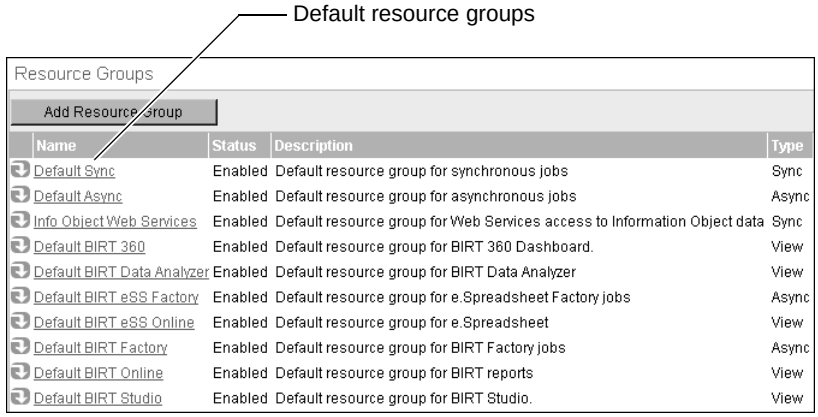

Figure 8-2 System Resource Groups lists resource groups

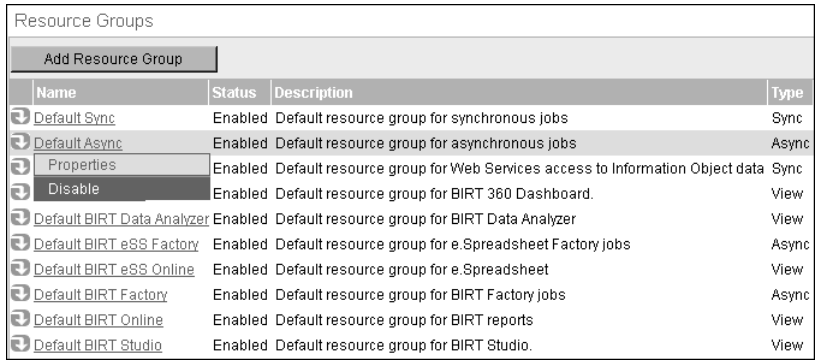

Figure 8-3 Changing resource group properties

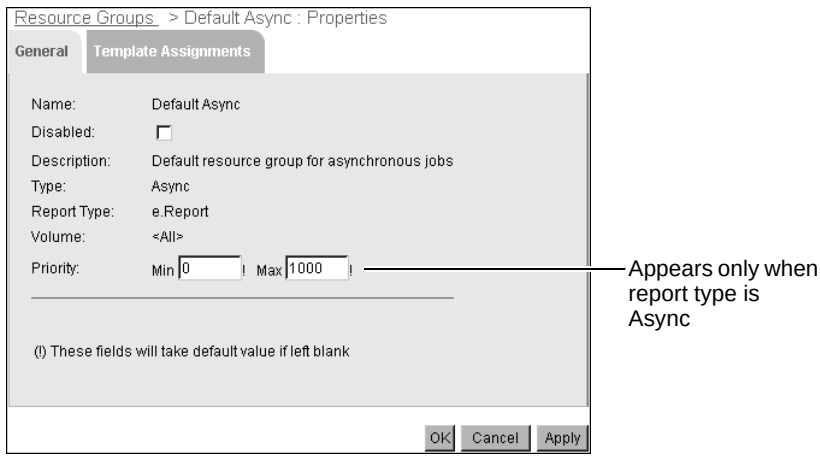

Figure 8-4 General properties of Default Java Async resource group

On General, you can perform the following tasks:

- Select Disabled to prevent iServer from running jobs sent to this resource group.
- All Encyclopedia volumes use the Factories of a default resource group. For a resource group that the administrator creates, select <All>, or select a particular volume to use this resource group.
- For a resource group whose report type is JavaReport, such as Default BIRT Online and Default BIRT Factory, Start Arguments appears. In Start Arguments, accept the default start arguments or change them to suit your Java Runtime Environment.
- For an Async resource group, change the minimum and maximum job priority ranges. Valid values are 0–1,000, where 1,000 is the highest priority. The minimum must be less than the maximum.

On Properties, choose Template Assignments.

3 Template Assignments, shown in Figure 8-5, lists the properties and names of the nodes belonging to the resource group that starts the Factory service.

Figure 8-5 Resource group server assignments

4 On Template Assignments, you can perform the following tasks:

- Select Activate to make that server a member of the resource group so it can use the resource group's Factory processes.

- Select the maximum number of Factory processes to assign to the resource group. For a resource group having a report type of JavaReport, type the minimum number of Factory processes also.
- For a resource group having a report type of JavaReport, enter start arguments for the Java Runtime Environment.

Restart the cluster node or master, or the stand-alone server if you change the file types or start arguments. You can enable or disable resource groups, change activation status, or change the number of Factory processes in a resource group without restarting iServer.

Setting start arguments for the JRE

The Default BIRT Factory resource group has the following start arguments by default:

- Heap limit option
  Specifies the amount of heap the Java process can use. Too large a heap can slow garbage collection because there is more heap to scan. This property affects Java view server memory usage. Actuate sets this option to -Xmx512M to accommodate generating typical BIRT documents. For example, -Xmx256m specifies that the Java process can use 512 MB of heap.
- MaxPermSize
  PermSize is additional heap space, separate from the space the Heap limit option specifies. The heap space that PermSize specifies holds reflective data for the JVM, such as class and method objects. By specifying MaxPermSize without also specifying PermSize, heap size does not increase unless an application needs more heap.
- Headless graphics option
  Includes the Java graphics environment in lieu of a native graphics environment when set to true. For example, -Djava.awt.headless=true specifies including the Java graphics environment.
- Protocol library specification
  For example, Djava.protocol.handler.pkgs=com.actuate.javaserver.protocol specifies the package name in which the Actuate protocol handler class can be found.
- Java server entry point specification
  For example, com.actuate.javaserver.Server specifies the Java server main class.

You can change the start arguments for a Java resource group, as shown in Figure 8-6.

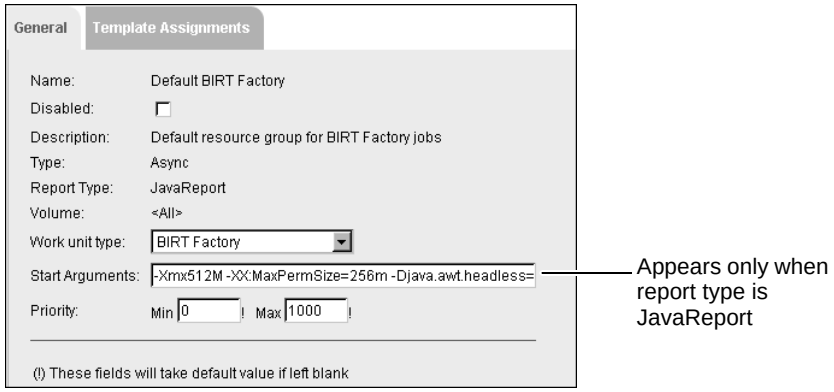

Figure 8-6 General properties of the Default BIRT Factory resource group

Adding a resource group

You create new resource groups using Configuration Console. From the Advanced view, select Resource Groups, then choose Add Resource Group. First you set general properties in Resource Groups—New Resource Group—General.

You can specify the following properties on Resource Groups—New Resource Group—General:

- The name of the resource group.
- Whether to enable or disable the resource group.
- A description of the resource group.
- The type of job that a resource group supports. The types of jobs from which to choose are asynchronous, synchronous, and view.
- The type of executable file that the resource group supports. The types of files from which to choose are e.Report and Java report.
  Examples of e.Report are Actuate Basic and Actuate Query designs. Examples of a Java report are BIRT and spreadsheet designs.
- The Encyclopedia volumes that can use the resource group's Factory processes.
- Start arguments for the JRE. This property displays only when report type is JavaReport.
- Work unit type. The property specifies the type of processing this resource group can perform. For example, generating a BIRT document asynchronously requires the BIRT Factory work unit type. Generating a BIRT document immediately requires the BIRT Online work unit type.

- The job priority range for an asynchronous resource group.
  The synchronous process executes designs as soon as possible and cannot assign priority to a job. If you set the type to Sync or View, iServer does not display Priority.

Figure 8-7 and Figure 8-8 contrast the differences between the general properties of an asynchronous resource group that processes e.reports and the general properties of a synchronous or view type resource group that processes Java reports.

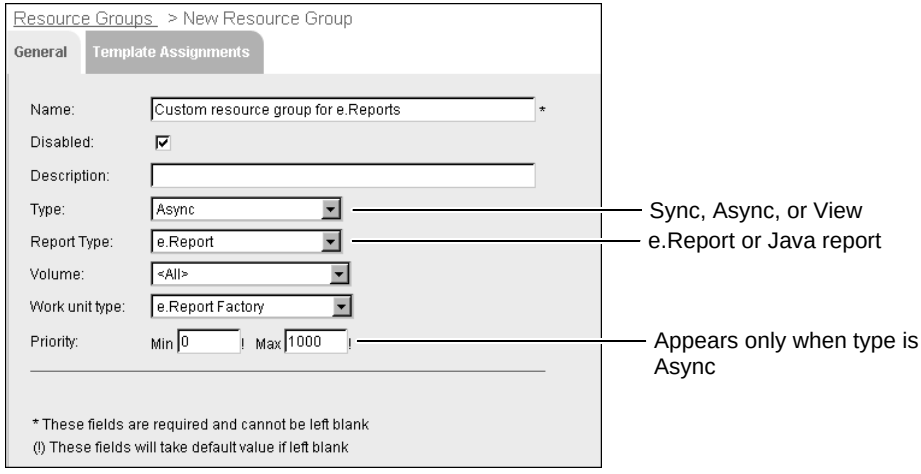

Figure 8-7 New Async resource group for e.reports

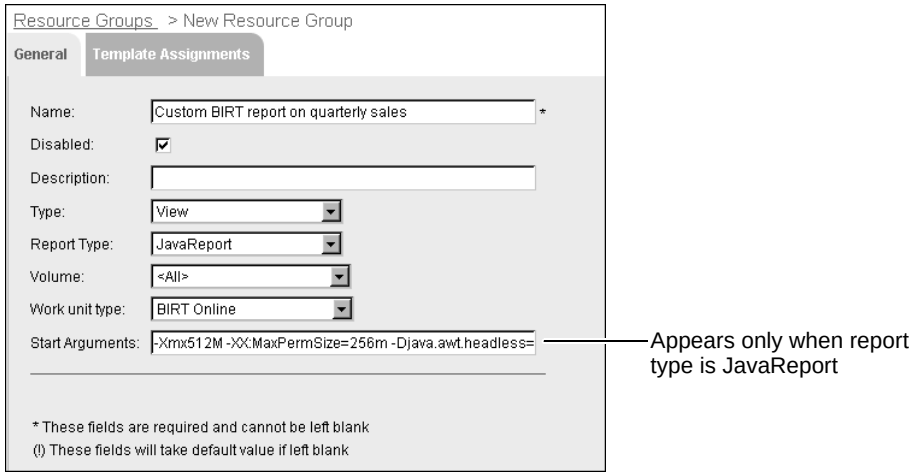

Figure 8-8 View resource group for Java reports

In Template Assignments, you set properties for a new resource group in the same way as when you change properties for an existing resource group.

How to add a resource group

1 From the Advanced view of Configuration Console, choose System Resource Groups.

2 On Resource groups, choose Add resource group.

Resource Groups—New Resource Group—General appears, as shown in Figure 8-9.

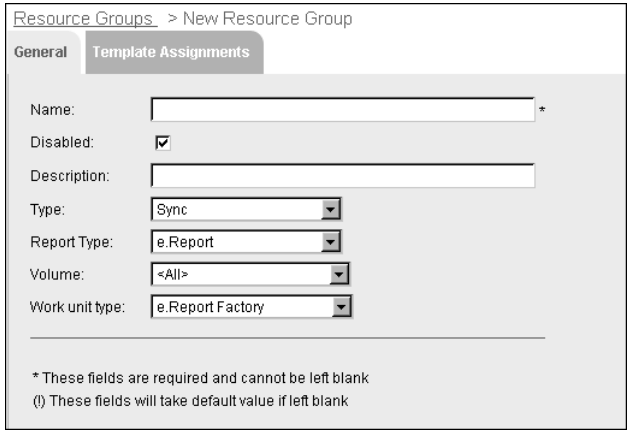

Figure 8-9 Adding a synchronous resource group

3 Supply the following parameter values:

- Type a name for the resource group. Choose the name carefully because you cannot change it later.
- Deselect the Disabled option, so that cluster nodes assigned to the resource group can run jobs that users send to the group. The default status of a new resource group is Disabled.
- Type a description of the resource group.
- Select the type of resource group Async, Sync, or View. You cannot change the type later.
- Select an appropriate report type, either e.Report or JavaReport, for the resource group type.
  - e.Report for Sync or Async
  - Java report for Sync, Async, or View
- To specify that all Encyclopedia volumes can use the Factories of the resource group, select All. Alternatively, select the name of the Encyclopedia volume that can use the Factory processes.
- Select a work unit type.

- If report type is JavaReport, modify Start Arguments if necessary.
- Type a minimum and maximum priority for jobs sent to an asynchronous resource group.

Choose OK.

4 On Resource Groups, choose the resource group you just created.

On Properties, choose Template Assignments.

5 In Template Assignments, change the following settings, as shown in Figure 8-10:

- Select Activate to make that server a member of the resource group so it can use the resource group's Factory processes.
- Select the maximum number of Factory processes to assign to the resource group.
- For resource groups that support Java reports, type the minimum number of Factory processes.
- For resource groups for which the report type is JavaReport, specify Start Arguments to optimize performance.

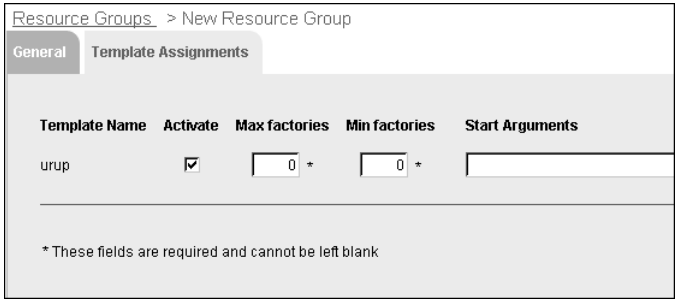

Figure 8-10 Making server assignments to the new resource group

Choose OK.

6 Restart the cluster node or stand-alone server.

Using a resource group

A user can specify a resource group in a job schedule to assign priority to a job. To specify a resource group to handle a job, specify the group as described in this section.

Selecting a resource group for a job

Log in to Management Console and choose Files and Folders. Next, point to the arrow next to the name of a design and choose Schedule. On Schedule—Schedule, select a resource group from Resource Group, as shown in Figure 8-11.

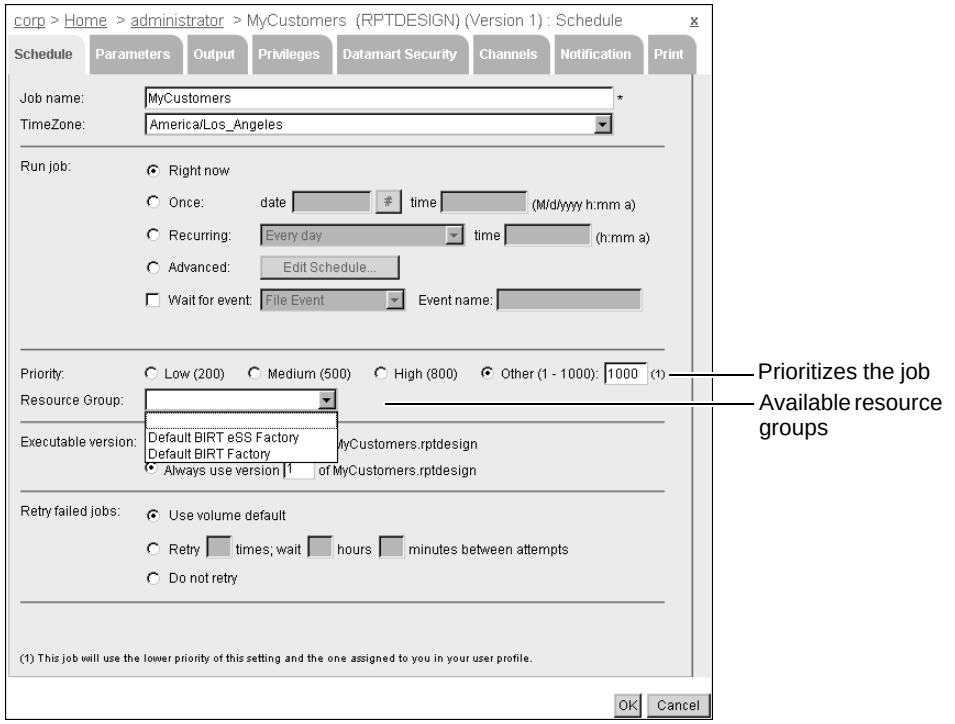

Figure 8-11 Specifying a resource group for a job

Prioritizing a job

You specify the priority that the Factory process gives a job when you create a resource group. iServer tries to match the priority setting in a resource group with the priority setting in the job schedule. The following conditions can occur:

- If no match is found, the job stays in a pending state until you change the priority specifications in the resource group.
- If the only available resource group is disabled, iServer sets the job to pending until you enable an asynchronous resource group for the Encyclopedia volume.
- If multiple resource groups have the same job priority settings and an available Factory process, iServer chooses a resource group that best balances the load.

About Factory and View service requirements

After you create a custom resource group for a cluster, you need to understand the Factory and View service requirements for nodes assigned to the resource group. It is also important to consider the effect that using resource groups has on performance.

You must enable only the View service on iServer to fulfill requests to view persistent documents. Persistent documents are documents that are saved on a volume. You must enable both the View and Factory services on iServer to fulfill all other requests to generate and view documents.

Managing loads with resource groups

You need to understand and avoid the following problems that can affect performance:

- Unequal loads
  In a cluster, iServer uses load-balancing mechanisms to distribute jobs among the cluster nodes. The load-balancing mechanisms attempt to maximize performance. Creating custom resource groups can restrict the capability to maximize performance and adversely affect the document-generation performance of a cluster.
- Configuration problems
  If a cluster node is a member of a resource group that is not configured to run all the executable design file types, designs can fail. For example, if a design requires access to a database, and the database driver is not installed on a node, the design fails when that node attempts to run the design.

Understanding the Java View service

The Java View service is capable of handling multiple on-demand design execution and document viewing requests concurrently. By default, the service can handle 10 requests concurrently and can queue up to 1,000 requests.

Initially, configure a Java View resource group for a two-CPU system, to use just a few factories, and then tune the system under load.

Stopping a resource group

Stop a resource group from running jobs by performing one of the following actions:

- Disabling the resource group

- Setting the resource group's maximum number of Factory processes to 0 for all nodes that belong to the resource group
- Removing active membership of nodes

Disabling an asynchronous resource group is the same as setting the maximum number of Factory processes to 0. If all resource groups are disabled, jobs sent to the disabled asynchronous resource groups go into a pending state until a resource group becomes available.

If you disable a synchronous resource group, it processes jobs that are currently being executed and those that are waiting. If you set the maximum number of Factory processes to 0 on any of the nodes, jobs wait to be executed until they time out.

Removing all active memberships from a resource group is the same as setting the maximum number of Factory processes to 0 on all nodes in a cluster.

If you remove a resource group from an Encyclopedia volume, you must assign a resource group with available Factory processes to the Encyclopedia volume. Otherwise, you cannot run a job. For example, if you change a resource group Encyclopedia volume assignment from volume1 to volume2, you must ensure that another resource group can handle the jobs volume users create in volume1. If you remove an Encyclopedia volume assigned to a resource group, iServer changes the Encyclopedia volume's resource group assignment to All volumes and disables the resource group.

Deleting a resource group

Delete a resource group by pointing to the arrow next to the resource group and choosing Delete, as shown in Figure 8-12. You cannot delete a default resource group.

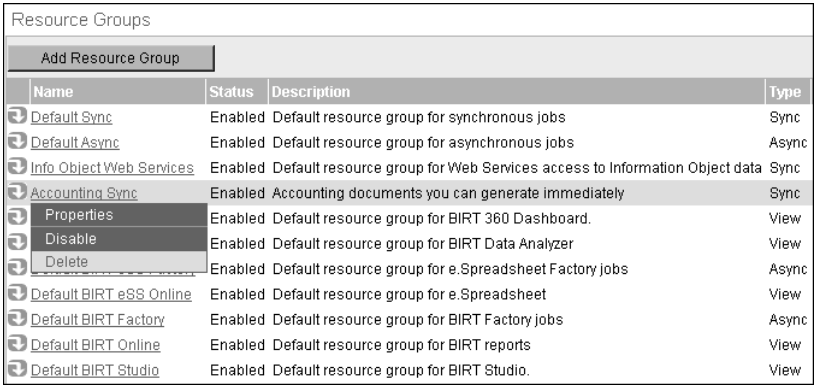

Figure 8-12 Deleting a resource group

Deleting a resource group produces the following results, depending on the state of related jobs:

- If a job is already running on a Factory assigned to a resource group that you delete, the job completes.
- If a scheduled job is assigned to a deleted resource group, the job either fails when iServer runs the job, or it remains in a pending state, depending on the job status at the time of deletion.

  You can delete a pending job on Jobs—Pending.
- If an unscheduled job is assigned to the deleted resource group, the job fails when iServer runs the job.

  If a job is running on a Factory assigned to a resource group that you delete, the job completes.

Using resource groups programmatically

Developers can write applications using resource groups with Actuate Information Delivery API (IDAPI). Using IDAPI, you can:

- Send a job directly to a resource group, bypassing the priority settings for asynchronous jobs and the Encyclopedia volume settings for synchronous jobs.
- Change the resource group configuration. For example, you can enable or disable resource group membership or change the number of Factory processes in a resource group.

Using Actuate Information Delivery API (IDAPI) to direct jobs to a specific resource group bypasses the cluster's load-balancing mechanisms. As a result, applications that use IDAPI to run designs can adversely affect the performance of the cluster.

Chapter

9

Clustering

This chapter contains the following topics:

- About a cluster configuration
- About distributing services to a cluster
- About the configuration home directory
- About the primary configuration files
- Creating a cluster
- Adding and modifying server templates
- Adding a node to a cluster
- Starting and stopping a node
- Removing a node from a cluster
- Managing a cluster
- Handling file system failure
- Configuring the cluster administrator e-mail account
- Managing console configurations and load balancing

About a cluster configuration

In the simplest configuration, an iServer cluster consists of two iServer nodes. A node is a single machine in a cluster. You can add nodes to a cluster to scale iServer System to your requirements. You install iServer on each node in a cluster. The node gets its configuration from a template in acserverconfig.xml, which is located in a shared configuration home directory. After the node is configured, it joins the cluster.

Figure 9-1 shows the relationships between a cluster and its Encyclopedia volumes, application servers, and database servers.

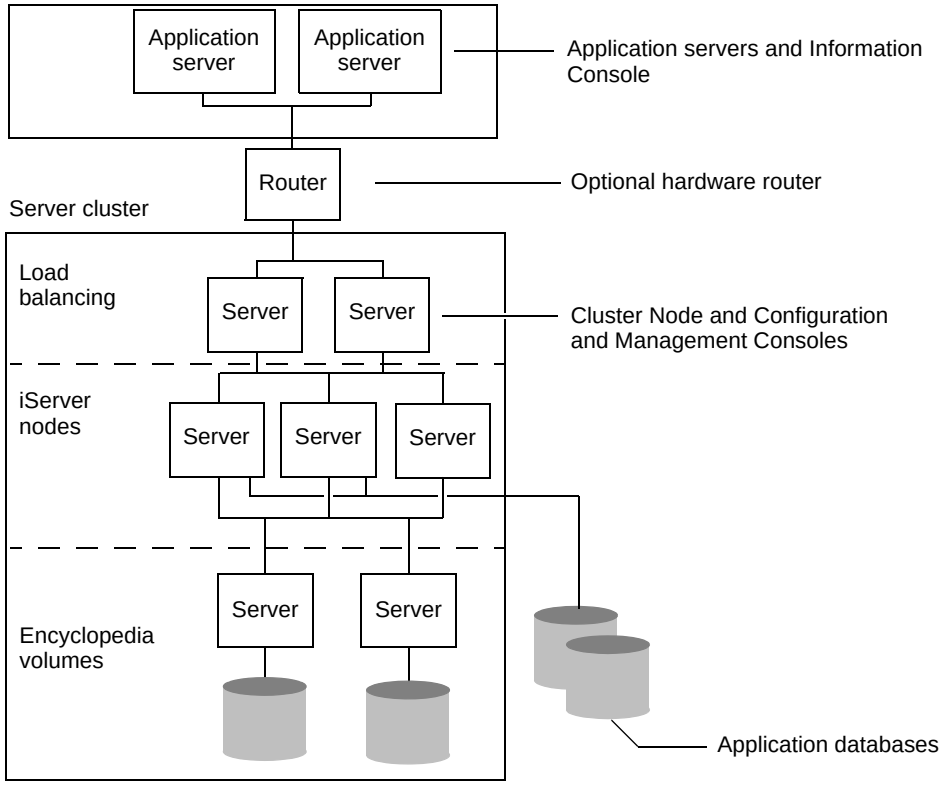

Figure 9-1 An iServer cluster

In this example, the cluster uses a network router to create a single virtual IP address to distribute the load-balancing requests that come into two nodes. The Actuate Information, Configuration, and Management Consoles support distributing requests to multiple machines, which handle load balancing in the

cluster. Figure 9-1 shows the following cluster sections:

- In the load-balancing section, the requests are routed to a node that performs load balancing.
- In the nodes section, iServer generates documents and delivers them to clients for viewing.
- In the Encyclopedia volumes section, iServer running on multiple machines maintains Encyclopedia volume management information and controls access to the volumes. The volumes can be on machines that are not running iServer but are accessible to a machine. A node shares all volumes.

In Figure 9-1, separate machines handle separate functions. You can combine these functions on one machine.

About distributing services to a cluster

You can control the configuration of a cluster and each node in a cluster, including:

- iServer services and settings
- Services and settings for a cluster node
- Cluster membership

You can enable one or more services in each server template.

In a cluster, you can use templates to configure the nodes instantiated in an iServer System, as shown in Figure 9-2. Each node is a computer running iServer, configured through the template definitions stored in acserverconfig.xml, which is accessible by all nodes through a shared directory. The following letters represent the available services in Figure 9-2:

- M is the Message Distribution service.
- V is the View service.
- F is the Factory service.
- I is the Integration service.
- C is the Caching service.

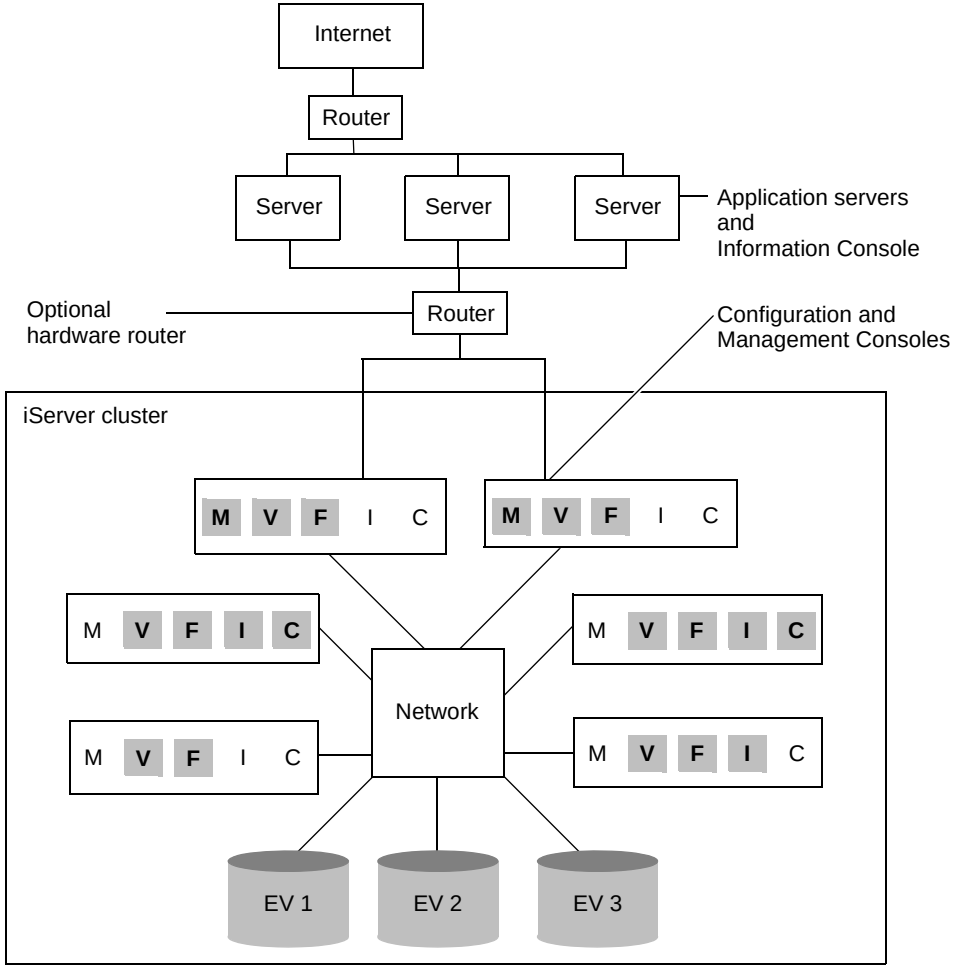

Figure 9-2 An iServer cluster configuration

A letter in a gray box represents an enabled service. The other letters represent disabled services. When you enable or disable a service in a template, the cluster nodes that use different templates can have different functionality.

Two nodes have the Message Distribution service enabled in the cluster in the preceding example. In the example, the two machines share one virtual IP address to communicate with the router.

Information Console or iServer perform load balancing independent of load-balancing capabilities in the router. All requests to the iServer cluster go to one of the two redirector nodes, which are the nodes with the Message Distribution service enabled. The redirector nodes dispatch the requests to other nodes in the cluster based on message type and cluster load.

The configuration file, acserverconfig.xml, located in a shared location provides centralized management of the cluster configuration through the use of server templates and other parameters. A server template is a set of parameters that specifies the configuration properties of a node or set of nodes.

The acpmdconfig.xml file for each node, located in the AC_SERVER_HOME/etc directory on the node machine, has its <AC_CONFIG_HOME> element set to point to the location for the shared acserverconfig.xml file. The <AC_TEMPLATE_NAME> element specifies which template the node uses.

One or more nodes in the cluster manage request message routing. The Message Distribution service uses HTTP to communicate between nodes in a cluster. An iServer cluster supports multicast network communication.

About the configuration home directory

The configuration home directory is the directory that holds the configuration file, acserverconfig.xml, and the licensing file, acserverlicense.xml.

The acserverconfig.xml file contains the information for connecting to Encyclopedia volumes and printers, Message Distribution state, configuration settings for Factory, View, Integration, and Caching services, the location of partitions, and licensing and Open Security information.

In a cluster, acserverconfig.xml plays a central role in the operation of cluster nodes. The file includes server templates for various server roles with each template containing configuration information for connecting to Encyclopedia volumes, printers, and services for all the nodes in the cluster. When a node joins a cluster, it first configures itself using its template in the acserverconfig.xml file located in the shared directory.

The acserverconfig.xml file is located by default in AC_DATA_HOME/config /11<service pack number>. This directory is referred to as the configuration home directory. Make the configuration home directory sharable to allow cluster nodes access to this file. To specify the location of this directory for each node, modify the value of the <AC_CONFIG_HOME> element located in acpmdconfig.xml, which by default is located in AC_SERVER_HOME/etc of the node.

How to change the location of the configuration home directory

To change the location of the configuration home directory containing acserverconfig.xml and acserverlicense.xml, perform the following tasks:

1. Shut down all cluster nodes.
2. Stop the Actuate 11 BIRT iServer service on each node.
3. Back up the configuration home directory.

4 Move the configuration folder to the new destination.

5 Share the configuration folder.

6 For every node dependent on this configuration home directory, update the <AC_CONFIG_HOME> element located in the node's acpmdconfig.xml file.

7 Start Actuate 11 BIRT iServer service for each node.

About the primary configuration files

There are two main configuration files:

- acpmdconfig.xml
  Located by default in AC_SERVER_HOME/etc
- acserverconfig.xml
  Located by default in AC_DATA_HOME/config/11<service pack number>

In a cloud configuration, server templates give cloud the flexibility to change configurations at the launch time. Acpmdconfig.xml contains the configurations that are node specific. When the Process Management Daemon (PMD) starts up, it reads these configurations first and exposes them to the process environment variable list.

The acserverconfig.xml file contains other cluster and node configuration parameters, which specify the host names, port numbers, volume names, and server templates to be used by the nodes. The following sections elaborate on these configuration files.

About acpmdconfig.xml

Acpmdconfig.xml sets the environment variables at the operating system level. The administrator can specify the node configuration settings in acpmdconfig.xml. Alternatively, the administrator can set the necessary environment variables in setsrvrenv and run this script before restarting an instance.

Using acpmdconfig.xml, set <AC_CONFIG_HOME> to the directory that contains acserverconfig.xml and acserverlicense.xml. Modify the <AC_TEMPLATE_NAME> element to specify which template this iServer uses.

When creating an image for a set of machines, create a template name, such as SharedTemplate, in acserverconfig.xml. In acpmdconfig.xml, set <AC_TEMPLATE_NAME >to the template name. The administrator can create different templates that use different images for machines with varying computing capacities and resources. Listing 9-1 shows acpmdconfig.xml for a machine named urup, a node containing the configuration home directory.

Listing 9-1 The acpmdconfig.xml file

```
<PMDConfig>
   <!--Actuate system Type -->
   <System>Cluster</System>
   <Mode>Default</Mode>
   <PMDConfigFileVersion>2</PMDConfigFileVersion>
   <!--Daemon SOAP endpoint information -->
   <DaemonSOAPPort>8100</DaemonSOAPPort>
   <!-- Disk Thresholds are in MB -->
   <MinDiskThreshold>100</MinDiskThreshold>
   <LowDiskThreshold>300</LowDiskThreshold>
   <!--Server information -->
   <Server>
      <Startup>Auto</Startup>
      <AC_TEMPLATE_NAME>urup</AC_TEMPLATE_NAME>
      <AC_DATA_HOME>C:\Actuate\iServer\data</AC_DATA_HOME>
      <AC_CONFIG_HOME>C:\Actuate\iServer\data\config\11SP4
         </AC_CONFIG_HOME>
      <AC_JRE_HOME>C:\Program Files\Common Files\Actuate\11.0\
         JDK160\jre</AC_JRE_HOME>
      <AC_JRE64_HOME>C:\Program Files\Common Files\Actuate\11.0\
         JDK160_64\jre</AC_JRE64_HOME>
      <AC_JAVA_HOME>C:\Program Files\Common Files\Actuate\11.0\
         JDK160</AC_JAVA_HOME>
      <AC_ODBC_HOME>C:\Program Files\Common Files\Actuate\11.0\
         odbc</AC_ODBC_HOME>
      <AC_SERVER_IP_ADDRESS>urup</AC_SERVER_IP_ADDRESS>
      <AC_SOAP_DISPATCH_ADDRESS>urup</AC_SOAP_DISPATCH_ADDRESS>
      <AC_DOC_BASE>http://www.actuate.com/documentation/R11
         </AC_DOC_BASE>
      <AC_ICU_DATA>C:\Program Files\Actuate11SP4\iServer\bin
         </AC_ICU_DATA>
   </Server>
   <!-- Servlet Container information -->
   <ServletContainer>
      <Startup>Auto</Startup>
      <JavaOpts
          Args="-Xms128m -Xmx512m -XX:MaxPermSize=128m"/>
   </ServletContainer>
</PMDConfig>
```

Use this configuration file to set the environment variables at the operating system level. Table 9-1 shows the configuration setting in acpmdconfig.xml.

Table 9-1 acpmdconfig.xml node-specific configuration settings

Configuration Name	Description
AC_TEMPLATE_NAME	Template name
AC_DATA_HOME	Location of data directory
AC_CONFIG_HOME	Location of acserverconfig.xml and the license file
AC_JRE_HOME	Location of Java run-time environment (JRE)
AC_JRE64_HOME	Location of 64-bit Java run-time environment (JRE)
AC_ODBC_HOME	Location of ODBC resources
AC_SERVER_IP_ADDRESS	iServer IP address
AC_SOAP_DISPATCH_ADDRESS	iServer dispatcher IP address
AC_ICU_DATA	Location of ICU library

About acserverconfig.xml

This configuration file can specify one or more templates to provide flexibility when instantiating iServer nodes in an environment where machines have varying resources. Acserverconfig.xml provides access to the following elements:

- System
- FileSystems
- MetadataDatabases
- Schemas
- Volumes
- Templates
- Resource groups
- Printers
- ServerList

When starting a cluster, the acserverconfig.xml file must be in a shared directory.

Listing 9-2 The acserverconfig.xml file

```
<Config>
   <System
      ClusterID="_4_fffefdfc_ce4fdb2c_e4c"
      SystemName="corp"
      DefaultLocale="en_US"
```

```
      DefaultEncoding="windows-1252"
      ConfigFileVersion="13"
      EncyclopediaOwnerID="_6_fffefdfc_ce4fdb2c_e4c"
      SystemDefaultVolume="corp"
      ClusterDatabaseSchema="ac_corp_system"
      DefaultCLocaleOnWindows="true"
      EncyclopediaVolumeServer="urup">
     <UsageAndErrorLogging/>
     <SMTPServers/>
  </System>
...
  <Templates>
     <Template
         Name="urup"
         PMDPort="8100"
         ActuateBuild="DEV110711"
         ActuateVersion="11 Service Pack 4"
         ServerSOAPPort="11100"
         AppContainerPort="8900"
         RequesterRSAPIVolume="corp">
...
     </Template>
     <Template
        ...
     </Template>
  </Templates>
</Config>
```

Creating a cluster

To create a cluster, the administrator first installs a stand-alone iServer or uses an existing installation. Next, the administrator shares the configuration home directory, so other servers joining the cluster can access it. Then, the administrator adds new nodes to form the cluster.

There are two methods of adding a new node to the cluster.

- Through the use of the configuration file, acpmdconfig.xml
- Through the installation wizard, when performing a custom cluster-node installation

Every cluster node must have network access to the following directory and resources to join the cluster:

- The shared configuration home directory

- Cluster resources, such as printers, database systems, disk storage systems, and Encyclopedia volume directories.

The administrator can configure nodes, using server templates, to run different services and to process different types of requests. Important factors to consider when configuring nodes include processing power and access to hardware and software resources, such as printers and database drivers.

From the configuration console the administrator can add resources, such as partitions, Encyclopedia volumes, and resource groups to the cluster.

Creating an initial cluster

1. Install two stand-alone iServers. In this example, the server names are urup and kozu.
2. On urup, share the following folders so that they are available to other servers:
   - \config\11SP<service pack number>
   - \encyc
3. If urup and kozu are both Windows machines, turn off the Windows firewall on both urup and kozu.
4. Confirm that urup and kozu can access each other on the network. In a command prompt on each computer, type the ping command followed by the IP address or host name of the other computer. Verify that the receiving computer replies to the sending computer.
5. Shut down urup and kozu by performing the following tasks on each machine:
   1. Log in to Configuration Console.
   2. On the Simple view, choose Stop system, as shown in Figure 9-3.

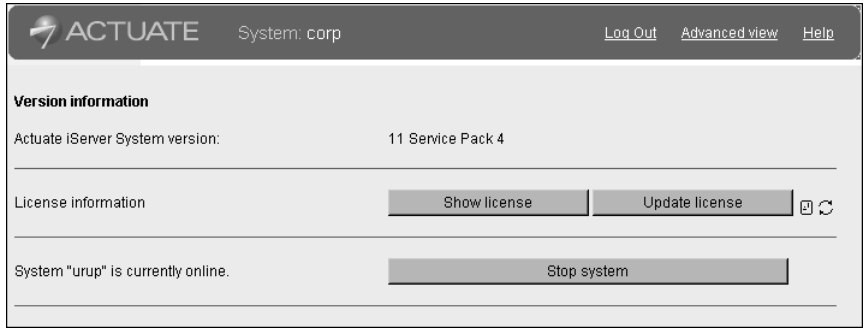

Figure 9-3 Stopping the system

6. In a Windows environment, using Administrative Tools—Services, stop the Actuate 11 BIRT iServer services for urup and kozu on each server machine.

7. On urup, make a backup of acserverconfig.xml, then open it and perform the following tasks:
   1. Locate the <ServerFileSystemSettings> element under the <Template> element.
   2. Under <ServerFileSystemSettings>, locate:

      ```
      <ServerFileSystemSetting
         Name="DefaultPartition"
         Path="$AC_DATA_HOME$/encyc"/>
      ```

      Using Universal Naming Convention (UNC) format, change the value of the Path attribute to specify the location of DefaultPartition. DefaultPartition contains the Encyclopedia data files. By default, the DefaultPartition path is:

      ```
      AC_DATA_HOME\encyc
      ```

      To specify this path using UNC format, type:

      ```
      \\urup\encyc
      ```
   3. Locate the <ConnectionProperties> element under the <MetadataDatabase> element.
   4. Under <ConnectionProperties> locate:

      ```
      <ConnectionProperty
                  Name="server"
                  Value="localhost"/>
      ```
   5. Change Value from localhost to the name of the machine on which the volume resides, in this example “urup”. For urup, the volume is located on localhost, but in a cluster setting the administrator must use the machine name, since it is not on a localhost from the reference point of the other nodes.
   6. Save acserverconfig.xml file.
8. On kozu, open acpmdconfig.xml located in AC_SERVER_HOME/etc and perform the following tasks:
   1. Change <AC_CONFIG_HOME> to point to the path specified by <AC_CONFIG_HOME> in acpmdconfig.xml on urup. For example, if <AC_CONFIG_HOME> on urup is set to AC_DATA_HOME\config, type \\urup\config as the value for <AC_CONFIG_HOME> on kozu.
   2. Change <AC_TEMPLATE_NAME> to the name specified by <AC_TEMPLATE_NAME> on urup.
9. Save the modified acpmdconfig.xml file.
10. Start the Actuate 11 BIRT iServer service for urup.

11 After urup starts, start the Actuate 11 BIRT iServer service for kozu.

12 Log in to the Configuration Console for either node.

13 Choose Advanced View.

14 In Advanced View, select Servers from the side menu.

Servers displays the nodes, urup and kozu. Both servers are using the urup template, as shown in Figure 9-4.

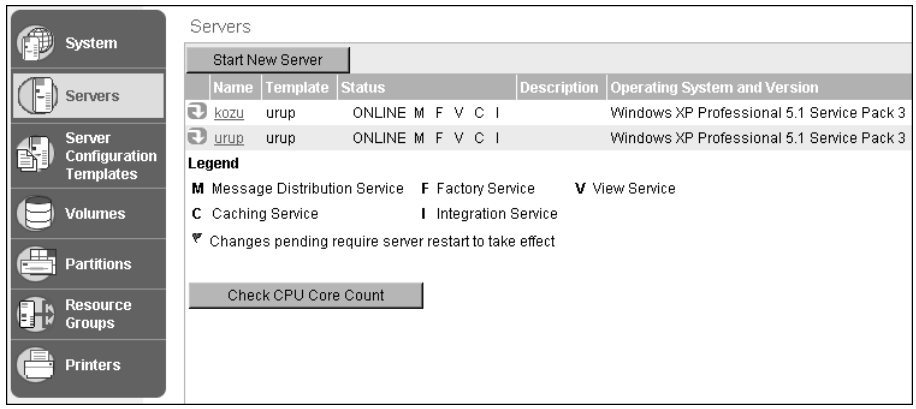

Figure 9-4 A two-node cluster

When adding a node to an already existing cluster, it is not necessary to shut down the cluster before adding the new node.

Configuring heartbeat messaging

Nodes in a cluster use heartbeat messaging to monitor the status of the other nodes in the cluster. On System—Properties, the administrator configures System Heartbeat, as shown in Figure 9-5.

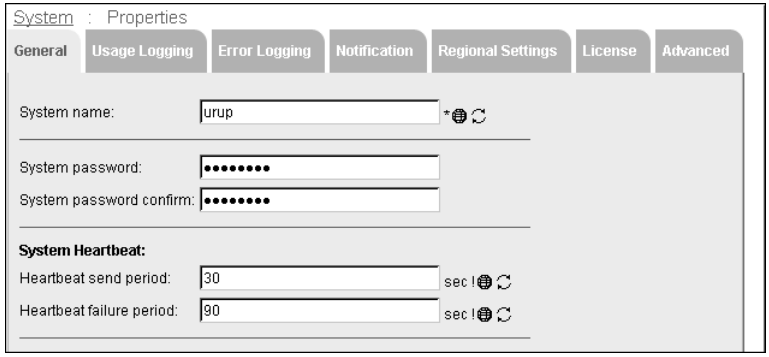

Figure 9-5 Configuring System Heartbeat

System Heartbeat consists of properties for measuring messaging frequency. System Heartbeat properties are:

- Heartbeat send period
  The interval to send a heartbeat message, typically 30-40 seconds. Change this property to increase or decrease the number of heartbeat messages.
- Heartbeat failure period
  The period in which the cluster nodes determine that another node is within the heartbeat failure period, typically 90-100 seconds. If the monitoring nodes do not receive a response within this period, the monitored node is assumed to be down.

Configuring Message Distribution service properties

In BIRT iServer Release 11, the administrator configures the Message Distribution service for a server template rather than for an individual node.

How to configure the Message Distribution service for a server template

To configure the Message Distribution service (MDS) for a particular template, perform the following tasks:

1. Log on to the Configuration Console on urup, and choose Advanced View.
2. From the side menu, choose Server Configuration Templates. Select the template for which you want to configure the Message Distribution service, as shown in Figure 9-6.

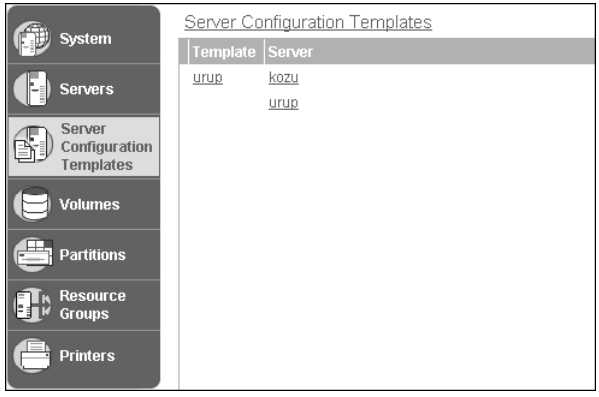

Figure 9-6 Server templates

3. In Advanced, expand the Message Distribution Service folder and the Process Management folder, as shown in Figure 9-7.

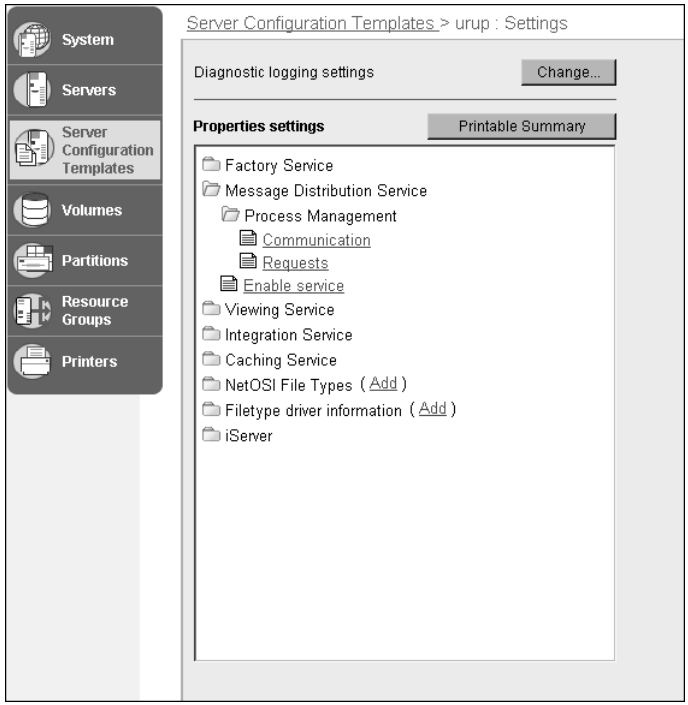

Figure 9-7 Message Distribution Service

4 In Message Distribution Service—Process Management—Communication, as shown in Figure 9-8, accept or set the value for the Message Distribution service port.

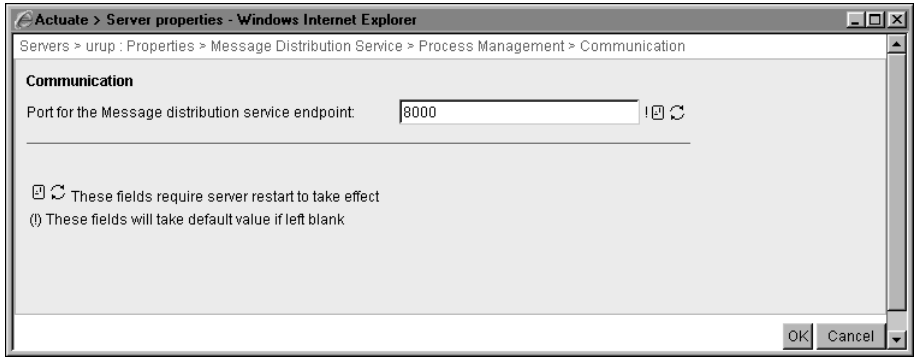

Figure 9-8 Setting communication properties

5 In Message Distribution Service—Enable service, enable or disable the Message Distribution service by selecting or deselecting Enable request service, as shown in Figure 9-9.

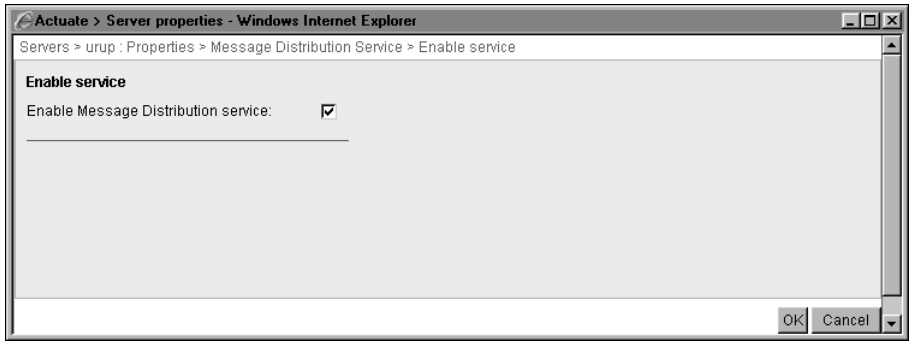

Figure 9-9 Enabling and disabling the Integration service

6 In Message Distribution Service—Process Management—Requests, accept or set the value for the total number of concurrent requests processed by iServer through the Message Distribution service, as shown in Figure 9-10. When the maximum number of requests is reached, iServer refuses to process any new incoming requests.

If you do not use the recommended default value of 1000, perform throughput tests after changing the value. If the value is 0, iServer does not accept any incoming requests. If the value is too low, iServer does not utilize all system resources when handling the maximum number of client requests. Increasing the value can reduce total throughput.

At peak usage times, iServer might need the maximum available system resources, such as system memory and temporary disk space, to handle requests.

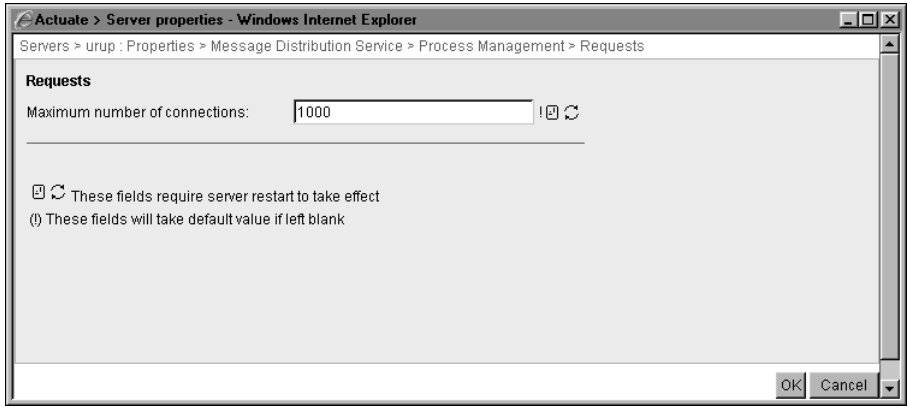

Figure 9-10 Specifying the maximum number of connections

You can also modify these settings through acserverconfig.xml, for each template.

Adding and modifying server templates

Every cluster node takes its configuration from the template to which it is assigned. To change a node configuration, the administrator either modifies the assigned template or creates a new template and assigns it to the node. The administrator creates a server template definition in the acserverconfig.xml file located in the shared configuration home directory.

The administrator can create a server template for any possible server configuration. The following sections cover the process of modifying an existing template, and creating a new template.

Modifying a server template

Back up the original acserverconfig.xml located in the shared configuration home directory, then open the file and perform the following tasks:

1. Shut down the nodes that use the template you want to modify.
2. Open acserverconfig.xml.
3. Under the <Templates> tag, locate the template you want to modify.
4. Locate and modify the elements associated with the parameters you want to configure.
5. After making the modifications to the template, save acserverconfig.xml.
6. Start the nodes, which now use the modified templates.

Creating a new server template

The easiest way to add a new template is to copy an existing template, and then modify it for new functionality.

1. Back up the original acserverconfig.xml file.
2. Open acserverconfig.xml.
3. Locate an existing template that closely approximates the new template you want to create.
4. Copy the template, then paste it under the last </Template> tag above the </Templates> tag.
5. Modify the template name attribute, to give the new template its own unique name.
6. Modify the parameters of the elements to produce the functionality you desire.
7. Save acserverconfig.xml.

The template is now ready for use by cluster nodes. To use the new template, modify the <AC_TEMPLATE_NAME> for the node in acpmdconfig.xml file to the new template name.

Adding a node to a cluster

To add a new node to the cluster perform the following tasks:

1. Shut down the node.
2. Open the acpmdconfig.xml file.
3. Modify <AC_CONFIG_HOME> to point to the shared configuration home directory that contains acserverconfig.xml for the cluster.
4. Modify <AC_TEMPLATE_NAME> to use a server template from the available server templates listed in acserverconfig.xml.
5. Save the acpmdconfig.xml file.
6. Start the new node. The node will automatically contact the acserverconfig.xml for the configuration profile and join the cluster, as shown in Figure 9-11.

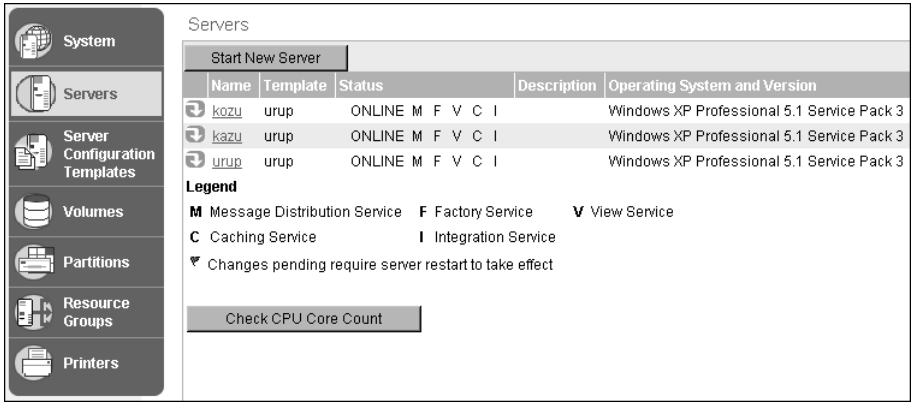

Figure 9-11 A three-node cluster

If the particular server configuration that you want to use for the new node is not available in the templates listed in acserverconfig.xml, create a new template. For more information on the process of creating a new server template, see "Creating a new server template," earlier in this chapter.

About node configuration

Every node is configured by its server template. To reconfigure a node, the administrator can either change the template that the node is using, or modify the

existing template. Use caution when modifying a template, because you are modifying the configuration of all the nodes that subscribe to the template.

You configure templates to perform tasks, such as generating and printing documents. Ensure that the cluster meets the following requirements:

- Configure access to printers and databases from templates that run the Factory service.
- Configure resource groups.

About cluster configuration

You can make the following configuration changes to a cluster:

- Add more nodes to run Factory and View services to handle an increased workload.
- Add new nodes to the cluster to increase robustness, and take over the functionality in the case of a node failure.

Testing a cluster

You can test a cluster using a URL, such as the following one:

```
http://host1:8900/<context root>/login.jsp?serverURL=
   http://host2:8000&daemonURL=http://host3:8100
```

where

- context root is acadmin, the name for Management Console.
- 8900 is the default port number for the application container running Management Console.
- 8000 is the default port number for iServer.
- 8100 is the default port number for the iServer service on Windows and the iServer service Process Management Daemon on UNIX and Linux.

This URL tests a three-node cluster that runs Management Console on host1, a node iServer on host2, and the iServer service on host3.

Starting and stopping a node

When iServer starts, it gathers the following licensing information:

- Validity of the license file
- Encyclopedia volume limitations

- CPU limitations, if applicable
- License expiration date, if applicable

You start or stop the nodes in a cluster using Servers in Configuration Console.

You can start and stop a node independent of the cluster.

How to stop a single cluster node

1. From the Advanced view of Configuration Console, choose Servers.
2. In the list of servers, point to the arrow next to the iServer name that you want to stop.
3. Choose Stop, as shown in Figure 9-12. iServer changes the status of the server.

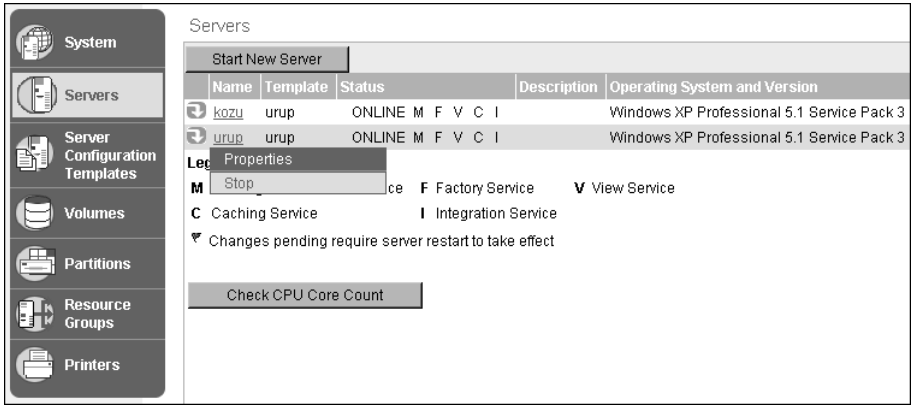

Figure 9-12 Starting or stopping a single node

About starting a node

When you start an iServer node, the following events occur:

- The node configures itself by contacting acserverconfig.xml located in a shared configuration home directory.
- The node joins the cluster.
- The services and resource groups of the node become available to the cluster.
- The status of the node changes to online.

How to start a node

To start an offline node which is already part of the cluster, perform the following tasks:

1. From the Advanced view of Configuration Console, choose Servers. Point to the arrow next to the offline node, and choose Start, as shown in Figure 9-13.

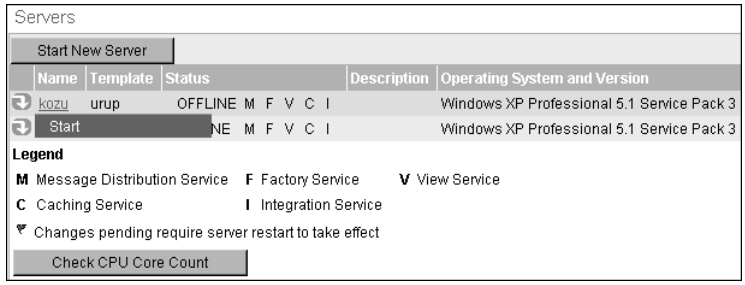

Figure 9-13 Choosing Start

2 In Servers—Start server, accept the default values for Host Name, iServer Process Manager Port Number, and Server Template, as shown in Figure 9-14. Alternatively, you can change these values.

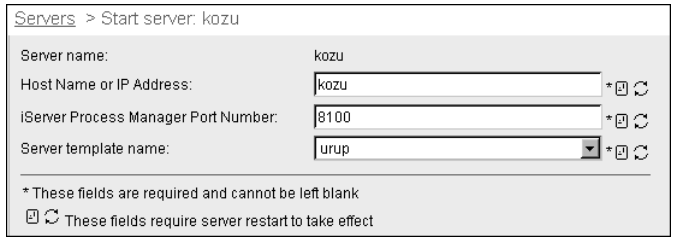

Figure 9-14 Starting a node

3 Choose Start. After a few moments, the node comes online using the chosen server template.

How to start a new node

To start an offline node that is not currently part of the cluster, but is configured to be part of the cluster, perform the following tasks:

1 In the Advanced view of Configuration Console, choose Servers, then choose Start New Server.

2 In Servers—Start New Server, enter the values for the Server Name, Host Name, iServer Process Manager Port Number, and choose a Server Template, as shown in Figure 9-15.

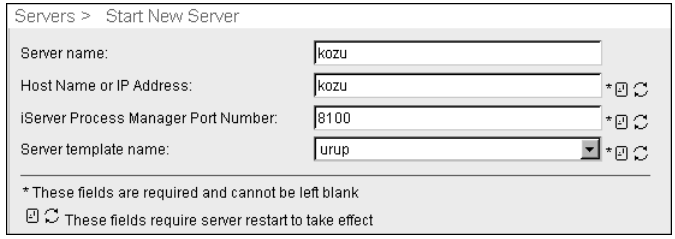

Figure 9-15 Starting a new node

3 Choose Start. After a few moments, the node comes online using the chosen server template, and joins the cluster.

About stopping a cluster or a node

Shutting down a cluster differs from stopping a node in a cluster. To shut down a a node, from the node's Configuration Console, on System—Status, choose Stop. When you stop a node, the following events occur:

- The node leaves the cluster.
- The services and resource groups on the node are no longer available to the cluster.
- The status of the node changes to offline.

To shut down a cluster, you shut down all the nodes forming the cluster.

Removing a node from a cluster

After you remove the node from the cluster, iServer can operate in a stand-alone configuration only if you originally installed it in a stand-alone configuration. When you remove a node from a cluster, the following events occur:

- iServer removes information about the node from the cluster configuration information.
- When you log in to Configuration Console for the cluster, you cannot access the node.

To remove a node from a cluster, complete the following tasks in the following order:

- Log in to Configuration Console for the cluster.
- Reconfigure the cluster to handle functionality assigned to the template the node used. For example, ensure that remaining nodes in the cluster can assume responsibility for the following functionality:
  - Running Message Distribution, View, Factory, Integration, and Caching services
  - Fulfilling resource group requirements
  - Other functionality that the node supports
- Stop the system on the cluster node. When you take the node offline, the cluster can no longer access its services.
- Remove the node from the cluster by performing the following tasks:

1. Stop the system on the node containing the shared configuration home directory.
2. In Windows Explorer, navigate to AC_DATA_HOME\config\ 11SP<service pack number>. Open acserverconfig.xml in a text editor, such as Notepad.
3. Find the <ServerList> element, located towards the end of the file.
4. Select and delete the <ServerInfo> element for the cluster node that you are removing.
5. Save the modified acserverconfig.xml file.
6. Select Start to start the server system, as shown in Figure 9-16.

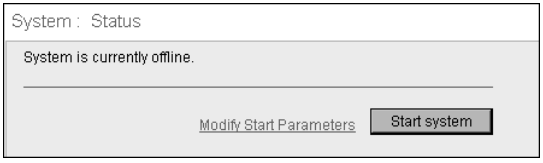

Figure 9-16 Starting the server

7. Choose Servers from the side menu. The name of the removed cluster node no longer appears in the server list.

Removing a node from the cluster does not change the configuration information for the cluster. For example, cluster nodes can still access the following resources, even after you remove the node from the cluster.

- Printers the cluster uses
- Partitions that refer to directories on the hard drive that the cluster can access

Managing a cluster

It is important to consider the following when managing a cluster:

- Accessing partitions
- Specifying the Encyclopedia volume location
- Cluster option requirements

Accessing partitions

All cluster nodes must have access to resources that the cluster uses to read and write to every partition in the cluster. In Add Partition, you specify a path to the partition, as shown in Figure 9-17. In BIRT iServer 11, the partition path is set for

each template rather than for each individual node.

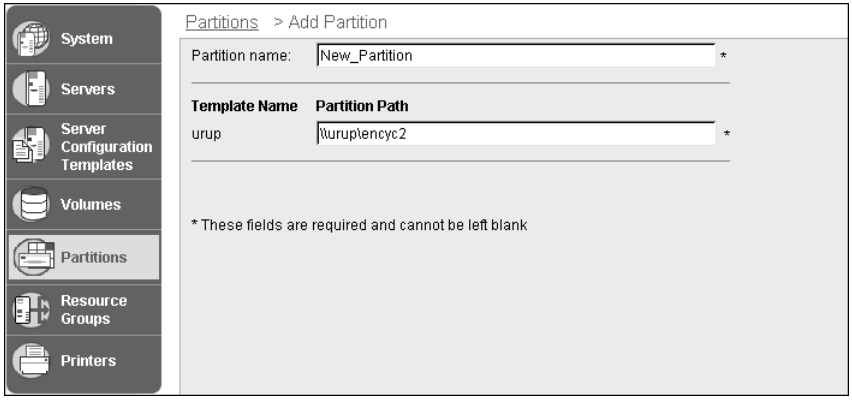

Figure 9-17 Adding a partition

Specifying the Encyclopedia location

The default installation of a stand-alone iServer configures an Encyclopedia volume, but the installation of a cluster node does not include an Encyclopedia volume. From the Configuration Console, you can add a new volume. In iServer Release 11, every server serves all volumes.

You specify the location of an Encyclopedia volume for a cluster in a template. When the node accesses the template to which it was assigned, it also receives the location of the partitions.

About cluster option requirements

A cluster uses a single license file. All nodes in a cluster have the same iServer System options enabled. Any node that you add to the cluster uses only the options that the cluster uses. For example, if you add an iServer that was in a stand-alone configuration with page-level security to the cluster, and the cluster does not have page-level security enabled, the new node does not support page-level security.

To support multiple Encyclopedia volumes in a cluster, you need the Multi-Tenant Option or an equivalent. For example, you need the Multi-Tenant Option to use both Encyclopedia volumes in a cluster that you create from multiple stand-alone iServers, each with their own Encyclopedia volume.

When iServer is offline, you can replace the license file in the configuration home directory.

Handling file system failure

iServer handles file system failure on stateless and stateful file systems. This overview uses Network File System (NFS) as an example of a stateless network file system and Common Internet File System (CIFS) as an example of a stateful network file system.

iServer handles some file system failures by retrying file I/O. Retrying file I/O works when a file system failure is transparent to iServer. For example, on an NFS-based network storage system, a file system failure can be transparent to iServer. Retrying file I/O is insufficient in a configuration where file system failure is not transparent to iServer, such as on a Windows-based CIFS file system.

On a stateless file system such as an NFS-based file system, iServer can handle a network storage system failure. The machine detects that the connection to a file system is lost and attempts to reconnect. When the file system recovers, the machine re-establishes a connection to the file system. If the connection to the file system does not time out during failure, iServer does not detect the failure.

On a stateful file system such as a Microsoft Windows-based CIFS network file system, a machine using the file system tracks file system connection states, including open files and locks. If the file system connection breaks, the machine loses connection state information. The CIFS client machine must manually reestablish file system connections. iServer can re-establish file system connections on a stateful network file system.

iServer identifies a file system failure as a failure of the following file I/O functions:

- Reading the configuration lock file
- Reading the Encyclopedia volume lock file
- Reading or writing to an Encyclopedia volume

Failure to read the configuration lock file affects the cluster nodes. The other two I/O failures affect the Encyclopedia volume.

Configuring the cluster administrator e-mail account

In System—Properties—Advanced—Cluster Operation—Administrative, shown in Figure 9-18, you can specify administrative e-mail account information.

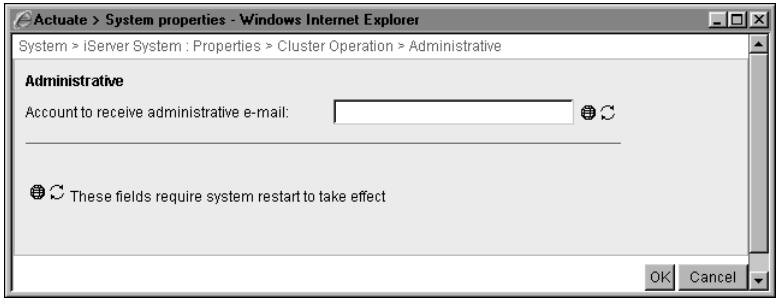

Figure 9-18 Specifying administrative e-mail account information

The AdminEmail property specifies the account that receives administrative e-mail, such as the notice iServer sends when licensing problems occur.

Managing console configurations and load balancing

You can configure iServer and Configuration and Management Consoles in several ways. The following sections describe the most common configurations.

Using the consoles directly

This configuration places both iServer and its Configuration and Management Consoles behind a corporate firewall, as shown in Figure 9-19.

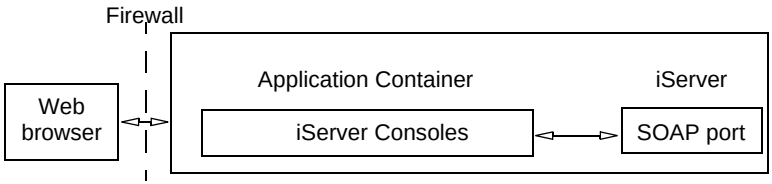

Figure 9-19 Using the consoles directly

Perform a stand-alone installation to set up this configuration.

Using the consoles through a firewall

Figure 9-20 shows iServer and Configuration and Management Consoles deployed behind a second firewall.

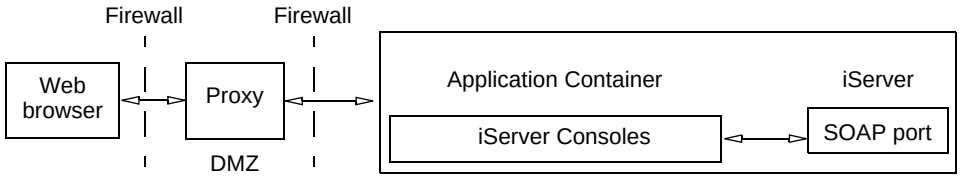

Figure 9-20 Using the consoles through a firewall

Deploy the load-balancing proxy separately from the application container, which exists in the demilitarized zone (DMZ). You can use a proxy from a third-party vendor or the Actuate proxy.

To install this configuration, you must complete the following tasks in this order:

- Install iServer, integrated with the consoles. Ensure that each console uses the same context root name. The default name is acadmin.
- Deploy the Actuate or third-party proxy on a machine that exists in the DMZ.
- Configure the proxy with the list of available nodes.
  Management and Configuration Consoles should be installed on the nodes.

Using multiple console installations

You can have multiple console installations in a cluster. The proxies maintain the session stickiness and distribute requests to the available installations of iServer consoles. The load balancer that iServer uses in this scenario does not have to support session stickiness.

Figure 9-21 illustrates a configuration with multiple installations of iServer consoles.

To set up this configuration, complete the following tasks in this order:

- Install the cluster, with the integrated consoles.
- Deploy the Actuate or third-party proxy on different machines than those where you installed iServer consoles.
- Configure the proxy with the list of available nodes with iServer consoles.

If you remove a node from a cluster, install the consoles for that iServer if they are not already installed.

About load balancing

iServer consoles ensure high availability and distribute tasks for efficient processing using load balancing. The installation of iServer consoles include a lightweight web application that distributes requests to the available iServer nodes in a cluster.

When deployed, this proxy enables load balancing among Management Console instances. The proxy is available as mgmtconsoleproxy.war (web archive) in the downloaded iServer product package. To use the proxy, install it under its own context root on your application server.

The easiest way to customize the proxy for your installation is to modify the WAR file to use a modified web.xml file for each installation.

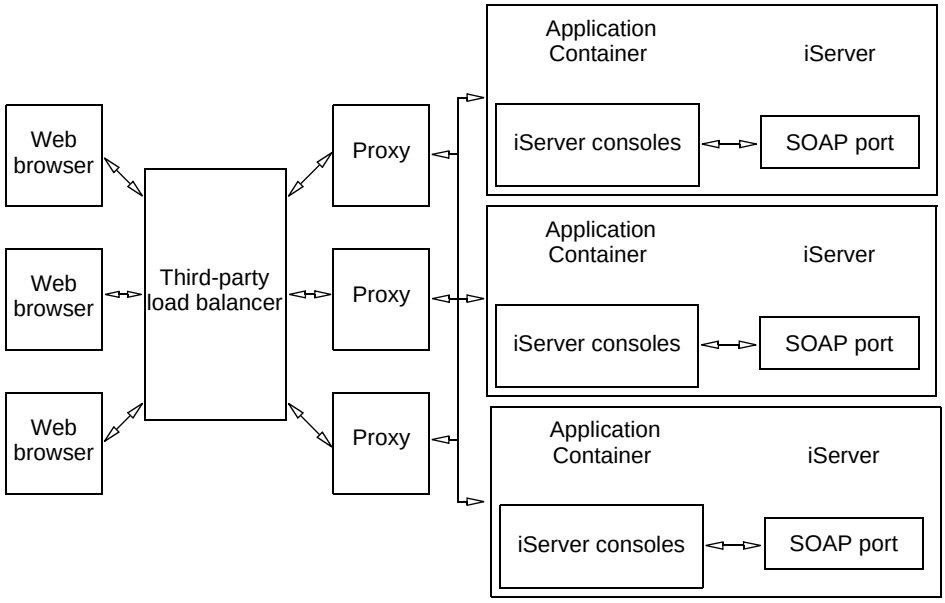

Figure 9-21 Using multiple iServer consoles

Deploying load balancing

Before you use the WAR file to deploy the Management Console load-balancing application, AcProxy, customize the WAR file for your installation as follows:

- Extract the configuration file, web.xml, from the WAR file.
- Customize web.xml.
- Create a new WAR file using the customized web.xml file.

How to extract the configuration file from the WAR file

1. Create a temporary directory, such as C:\temp\AC on a Microsoft Windows server or /temp/ac on a UNIX or Linux server.
2. Decompress the mgmtconsoleproxy.war file onto the temporary directory.

For example, on Windows, open a Command window and type the following commands, replacing the X: drive letter with a drive letter appropriate to your system:

```
mkdir C:\temp\AC
copy X:\mgmtconsoleproxy.war  .
jar -xf mgmtconsoleproxy.war
```

The AcProxy files appear in the temporary directory.

Using a UNIX or Linux server, type the following commands:

```
mkdir /temp/ac
cp /dev/dsk/iServerProduct/mgmtconsoleproxy.war  .
jar -xf mgmtconsoleproxy.war
```

The AcProxy files appear in the temporary directory.

3 If you used the temporary path in step 3, the file location is:

Windows—C:\temp\AC \WEB-INF\web.xml

UNIX or Linux—/temp/ac/WEB-INF/web.xml

How to customize web.xml for your installation

1 Using a text editor that accepts UTF-8 encoding, edit web.xml to configure AcProxy for your application server.

2 Specify the list of available nodes with iServer consoles in the SERVER_LIST context parameter. The list contains all iServer console URLs that AcProxy uses to balance requests. The default list is empty. Add your URLs in a comma-separated list. The following code is a sample SERVER_LIST entry:

```
<context-param>
   <param-name>SERVER_LIST</param-name>
   <param-value>http://hostname1:8900/,http://hostname2:8900/
      </param-value>
</context-param>
```

3 Save and close web.xml.

How to customize the WAR file for your installation

To create a new WAR file, type the following command:

```
jar -cf ..\newmgmtconsproxy.war *
```

This command creates Newmgmtconsproxy.war in the /temp directory. This file is a new WAR file for AcProxy, and it contains the modified configuration values.

Use Newmgmtconsproxy.war to deploy to your application servers instead of Mgmtconsoleproxy.war.

Chapter

10

Configuring Integration and Caching services

This chapter contains the following topics:

- About the Integration service
- Managing Integration service resources
- Using information objects
- About Actuate Caching service
- Configuring the Caching service

About the Integration service

Whenever possible, the Integration service uses a single database connection for a query. For example, if a design uses information object data sources that depend on multiple data connection definition files (.dcd), the Integration service may use only one database connection if the DCDs have the same values for the User name, Password, and Server properties and the query can be completely pushed to the database. For concurrent queries, however, the Integration service creates multiple database connections.

Managing Integration service resources

The administrator configures Integration service properties on Server Configuration Templates—Advanced to control how iServer and an Integration handle data from a data source. These settings also determine how the process manages RAM memory and disk-based memory when processing information object data. Figure 10-1 shows the properties in Server Configuration Templates—Settings—Integration Service.

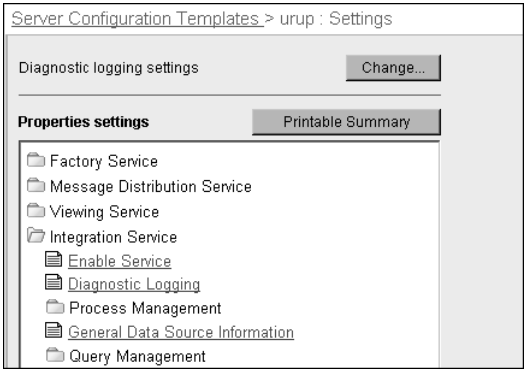

Figure 10-1 Viewing Integration service properties

Enabling the Integration service

The administrator can enable or disable the Integration service from Server Configuration Templates—Settings, or by setting the EnableIntegrationService parameter in AC_DATA_HOME/config/acserverconfig.xml to true or false.

How to enable the Factory service

1 Expand Integration Service, as shown in Figure 10-1, and choose Enable Service.

2 For Enable Integration service, accept the default value, which is selected, as shown in Figure 10-2.

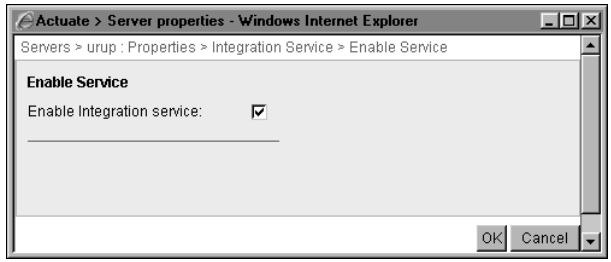

Figure 10-2 Enabling or disabling the Integration service

About diagnostic logging

The administrator can configure diagnostic logging by expanding Integration Service, and choosing Diagnostic Logging. For more information, see Chapter 3, "Using diagnostic, usage, and error logging."

Setting port numbers for process communication

In Server Configuration Templates—Settings—Integration Service—Process Management—Communication, as shown in Figure 10-3, the administrator accepts or sets the value for each of the following properties:

- Port for Integration server message endpoint
  Port for Integration service process communication with Encyclopedia volume processes.
- Port for Integration server query endpoint
  Port for Integration service process communication and iServer when running information objects to pass queries and data.

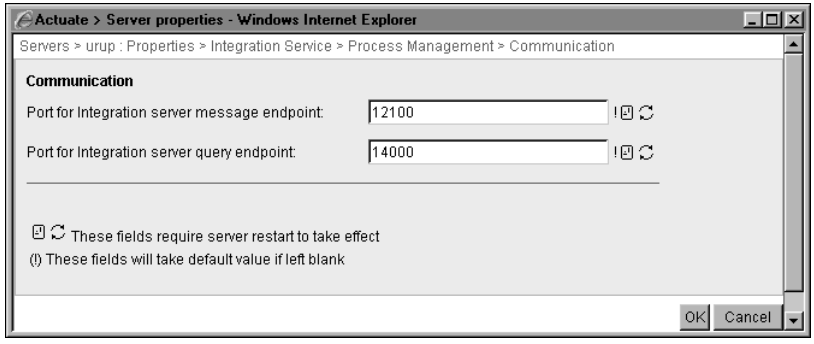

Figure 10-3 Setting port numbers for Integration service process communication

Specifying load settings

In Server Configuration Templates—Settings—Integration Service—Process Management—Requests, as shown in Figure 10-4, the administrator accepts or sets the value for each of the following properties:

- Max SOAP requests
  Maximum number of connections for communicating with Encyclopedia volume processes
- Max SOAP request threads
  Maximum number of threads for communicating with volume processes

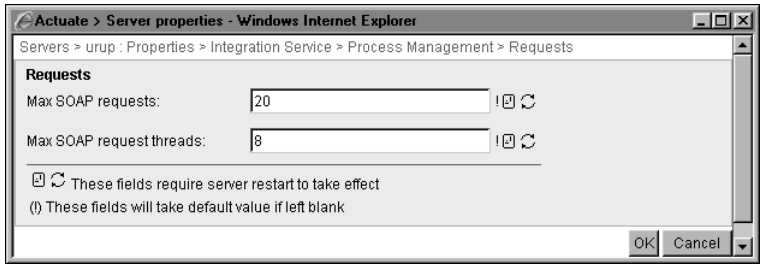

Figure 10-4 Specifying load settings

Managing Actuate SQL query execution

In Server Configuration Templates—Settings—Integration Service—General Data Source Information, shown in Figure 10-5, the administrator sets properties to specify the data source, including database collation.

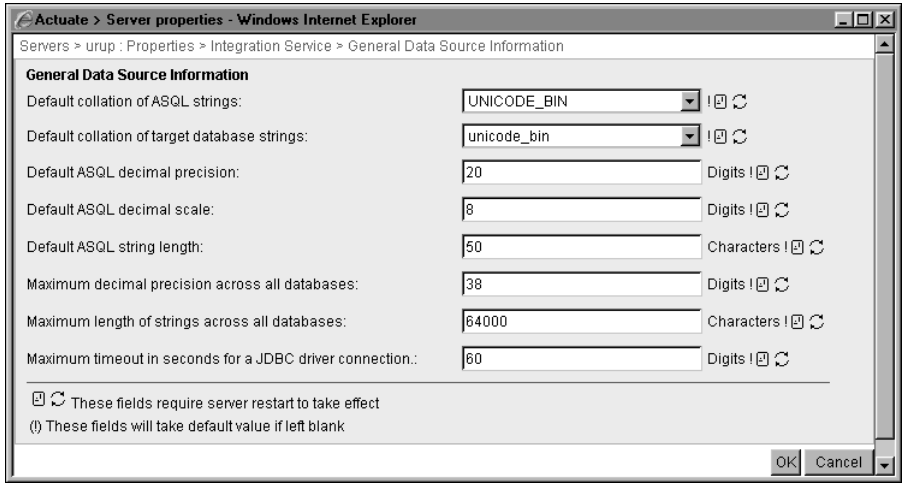

Figure 10-5 Specifying Integration service general data source information

Collation is an algorithm for ordering strings. When an Actuate SQL query executes, the type of collation determines the result of sort and comparison operations. Databases support one or more collations, usually determined by the database locale. The Integration service, however, supports only the Unicode and ASCII code-point collations, which order strings based on the Unicode or ASCII numbers corresponding to each character.

Based on properties the administrator sets on Servers—Properties—Advanced—Integration Service—General Data Source Information, the database collation and the Integration service collation determine which operations are sent to the database and which the Integration service performs. These properties are:

- Default collation of ASQL strings
  Specifies database collation of an Actuate SQL query. You can specify one of the following values:
  - UNICODE_BIN, the default, sets unicode code point order (binary order). All characters are different from one another and are sorted by their unicode values.
  - ASCII_CI sets code point order. Uppercase characters have the same value as lowercase characters.
- Default collation of target database strings
  Specifies the type of collation used by the database. Refer to your database documentation to determine the appropriate category for your database collation. You can specify one of the following values for the Default collation of target database strings property:
  - unicode_bin, the default, specifies that the collation of target strings is the same as the Integration service collation, UNICODE_BIN.
  - ascii_ci specifies that the collation of target strings is the same as the Integration service collation, ASCII_CI.
  - null specifies that the collation of target strings does not correspond to either the UNICODE_BIN or ASCII_CI used by the Integration service collation. Each character has a unique value.
  - null_sensitive also specifies that the collation of target strings does not correspond to either UNICODE_BIN or ASCII_CI used by the Integration service collation, but more than one character can have the same value, for example 'E' = 'e'.
- Default ASQL decimal precision
  Specifies the maximum number of digits after the decimal point in an Actuate SQL query.

- Default ASQL decimal scale
  Specifies the maximum scale for NUMERIC and DECIMAL types in an Actuate SQL query. For example, 15 represents decimals that can have up to 15 digits in all, including decimals after the decimal point.
- Default ASQL string length
  Specifies the maximum size in an Actuate SQL query for CHAR, VARCHAR, and LONGVARCHAR data types.
- Maximum decimal precision across all databases
  Specifies the AIS maximum decimal precision.

  If the value is greater than the cache database maximum precision, the cache database might return truncation errors when creating a cache due to a mismatch between the precision of the AIS data and the maximum precision of a cache database. To avoid truncation errors, use the Actuate SQL CAST( ) function to change the precision of the data.
- Maximum length of strings across all databases
  Specifies the maximum string length of AIS strings.

  If the value is greater than the cache database maximum string length, the cache database might return truncation errors due to the mismatch between the length of the AIS data and the maximum string length of a cache database. To avoid truncation errors, use the Actuate SQL CAST( ) function to change the length of the data.
- Maximum timeout in seconds for a JDBC driver connection
  Specifies how long iServer waits to connect to a database over JDBC. The JDBC driver never times out when connecting to the database if you set this value to 0.

Specifying query settings

In Server Configuration Templates—Settings—Integration Service—Query Management—Query Execution, shown in Figure 10-6, the administrator accepts or sets the value for each of the following properties:

- Maximum run time for the query
  Specify a non-zero value to set a limit on the amount of time a query can run.
- Maximum query size in rows
  Specify a non-zero value to set a limit on the number of rows a query returns.
- Max memory per query
  Specify a non-zero value to set a limit on the amount of memory a query can consume.

- Max Fetch Scroll Memory Size per Query
  Defines the maximum size of the memory buffer that binds to the ODBC per query. Change this value if there is sufficient memory on the machine. Use trial and error to determine the ideal setting.
- Max Rows Fetched Per Fetch Scroll
  Determines the total number of ODBC rows retrieved for each call of SQLFetchScroll and sets a limit on the memory usage per execution. Use trial and error to determine the ideal number of rows to retrieve per call.
- Max ODA Fetch Size Per Query
  Sets the fetch buffer size when fetching data from an ODA driver.

These settings are per connection, therefore all queries on the particular connection use the same setting. Using a value other than the default on upper limit could result in increased memory usage by iServer.

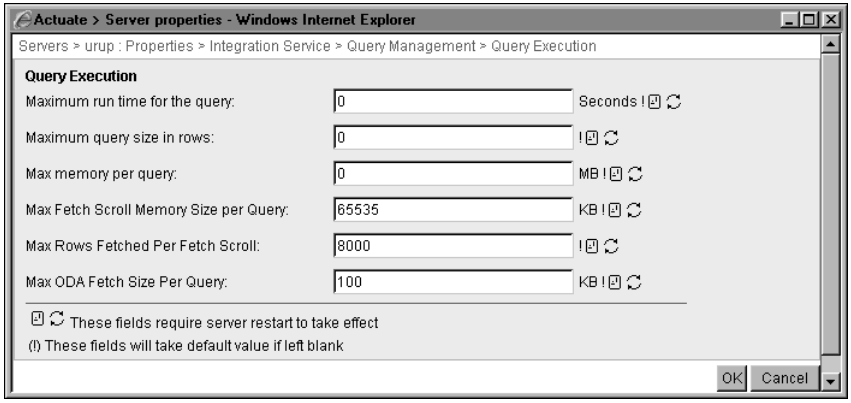

Figure 10-6 Specifying Integration service query settings

Specifying query optimization settings

In Server Configuration Templates—Settings—Integration Service—Query Management—Query Optimization, shown in Figure 10-7, the administrator sets properties to improve the overall system performance. iServer reduces the execution time and system resource usage for queries across multiple data sources.

The properties are:

- Enable cost-based optimization
  Determines whether to apply the cost-based query optimization.

  Change the default value when the query plans generated with cost-based optimization are inefficient or when you cannot provide the cost information for the information objects.

- Minimum rows to trigger creation of an index during materialize operation
  Creates an index on materialization that improves performance of searching for matched tuples in the materialization.

  When the materialization has more rows than this parameter value, iServer creates the index.

  Increase the default value when iServer executes too many concurrent queries, exhausting the memory for indexing and impacting overall system performance.

  Reduce the number of queries that invoke the creation of an index, thereby slowing the execution of some queries and reducing system resource usage.

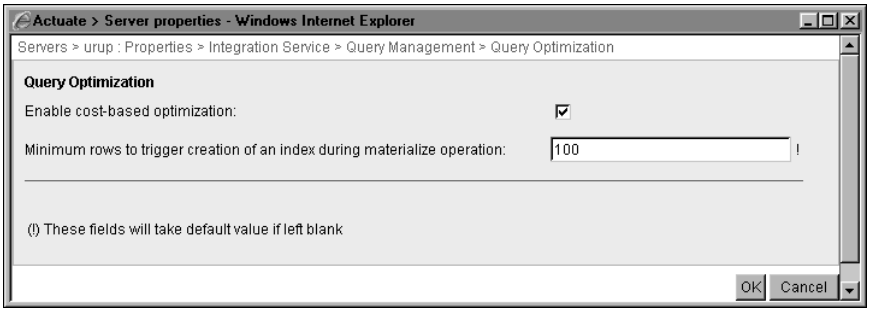

Figure 10-7 Specifying Integration service query optimization settings

Setting resource management properties

In Server Configuration Templates—Settings—Integration Service—Query Management—Resource Management, shown in Figure 10-8, the administrator accepts or sets values for each of the following Integration service resource properties:

- Temporary storage path for spill to disk
  Path to the directory that holds temporary files used by an Integration service process. The default directory is the temp subdirectory of the iServer home directory.
- Minimum Disk Threshold for spill partition
  Amount of memory that must be consumed before iServer writes overflow to disk.
- Page pool size
  The maximum page size required to process the information object data.
- Buffer pool size
  The maximum buffer pool size required to process information.

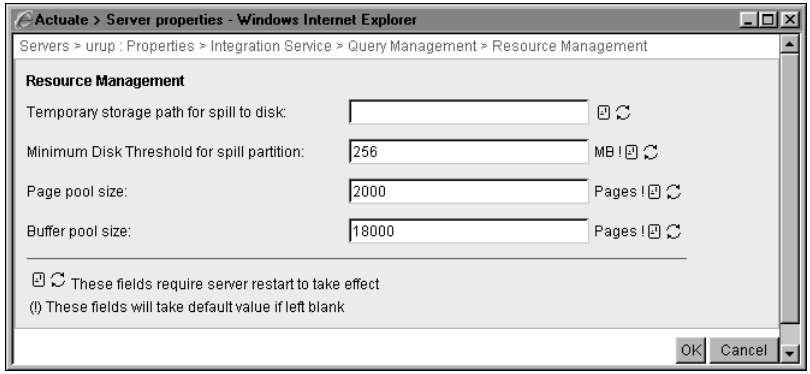

Figure 10-8 Specifying the Integration service resource property values

The Integration service process uses disk-based files to store temporary data when processing an information object that requires a large amount of memory.

When allocating disk space for the directory, consider the maximum amount of memory required to process the information object data and the maximum number of concurrent information objects that iServer can run.

Setting query statistics logging properties

In Server Configuration Templates—Settings—Integration Service—Query Management—Query Statistics, shown in Figure 10-9, the administrator accepts or sets the value for each of the following query statistics logging properties:

- Enable query statistics logging
  Enables and disables logging of the statistics.
- Query statistics log level
  - Standard logs query level statistics
  - Info logs query and Operator-level statistics
  - Detail logs query, Operator, and database-level statistics

  The cumulative execution time is logged at the Info level.

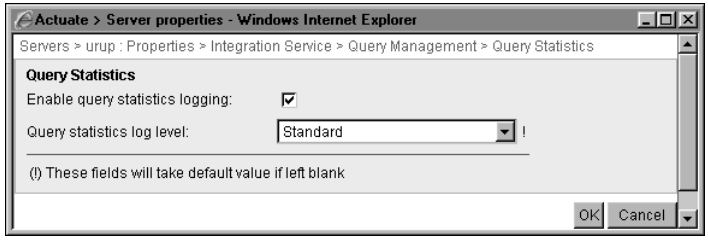

Figure 10-9 Specifying query statistics logging settings

Specifying the default Actuate Query template

In Volumes—Properties—Advanced—Actuate Query Generation, the administrator specifies the path to the default Actuate Query template to apply to Actuate Query output, as shown in Figure 10-10.

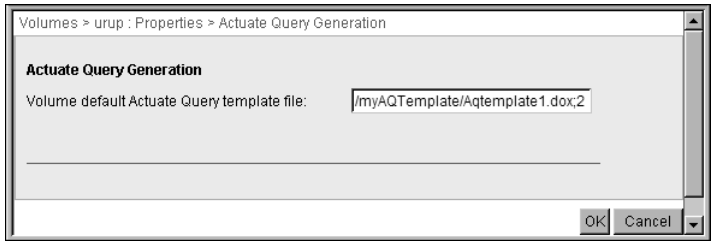

Figure 10-10 Specifying the default Actuate Query template file path

Specify the full path and name of an existing Actuate Query template in the Encyclopedia volume. iServer uses an Actuate Query template file to format output. iServer uses the default template file to format output when a user runs an Actuate Query using an information object that does not specify a template file.

If you do not specify a version number for the template, and multiple versions exist, iServer uses the latest version of the file. The following example specifies version 2 of a file called Aqtemplate1.dox:

```
/myAQTemplate/Aqtemplate1.dox;2
```

If the value of this property is missing, empty, or invalid, iServer uses the template file that installs with the Encyclopedia volume.

Using information objects

Information architects use information objects to control information retrieved from data sources, consolidate information from multiple data sources, and cache data from remote data sources for offline use.

To ensure data is available for an information object job, such as a BIRT design that uses information objects, you can cache data. Future queries using that information object use the data stored in the cache database.

iServer stores cached information objects in one or more of the following databases:

- DB2
- Oracle
- MS SQL

Caching information object data optimizes the timeliness of data, the load constraints of operational data stores, and response time. For example:

- Response time is faster when you execute large volumes of requests against data in the cache database instead of querying production data sources.
- Performance can improve when you:
  - Schedule population of the cache during non-peak traffic hours.
  - Populate the cache incrementally instead of retrieving all the rows of the output of an information object.
  - Retrieve increments to the output since the last time the cache was updated.

Setting up caching

In iServer Release 9, the Encyclopedia volume can cache data using a single cache connection to a database. In iServer Release 10 and later you can set up multiple cache connections to multiple databases.

About setup in Actuate Information Object Designer

In iServer Release 9, you run SQL scripts to create an information cache definition (ICD) file. In iServer Release 10 and later, an information architect creates an information cache definition file using Actuate Information Object Designer. For each ICD, the information architect specifies the name of its cache data table, the names of the columns of that table, and any table indexes.

About setup in iServer

From Configuration Console, you configure properties on Servers—Properties—Advanced—NetOSI File Types—ICD to communicate between the information object and Actuate Caching service (ACS) using SOAP. From Management Console, you activate the cache when scheduling a job. iServer executes the ICD schedule and populates the cache table. When you run a query against the information object, iServer retrieves data from the cache table.

You need to provide the information architect with the location and connection properties to use for caching, so the information architect can publish either information objects or designs that contain them to iServer. The information objects become available in the Encyclopedia volume, and designs can use them as data sources. Users can retrieve data directly from information objects by running Actuate Query.

Updating cache files

Actuate 11 supports caches and cache definitions created in Actuate 11 and later releases. Ensure that the ICD files in the Encyclopedia volume are updated regularly. To update an information object cache, run or schedule the ICD as a job using Management Console.

When iServer updates the cache data, it updates all of the data from the data source, not just the new or changed data. An information object can access updated information object data when the job completes. Until the job completes, the information object uses the existing cache data. When the information architect deletes cache definitions from an Information Object Designer project, you must delete the cache definitions from the Encyclopedia volume.

To move an information object project to a new location within the same Encyclopedia volume, use Management Console to move a folder to the new location.

Licensing Actuate Caching service

To use information objects with Actuate Query, you must first purchase and enable Actuate Query Option for iServer System. Using Actuate Caching service requires an iServer System license that enables the Information Object Caching Option and another license that enables the Data Integration Option. You cannot cache data for an information object that uses pass-through security.

About Actuate Caching service

Actuate Caching service manages one or more configurable ACS databases, and performs functions such as adding and dropping databases, tables, and indexes, and inserting data into databases. Actuate Integration service (AIS) and ACS are persistent, multithreaded processes that accept multiple simultaneous requests. A Factory process communicates directly with AIS through the ODA interface, not through the Message Distribution service (MDS).

In a stand-alone configuration, iServer runs single AIS and ACS processes. In a cluster, each node can run AIS and ACS processes. A cluster distributes information object jobs among the cluster nodes with AIS enabled. You start and stop the AIS and ACS processes by enabling or disabling the AIS and ACS services. When using ACS in a cluster, Actuate recommends enabling both ACS and the Factory service on the cluster node.

iServer uses the AIS to run an information object. Figure 10-11 shows the communication among components when using a cache database.

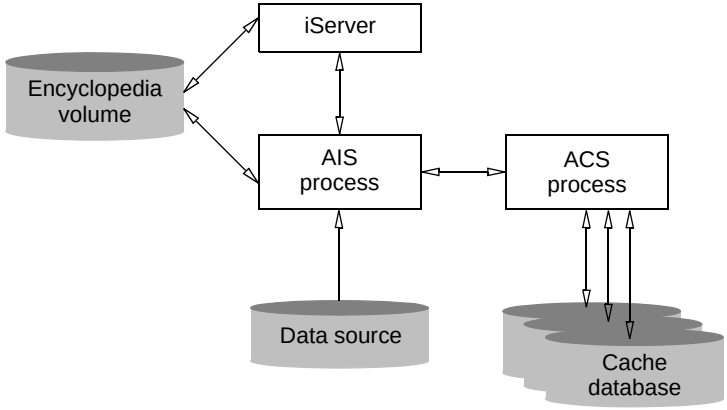

Figure 10-11 iServer component communication

Both the Factory process and an Actuate desktop application access an information object and the Actuate Integration service using the ODA interface. The AIS ODA driver supports the following design-time and run-time interfaces of ODA:

- The ODA driver requests a connection to an iServer, then creates a connection to that host if it has AIS enabled.
- For a Factory process, the ODA driver uses the AIS server specified by the configuration. The default configuration specifies that all Factory processes use the AIS server.

The following types of files play a role in caching information objects:

- DCD (Data connection definition)
  Contains properties to identify and connect to a particular external data source, such as a database or an external application.

  Properties consist of data source type, connection properties, and pass-through security type.
- ICD (Information object cache definition)
  Contains configuration information for caching data that a data source map SMA or information object IOB uses. Each cache is stored in an external DBMS configured for the Encyclopedia volume.

  Properties consist of SMA information and cache state information.
- IOB (Information object)
  Presents a view of a data source, a logical set of data from other maps or information objects. Supports row-level security.

Properties consist of schema information such as parameter and column names, other properties such as a query, editing state, and caching configuration.

- SMA (Data source map)
  Represents a single set of data from a data source, such as a database table.

  Properties consist of schema information such as parameter and column names, information to access a data source, and caching configuration.

You cannot use more than one version of a DCD, ICD, IOB, or SMA file in an Encyclopedia volume. If an information object uses another information object or map as input, that relationship information is internal to the information object and is specified only by name. The information object does not use Encyclopedia version information, such as file ID or version number.

If you enable the Factory service and disable the ACS on a cluster node, it must use another node's Caching service. To specify a node having the Caching service enabled, use either one of the following methods to modify the configuration of the node that has the disabled Caching service:

- Change the hostname parameter in Servers—Properties—Advanced—NetOSI File Types—ICD to the name of the machine with the ACS enabled. The default hostname value is localhost.
- Remove the ICD file type from the server node using Servers—Properties—Advanced—NetOSI File Types—ICD (Delete).

Configuring the Caching service

The administrator can improve the database performance by using the Actuate Caching service (ACS) to cache information object data in the iServer environment. Caching information object data eliminates repetitive queries, reducing the load on the network and data source.

In Servers—Properties—Caching Service, the administrator can change the following properties:

- Caching service port
  iServer uses this port for communication between the ACS process and Encyclopedia volume.
- Caching service request settings
  The maximum number of SOAP requests and request threads for communicating with an Encyclopedia volume process.
- Bulk load settings
  Path to the directory of the intermediate files ACS uses to perform a bulk load to a data source, such as an Oracle, DB2, or SQL Server database. iServer saves

temporary files only when an ACS bulk load job fails. Keep temp files instructs iServer to always or never save temporary files or only save these files when a job fails.

Figure 10-12 shows the selections the administrator makes in Server Configuration Templates—Settings to configure properties for the Caching service.

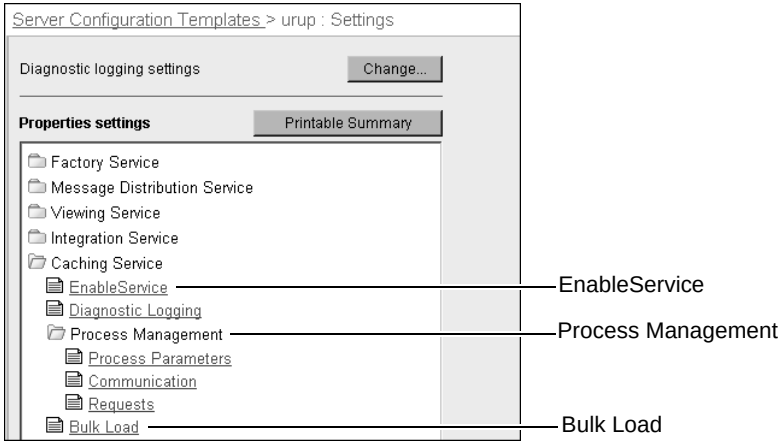

Figure 10-12 Configuring properties for the Caching service

The properties that appeared in Servers—Properties—Caching Service in previous iServer releases appear in Server Configuration Templates—Settings in Release 11. In Release 11, set Caching service properties in Server Configuration Templates—Settings as follows:

- To set Caching service port, choose Caching Service—Process Management—Communication.
- To set Caching service request settings, choose Caching Service—Process Management—Requests.
- To set bulk load settings, choose Caching Service—Bulk Load.

Enabling the Caching service

The administrator can enable or disable the Caching service from Server Configuration Templates—Settings, or by setting the EnableCachingService parameter in AC_DATA_HOME/config/11SP<service pack number> / acserverconfig.xml to true or false.

How to enable the Caching service

1 Expand Caching Service and choose EnableService, as shown in Figure 10-12.

2 For Enable Caching service, accept the default value, which is selected, as shown in Figure 10-13.

Figure 10-13 Enabling or disabling the Caching service

About diagnostic logging

The administrator can configure diagnostic logging by expanding Caching Service, and choosing Diagnostic Logging. For more information, see Chapter 3, "Using diagnostic, usage, and error logging."

Configuring Process Management properties

The Administrator configures Process Management properties for the Caching service by setting property values in the following Process Management categories:

- Process Parameters
- Communication
- Requests

Configuring Process Parameters

In Process Parameters, the administrator sets the startup parameters for Caching service processes property.

How to configure Process Parameters

1 Expand Caching Service, Process Management, as shown in Figure 10-12, and choose Process Parameters.

2 In Start parameters for Caching service processes, accept the default value, or alternatively, modify the value, as shown in Figure 10-14.

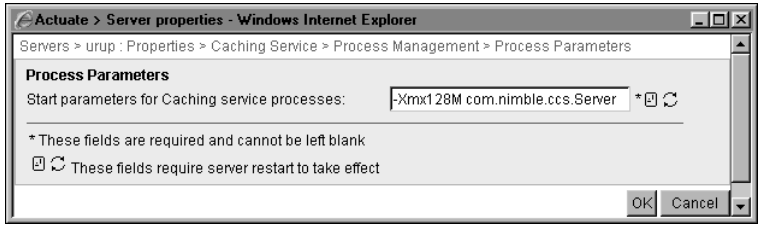

Figure 10-14 Setting the start parameters for the Caching service

Choose OK.

3 Restart iServer.

Configuring Communication

In Communication, the administrator sets the Port for Information Object Caching server messages property.

How to configure Communication

1 Expand Caching Service, and Process Management, as shown in Figure 10-12, and choose Communication.

2 In Port for Information Object Caching server messages, accept the default value, 11550, or alternatively, specify a different value, as shown in Figure 10-15.

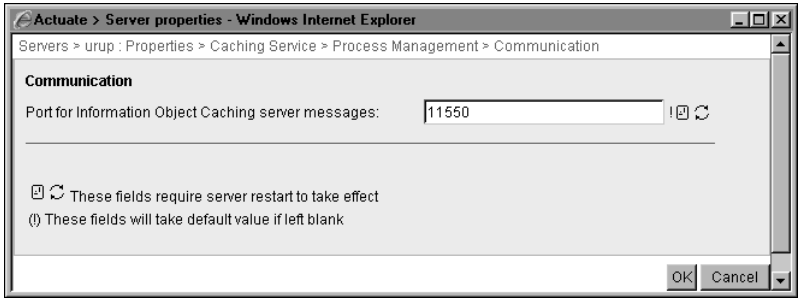

Figure 10-15 Configuring the Communication property

Choose OK.

3 Restart iServer.

Configuring Requests

In Requests, the administrator sets the following properties:

- Max SOAP requests
- Max SOAP request threads

How to configure Requests

1 Expand Caching Service, and Process Management, as shown in Figure 10-12, and choose Requests.

2 In Max SOAP requests, accept the default value, 20, or alternatively, specify a different value, as shown in Figure 10-16.

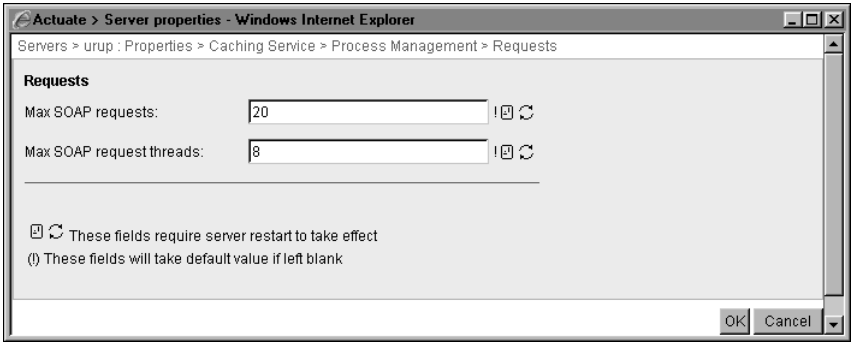

Figure 10-16 Configuring the Requests property

3 In Max SOAP request threads, accept the default value, 8, or alternatively, specify a different value.

Choose OK.

4 Restart iServer.

Configuring Bulk Load

In Bulk Load, the administrator sets the following properties:

- Keep temporary bulk load files
- Client bulk load path

How to configure Bulk Load

1 Expand Caching Service, and choose Bulk Load, as shown in Figure 10-12.

2 In Keep temporary bulk load files, accept the default value, or alternatively, specify a different value, as shown in Figure 10-17.

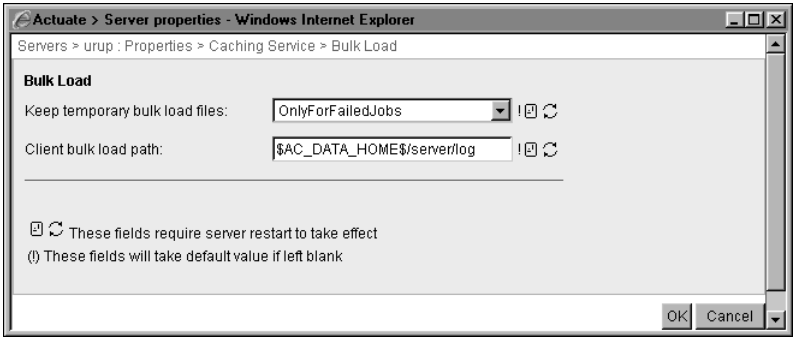

Figure 10-17 Configuring the Bulk Load property

3 In Client bulk load path, accept the default path, or alternatively, specify a different path.

Choose OK.

4 Restart iServer.

Table 10-1 lists the Caching service properties, and relates the property names that appear in Configuration Console with the corresponding parameter names in acmetadescription.xml, indicating default settings, ranges, and when a property change takes effect.

Table 10-1 Actuate Caching service parameters

Property name	Parameter name	Default	Range	Takes effect
Client bulk load path	BulkLoadPath			Server Restart
Enable caching service	EnableCachingService	False		Fixed
Keep temporary bulk load files	KeepTempFiles	OnlyForFailed-Jobs	OnlyForFailed-Jobs Always Never	Server Restart
Max SOAP requests	MaxConnections	20	2 - 1024	Server Restart
Max SOAP request threads	MaxThreads	8	2 - 200	Server Restart
Start parameters for Caching service processes	StartArguments	-Xmx128M com.nimble.ccs.Server		Server Restart
Port for Information Object Caching server messages	SOAPPort	11550	1024 - 65535	Server Restart

About bulk loading files to the cache

Use database tools, such as sqlldr and db2cmd, for Oracle and DB2 respectively, for bulk loading. Install these tools on the node where the caching server is enabled. iServer also supports bulk loading through the JDBC driver using jdbc insert. To prevent problems with client tools, such as the SQL Server Bulk Copy Program (BCP), specify a fully qualified name for a database object, using database.schema.object notation, as shown in the following example of BCP command-line syntax:

```
bcp {[[database_name.][schema].]{table_name | view_name} |
   "query"}
     {in | out | queryout | format} data_file
     [-S server_name[\instance_name]] [-U login_id] [-P password]
```

Configuring Actuate Caching service and NetOSI file type

The Encyclopedia volume information object cache definition, ICD, stores information object data in a cache. iServer uses properties that you configure on Server Configuration Templates—Settings—NetOSI File Types—ICD to communicate between the information object and Actuate Caching service (ACS) using SOAP.

Choose Server Configuration Templates—Settings—NetOSI File Types (Add) to add a new NetOSI interface.

Figure 10-18 shows Server Configuration Templates—Settings—NetOSI File Types (Add).

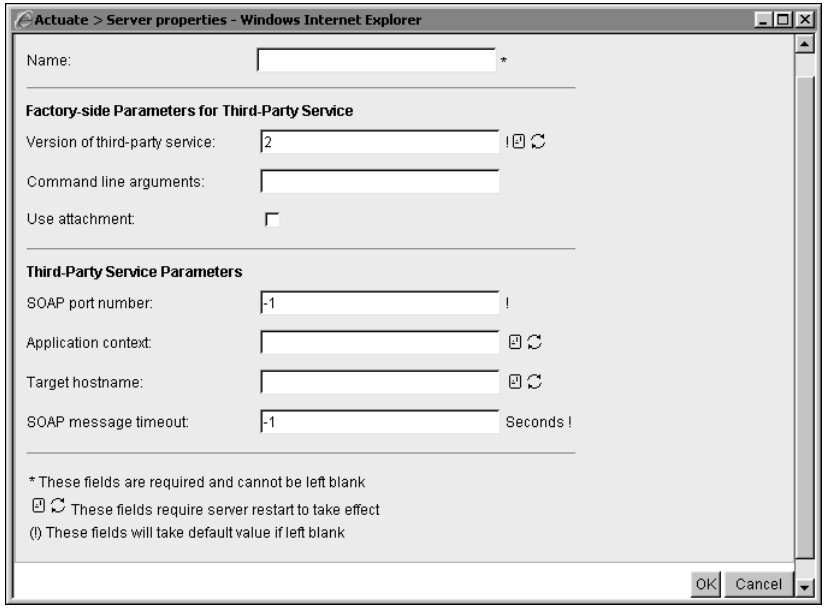

Figure 10-18 Adding a NetOSI interface

Table 10-2 describes the properties for adding a NetOSI interface.

Table 10-2 Properties for adding a NetOSI interface

Property	Value
Name	Name of the NetOSI file type. iServer displays the name in the list of file types.
Version of third-party service	Actuate recommends contacting Actuate Customer Support before you change this value.
Command line arguments	Command line arguments for the open server driver. iServer uses these arguments when the open server SOAP interface and a Factory process need to start the open server service.
Use attachment	Indicator of the type of file transfers between Factory process and open server service. iServer uses this setting in conjunction with the open server SOAP interface. Select this property to send open server files as attachments instead of specifying a link to the path. If the Factory process and open server service are on different machines, linking to the files is impossible. You must use attachment mode.
SOAP port number	Port number to connect with the open server service that is running on an application server. The default value, -1, uses the default port.
Application context	String used as the application context when sending a SOAP message to the open server service deployed on the application server.
Target hostname	IP address or name of the host machine that hosts the open server service.
SOAP message timeout	Seconds to wait for SOAP messages between the Factory process and open server service. The default value, -1, disables time-out. If the value is larger than the document request time-out, the Factory process aborts the request. If the time-out value is too small, the Factory process breaks the connection before the open server service is able to respond.

You can delete a NetOSI file type in Server Configuration Templates—Settings—NetOSI File Types—ICD (Delete).

Chapter

11

Configuring iServer security

This chapter contains the following topics:

- Understanding the Report Server Security Extension
- Installing iServer using Open Security
- Understanding LDAP configuration
- Working with RSSE page-level security

Understanding the Report Server Security Extension

The Report Server Security Extension (RSSE) supports Open Security and page-level security. Open Security is the framework that a developer uses to create an interface to an external security source, such as Lightweight Directory Access Protocol (LDAP) or Microsoft Active Directory. Using the interface, the Encyclopedia volume controls access using information from the external security source.

Using page-level security, a developer can create an RSSE application that associates security IDs in the access control list (ACL) of a design to Encyclopedia volume users or roles. To use page-level security when working with BIRT designs and documents, obtain a license for the BIRT Page Level Security Option. To use page-level security when working with Actuate Basic designs and documents, obtain a license for the e.Report Page Level Security Option. iServer Integration Technology contains reference implementations of RSSE applications, which include source code and JAR files.

iServer supports the following types of SOAP-based RSSE applications that you can install with iServer:

- External authentication
  Authenticates users in the Encyclopedia volume based on an external, third-party security system, such as LDAP.
- External registration
  Control access to Encyclopedia volume items based on an external, third-party security system, such as LDAP. With this strategy, you externalize users, roles, groups, and user properties.
- Page-level security
  Controls user access to sensitive information in a document by implementing page-level security.

To set up a Release 11 iServer that uses RSSE to connect to an LDAP Directory Server, the administrator must first install an iServer specifying the custom security source option. Then, using Configuration Console, the administrator configures an Encyclopedia volume to use a web service that supports RSSE processing.

Working with RSSE

After installing iServer using the custom security source option, iServer is ready to use RSSE implementation. To prepare an LDAP security source to interoperate with iServer, configure the security source by populating it with the appropriate security information.

In Configuration Console, configure the Encyclopedia volume for RSSE by specifying the web service parameters in Volume—Properties—Open Security. The SOAP-based RSSE application runs as a web service from the iServer application container. Developers can create a custom RSSE application that uses other data sources. For more information on creating custom RSSE applications, see the Server Integration Technology RSSE reference implementations.

About iServer and RSSE application interaction

Figure 11-1 illustrates the communication between iServer, the RSSE application as a web service, and the external security source.

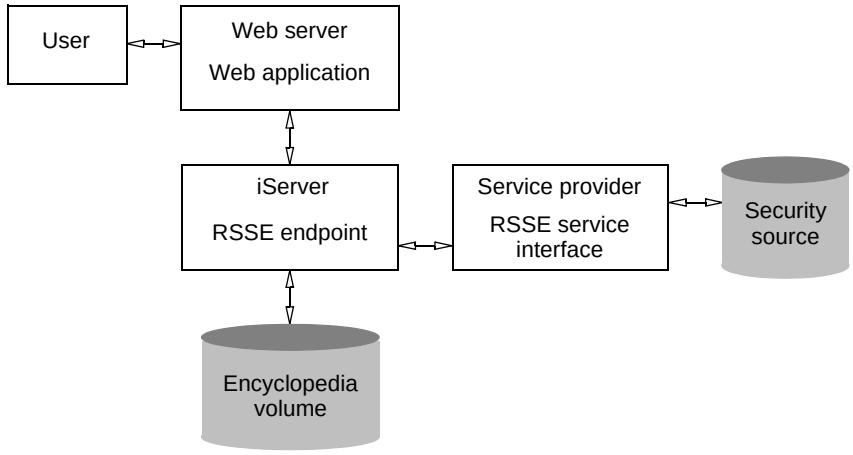

Figure 11-1 iServer communication flow diagram

The service runs from the application container that ships with iServer. The installation program installs the application container with the following components:

- Management and Configuration Consoles
- RSSE web services application

The application container, iServer Consoles, and the RSSE web services application are installed with iServer under the AC_SERVER_HOME directory in the following subdirectories:

- The application container is in the servletcontainer directory.
- iServer Consoles are in the servletcontainer/mgmtconsole directory.
- The RSSE web services application is in the servletcontainer/webapps/acrsse directory.

iServer implements the RSSE application as a Java web service using Apache Axis.

Installing iServer using Open Security

After installing and configuring an LDAP Directory Server, install a new iServer with an Encyclopedia volume that uses LDAP and configure the SOAP-based Java RSSE web service.

Installing iServer with the LDAP Option

The following example installs an iServer that sets up an Encyclopedia volume that uses an LDAP Directory Server and an SOAP-based Java RSSE web service in a Windows environment.

How to install iServer using the RSSE Option

If you downloaded iServer, run ActuateBIRTiServerEnterprise.exe. If you have an ftp distribution, run setup.exe. The welcome message appears, as shown in Figure 11-2. Choose Next.

Figure 11-2 Viewing the welcome message

1 Follow the setup instructions in the Installing BIRT iServer for Windows or Installing BIRT iServer for Linux and UNIX manual, except choose a Custom setup type, select all features, and select the Stand-alone installation option. When you reach Select Security Source, select Use an LDAP Directory Server, as shown in Figure 11-3. Choose Next.

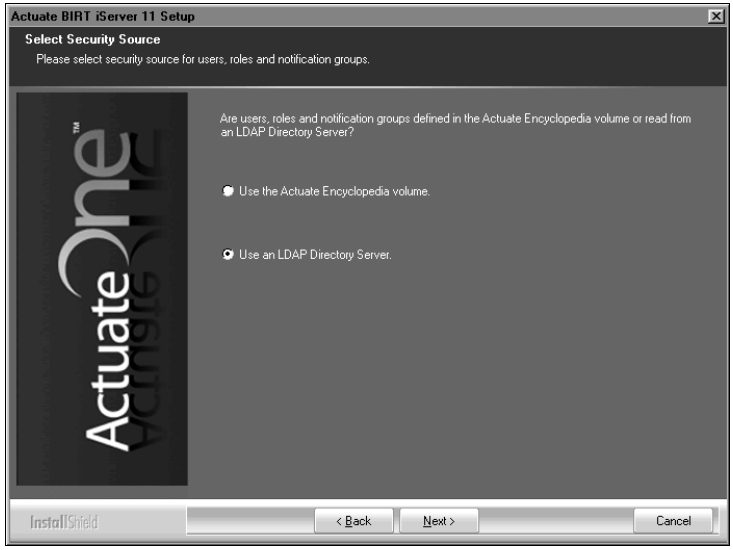

Figure 11-3 Selecting a security source

2 In LDAP Configuration, specify the server name where the LDAP Directory Server is running and the port number where it listens, as shown in Figure 11-4. Choose Next.

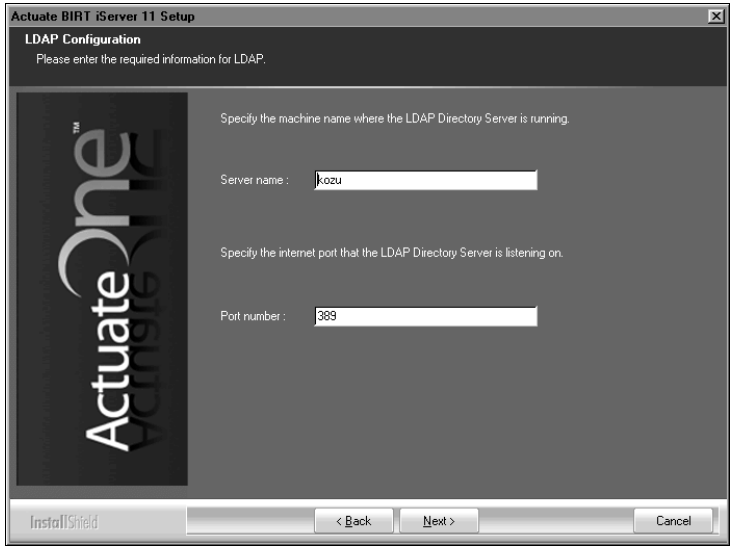

Figure 11-4 Specifying the LDAP directory server name and port

3 Specify the query account and password, as shown in Figure 11-5. The Query account is for anonymous operations to the LDAP Directory Server, such as validation. Choose Next.

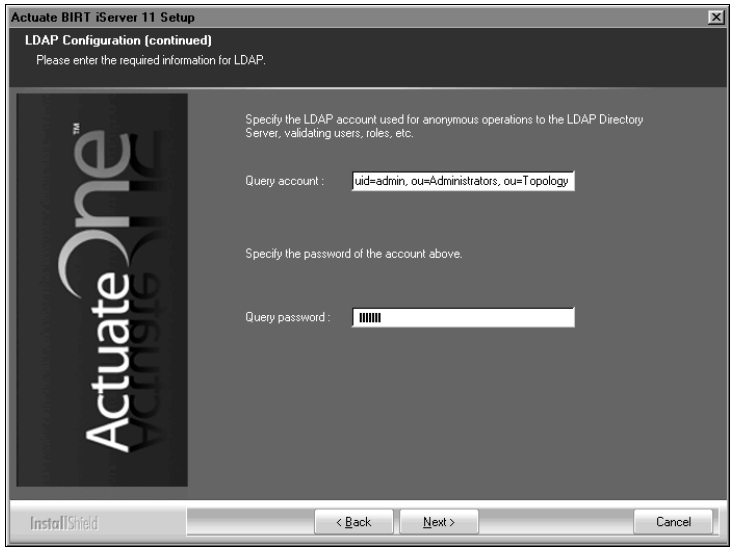

Figure 11-5 Specifying the query account and password

4 Specify the name of the organization, as shown in Figure 11-6. Choose Next.

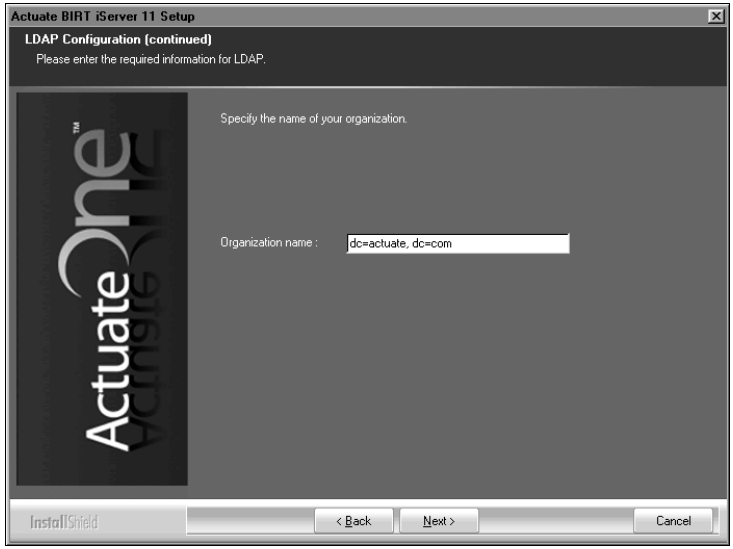

Figure 11-6 Specifying the organization name

5 Specify the base domain used for queries of users and the object used as a filter for queries of users, as shown in Figure 11-7. Choose Next.

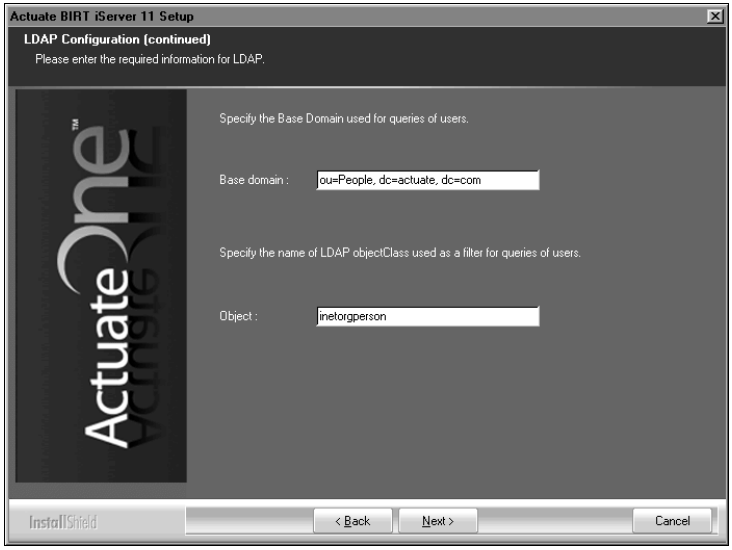

Figure 11-7 Specifying the base domain and object for queries of users

6 Specify the base domain used for queries of roles and specify the object used as a filter for queries of roles, as shown in Figure 11-8. Choose Next.

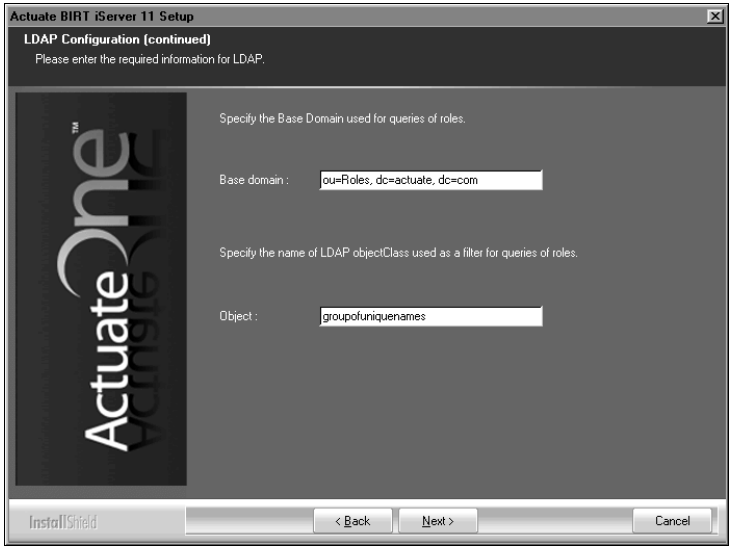

Figure 11-8 Specifying the base domain and object for queries of roles

7 Specify the base domain used for queries of groups and specify the object used as a filter for queries of groups, as shown in Figure 11-9. Choose Next.

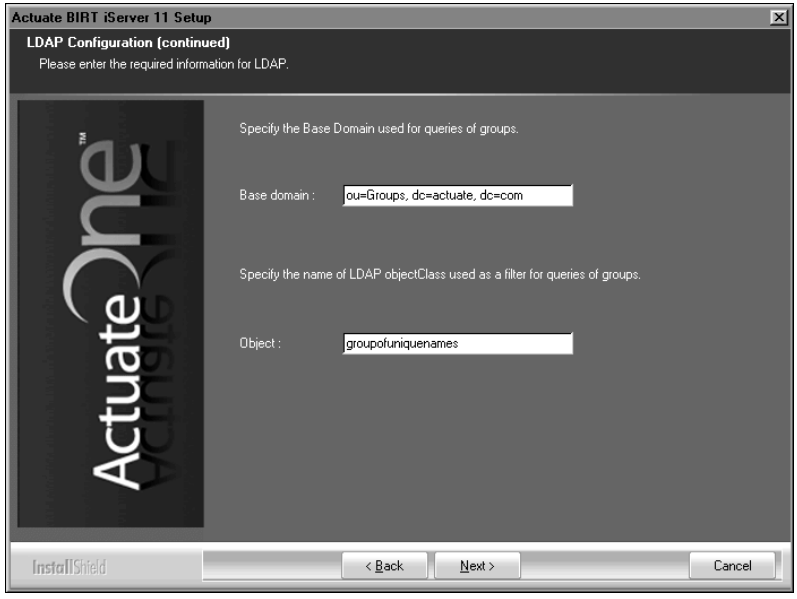

Figure 11-9 Specifying the base domain and object for queries of groups

8 Specify the name of a user with an administrator role and the name of a user with an operator role, as shown in Figure 11-10. Choose Next.

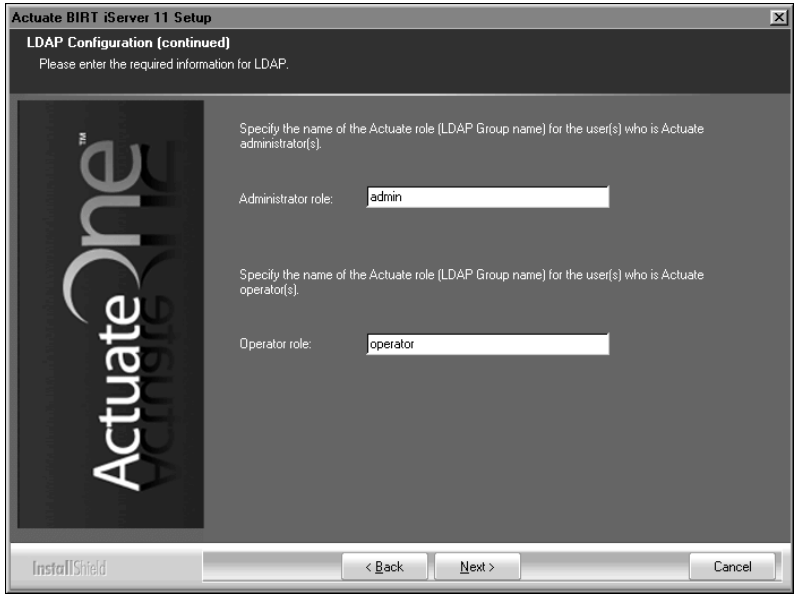

Figure 11-10 Specifying the names of users with administrator and operator roles

After the installation, inspect the ldapconfig_<VolumeName>.xml configuration file, which the installation program creates in AC_SERVER_HOME/etc.

Installing the RSSE web service application

For an Encyclopedia volume, the RSSE web service configuration information is in the ldapconfig_<volume>.xml file, where <volume> is the name of the Encyclopedia volume using the RSSE web service application. For example, the configuration file for the sales1 volume is:

```
ldapconfig_urup.xml
```

The ldapconfig_<volume>.xml configuration file is in the following directory:

```
AC_SERVER_HOME/etc
```

Each volume that uses the RSSE web service application must have a configuration file.

The iServer application container that runs the RSSE web service application is in the following directory:

```
AC_SERVER_HOME/servletcontainer/webapps/acrsse
```

To configure the RSSE web service application, change the values in the ldapconfig_<volume>.xml file, and restart the iServer application container that runs the RSSE web service application. For more information about setting ldapconfig_<volume>.xml parameters, see “Setting ldapconfig_<volume>.xml parameters,” later in this chapter.

Using the RSSE application with a service provider

The RSSE web service application runs using a service provider that supports Apache Axis, such as Apache Tomcat. Use the RSSE web service application with an Apache Tomcat servlet container by performing the following steps:

- Install the RSSE web service application on the Tomcat servlet container.
- Configure the Encyclopedia volume to use the RSSE web service application.

The following sections briefly describe how to use the RSSE web service application with a Tomcat servlet container.

Installing an RSSE application on Tomcat

To install the web service RSSE application on a Apache Tomcat servlet container, copy the files from the iServer application container directory to the Tomcat directory.

On a Windows system, the default installation places the RSSE web services application in the following directory:

```
AC_SERVER_HOME\servletcontainer\webapps\acrsse
```

Copy the acrsse directory to the Tomcat webapps directory. For release 7.0 of Tomcat, the directory is:

```
\Program Files\Apache Tomcat 7.0\webapps
```

Stop and restart Tomcat to update the configuration. Check the HTTP server log files to ensure proper start-up of the application.

Configuring the Encyclopedia volume to run RSSE

Configure each Encyclopedia volume to run an RSSE web service application separately. A SOAP-based Java RSSE application runs as a web service in the iServer servlet container. To run SOAP-based RSSE applications on multiple Encyclopedia volumes, configure a separate location for each RSSE application.

Install an RSSE application on an Encyclopedia volume to run in its own location on iServer by performing the following tasks:

- Make a copy of the AC_SERVER_HOME/servletcontainer/webapps/acrsse directory.
  For example, copy the directory to the following location:
  ```
  AC_SERVER_HOME/servletcontainer/webapps/acrsse_AUTH
  ```
- Copy the file, rsseAcl.jar, to the lib directory of the servlet container in the following location:
  ```
  AC_SERVER_HOME/servletcontainer/webapps/acrsse_AUTH/WEB-INF/lib
  ```
- Extract the file, class.properties, from the archive file, rsseACL.jar, to the following location:
  ```
  AC_SERVER_HOME/servletcontainer/webapps/acrsse_AUTH/WEB-INF
     /classes/com/actuate11/rsse/wsdl
  ```
  If necessary, create the subdirectories, /classes/com/actuate11/rsse/wsdl, manually or using an archive extraction tool create the subdirectories when extracting the class.properties file.
- Use a source code editor to open the class.properties file and change the code reference specification to:
  ```
  class=com.actuate11.rsse.authSample.SampleRSSE
  ```

Configuring Open Security

Use Configuration Console to enable a RSSE web service application for an Encyclopedia volume. The following parameters appear on Volumes—Properties—Open Security, as shown in Figure 11-11:

- Do not enable
  Disables Open Security and the RSSE application.

- Enable as a web service
  Enables Open Security and the RSSE application.
- Cache timeout
  Sets the maximum time, in minutes, before iServer deletes cached open security data. The minimum cache time-out period is 1 minute. Set to -1 to specify that the cache never expires.
- IP address
  Sets IP address or machine name of the server that runs the RSSE web service.
- Soap port
  Establishes the port number for the RSSE web service.
- Context string
  Specifies the location of the RSSE web service for iServer to use when sending messages to the web service. The path for the default Encyclopedia volume is /acrsse/servlet/AxisServlet.

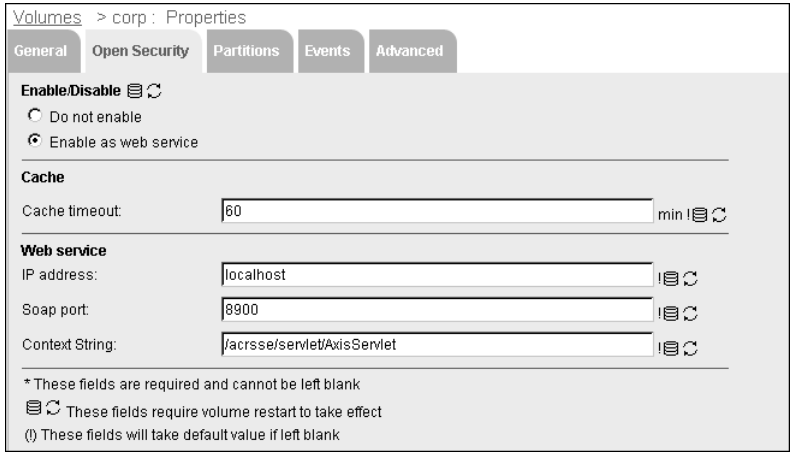

Figure 11-11 Specifying open security property values

How to enable Open Security for an Encyclopedia volume

1. Create a configuration file for the RSSE web service application.

   The configuration file maps the Encyclopedia volume management information to LDAP security directory information.
2. From the Advanced View of Configuration Console, choose Volumes.
3. In Volumes, point to the arrow next to the Encyclopedia volume name and choose Properties.

4 In Volumes—Properties—General, choose Open Security, as shown in Figure 11-11.

5 In Volumes—Properties—Open Security, choose Enable as a web service.

6 Specify web service parameter values as necessary. Choose OK.

7 On Volumes, point to the arrow next to the Encyclopedia volume name and choose Put offline.

8 Log out of Configuration Console. Stop and start the iServer Application container using the StartMC and StopMC scripts in AC_SERVER_HOME/bin. Log back into Configuration Console. Take the Encyclopedia volume online.

Test whether the RSSE web service application is working by logging in to the Encyclopedia volume or, if using a page-level security, by deploying a design file to the Encyclopedia volume.

If you are using a page-level security and change the assignments in the users.acls file, be sure to wait for the volume cache time-out period or recycle the volume before checking to see if the changes are effective. Using Configuration Console, you can re-configure the volume cache time-out period by going to Servers—Properties—View Service. On View Service, in View process cache for executables, the default value for Max cache timeout is 3600 seconds.

In View Service, in Extended viewing cache, you can also set Cache level to Cache only page-level security requests to retain previously viewed documents that have page-level security enabled. In Extended viewing cache, the default value for Cache timeout is 1200 seconds.

Specifying RSSE service startup parameters

Volumes—Properties—Advanced—Security Extension—Service, shown in Figure 11-12, allows the administrator to supply an operating system command that iServer uses to start the RSSE service. The administrator can also specify license options that iServer assigns to users by default when upgrading an RSSE-enabled volume from an iServer version that has no user-based licensing.

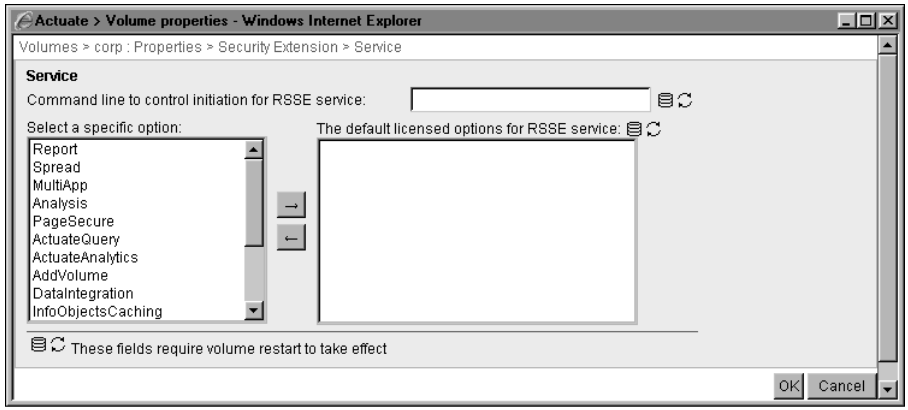

Figure 11-12 Specifying RSSE startup command and default license options

Table 11-1 describes the parameters that appear on Volumes—Properties—Advanced—Security Extension—Service.

Table 11-1 RSSE service startup parameters

Parameter	Description
Command line to control initiation for RSSE service	Command that iServer issues to start the web service for RSSE, if the RSSE service is not running. If iServer cannot connect to the web service for the RSSE service, iServer starts an operating system shell and runs the value of this parameter as a command-line request.
The default licensed options for RSSE service	Supports specifying license options that iServer assigns to all users by default when upgrading an RSSE-enabled volume from an iServer version that has no user-based licensing. The administrator is free to change license option assignments after the upgrade. Assigning default license options in The default licensed options for RSSE service does not affect the list of available options from which to select when assigning options to a user in Management Console.

Understanding LDAP configuration

When iServer uses the RSSE application, Encyclopedia volume users that are defined in the LDAP server must have a password. The user information must use:

- A single LDAP organization
- The appropriate LDAP object classes

The ldapconfig_<volume>.xml file defines the mapping between Encyclopedia volume user information and LDAP information. Encyclopedia volume user management information typically maps to LDAP information in the following way:

- The Encyclopedia volume users, groups, and roles map to LDAP object classes. For example, you can specify that Actuate users map to the LDAP inetorgperson object class and that roles and groups map to the LDAP groupofuniquenames object class.
- Individual users, roles, and groups are LDAP objects. For example, after specifying how Actuate users map to the LDAP inetorgperson object class of an organizational unit, you create LDAP users based on the inetorgperson object class. In LDAP, you can create users named MBarajas and JThompson based on the inetorgperson object class. MBarajas and JThompson are Encyclopedia volume users.
- Actuate user properties, such as e-mail address and home folder, are specified in LDAP attribute-value pairs of the LDAP object class. For example, after specifying that the Actuate users are based on the inetorgperson object class, you specify that Actuate e-mail maps to the mail attribute of the LDAP inetorgperson object class. You create object class attributes to store Actuate user properties.

In LDAP, you can also create a set of Encyclopedia volume users, notification groups, and roles by creating the LDAP objects within different LDAP organizational units and specifying LDAP distinguished names to point to the different LDAP organizational units.

Mapping Encyclopedia volume management information to LDAP objects

The following topics discuss how to map particular types of Encyclopedia volume management information:

- Mapping Encyclopedia volume user properties
- Mapping roles
- Mapping groups
- Mapping channels
- Mapping pass-through security information

Mapping Encyclopedia volume user properties

Encyclopedia volume users and their properties map to attributes of an LDAP object. User properties include login name, password, e-mail address, and default privilege template. For example, if Actuate user information maps to an LDAP

inetorgperson object class, the user properties map to the LDAP inetorgperson object class attributes.

All LDAP user object attributes are string attributes. To specify multiple values, the LDAP user object attributes for a privilege template list and an Encyclopedia volume channel membership list must be multivalued. For example, to specify that a user belongs to multiple volume channels in LDAP, the user object Encyclopedia volume channel attribute must be multivalued with an Actuate channel name for each value.

The Encyclopedia volume login name is the user ID attribute of the LDAP user object, called the uid attribute. The volume password is the password attribute of the LDAP user object. When using an RSSE application, a user must log in to an Encyclopedia volume using a password.

To indicate that an Actuate user is a member of an Actuate role or an Actuate notification group, add the LDAP Actuate user as a member of the appropriate LDAP Actuate group or role.

Actuate uses the default value for an Actuate user property in ldapconfig_<volume>.xml when:

- The LDAP server does not contain a definition for the LDAP user object attribute.
- The LDAP attribute for a user object does not contain any values.

Mapping roles

Encyclopedia volume roles map to an LDAP object class, such as groupofuniquenames. The name Actuate displays is the LDAP object's common name attribute, called the LDAP cn attribute.

When using the RSSE application with an LDAP server, you cannot nest roles. Roles are an LDAP object. To indicate that an Actuate user is a member of one or more Actuate roles, add the LDAP Actuate user object as a member of the LDAP Actuate roles.

iServer uses the members specified in the LDAP Actuate role objects when it performs authorization functions for Actuate roles. iServer also uses the LDAP role objects when it lists roles used to specify privileges.

To specify privileges to access a file or folder in the Encyclopedia volume for an LDAP role object, first create the Actuate role object in the LDAP directory. Then, in the Encyclopedia volume, specify privileges for the Actuate role in the Encyclopedia volume file or folder.

Information Console supports using different levels of user functionality based on a user's membership in particular Encyclopedia volume roles. To use the Information Console roles and functionality levels, create corresponding LDAP roles and add users to the roles.

Mapping groups

Encyclopedia volume groups map to an LDAP object class, such as groupofuniquenames. The name Actuate displays in the Encyclopedia volume is the LDAP object cn attribute.

If the e-mail notification group maps to the LDAP groupofuniquenames object class, the LDAP group objects do not require Actuate-specific attributes. To specify that an Actuate user is a member of an Encyclopedia volume group, first create an LDAP group object. Then, add the LDAP user object as a member of the LDAP group object.

Mapping channels

The Encyclopedia volume stores channel names. An LDAP user object attribute specifies the list of channels to which an Actuate user subscribes. The RSSE application does not verify that the Encyclopedia volume's channel names match the LDAP user object channel attribute values. You must ensure that the channel names match the values in the LDAP user object channel attribute.

Mapping pass-through security information

When a user runs an information object that uses pass-through security, iServer requires a database user name and password. When using the RSSE application, you specify two LDAP user object attributes that iServer uses with pass-through security. One attribute contains the value for the database user name and the other attribute contains the database password. The following example shows pass-through security parameters in the LDAP configuration file:

```
<ConnectionPropertyList>
   <ConnectionProperty>
      <Name>username</Name>
      <Value>acdbname</Value>
   </ConnectionProperty>
   <ConnectionProperty>
      <Name>password</Name>
      <Value>acdbpassword</Value>
   </ConnectionProperty>
</ConnectionPropertyList>
```

The ConnectionPropertyList contains ConnectionProperty elements. The ConnectionProperty name-value pairs specify the LDAP user object attributes that contain the database user name and password that iServer uses when a design accesses data through an information object.

The value for username is the LDAP user attribute that contains the database user name. In the following example, iServer uses the value in the LDAP user's dbname attribute as the database user name:

```
<ConnectionProperty>
   <Name>username</Name>
   <Value>dbname</Value>
</ConnectionProperty>
```

The value for password is the LDAP user attribute that contains the database password.

In the following example, iServer uses the value in the LDAP user's dbpassword attribute as the database password:

```
<ConnectionProperty>
   <Name>password</Name>
   <Value>dbpassword</Value>
</ConnectionProperty>
```

Setting ldapconfig_<volume>.xml parameters

The RSSE application uses a mapping file, ldapconfig_<volume>.xml, to map Encyclopedia volume management information to LDAP objects and object attributes.

In the Actuate ldapconfig_<volume>.xml file, a parameter is an XML element. Specify the value for a parameter as shown in the following example:

```
<parameter-name>value 1, value 2</parameter-name>
```

where

- The parameter name is one of the valid parameter names specified in ldapconfig_<volume>.xml.
- A comma separates multiple parameter values.

ldapconfig_<volume>.xml can contain comments. Enclose comments in <- - and - -> tags, as shown in the following example:

```
<
--This is the port number on which the LDAP server is listening.--
   >
```

Table 11-2 contains example values for parameters that appear in ldapconfig_<volume>.xml.

Table 11-2 ldapconfig_<volume>.xml parameters

Parameter	Description and example

AdminRole	Actuate role attribute value that indicates that an LDAP user object can perform Encyclopedia volume management.
	`<AdminRole>`
	`actuateAdmin`
	`</AdminRole>`

AllRole	LDAP role object name that maps to the All role in the Encyclopedia volume. Use the All role to grant privileges to all Encyclopedia volume users.
	`<AllRole>`
	`actuateAll`
	`</AllRole>`

GroupBase DN	Base LDAP distinguished name used to locate the LDAP Actuate notification group object in queries of notification group names.
	`<GroupBaseDN>`
	`ou=Groups, dc=actuate, dc=com`
	`</GroupBaseDN>`

(continues)

Group Object	LDAP object class that the RSSE application uses to find Actuate notification group names.
	`<GroupObject>`
	`groupofuniquenames`
	`</GroupObject>`

GroupTo Notify	Name of the LDAP notification group that receives notification about all iServer requests in the manner of the administrator user when the Encyclopedia volume uses default, internal security. The GroupBaseDN parameter defines the base DN of this group name.
	`<GroupToNotify>`
	`specialGroup`
	`</GroupToNotify>`
	When combined with the GroupBaseDN value, this parameter specifies the LDAP Actuate notification group object. iServer uses that object for LDAP notification. For example:
	`"cn=AdminGroup, ou=Actuate Groups, o=actuate.com"`

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example

Operator Role	LDAP role object name that maps to the Encyclopedia volume Operator role. A user must have this role name to perform Encyclopedia volume Operator functions, such as online Encyclopedia volume backups.
	`<OperatorRole>`
	`actuateOperator`
	`</OperatorRole>`

Port	Internet port on which the LDAP server listens.
	The default value is 389.
	`<Port>`
	`389`
	`</Port>`

Query Account	LDAP account that the RSSE application uses for query operations to the LDAP server.
	`<QueryAccount>`
	`uid=actuate, ou=Administrators, ou=TopologyManagement, o=NetscapeRoot`
	`</QueryAccount>`
	The RSSE application uses this account to validate users, roles, ACLs, and other Encyclopedia volume user information. For example:
	`"uid=admin, ou=Administrators, ou=TopologyManagement, o=NetscapeRoot"`

Query Password	Password for the LDAP account specified by the QueryAccount parameter.
	`<QueryPassword>`
	`Actu8`
	`</QueryPassword>`

RoleBaseDN	Base LDAP distinguished name that the RSSE application uses to locate the LDAP role object in queries of roles.
	`<RoleBaseDN>`
	`ou=AcRoles, dc=actuate, dc=com`
	`</RoleBaseDN>`

RoleObject	LDAP object class that the RSSE application uses to find Actuate role names.
	`<RoleObject>`
	`groupofuniquenames`
	`</RoleObject>`

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example

Server	Name of the LDAP server that the RSSE application and iServer use. Use the fully qualified name, including the domain name. You can use the server's IP address.
	The default value is the name of the machine.
	`<Server>`
	`helium.actuate.com`
	`</Server>`

UserBase DN	LDAP distinguished name that the RSSE application uses to locate the LDAP user object. When you add a user's name as a prefix to a base-distinguished name, the resulting name uniquely identifies the user in the external data source. Most base-distinguished names consist of the organizational unit or a series of organizational units and an organization.
	`<UserBaseDN>`
	`ou=People, dc=actuate, dc=com`
	`</UserBaseDN>`

Channel Subscription ListAttr	LDAP attribute that specifies the channels to which an Actuate user subscribes.
	In the LDAP directory server, the attribute has multiple values with a single channel name for each value.
	`<ChannelSubscriptionListAttr>`
	`actuateChannelList`
	`</ChannelSubscriptionListAttr>`

(continues)

Channel Subscription ListDefault	Value to use for ChannelSubscriptionListAttr when LDAP does not contain a value for that attribute.
	`<ChannelSubscriptionListDefault>`
	`portfolio update, sales forecasts`
	`</ChannelSubscriptionListDefault>`
	The value is a comma-separated lists of channel names. For example:
	`"portfolio update, sales forecasts"`

Privilege Template Attr	LDAP attribute that specifies which privilege template to use for files and folders that an Encyclopedia volume user creates.
	`<PrivilegeTemplateAttr>`
	`actuateDefaultPriv`
	`</PrivilegeTemplateAttr>`

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example

Privilege Template Default	Value to use for PrivilegeTemplateAttr when LDAP does not contain a value for that attribute.
	The value is a comma-separated list of user or role privileges. This example gives read and visible privileges to a role called viewing only and gives read, write, execute, and delete privileges to a user named jbob.
	`<PrivilegeTemplateDefault>`
	`viewing only~rv, jbob=rwed`
	`</PrivilegeTemplateDefault>`
	A user permission is a user login name followed by "=" and a zero (0) or more permission characters. A role permission is a role name followed by tilde (~) followed by a zero or more permission characters. The following table is a list of the privilege characters and their meanings:
	r = read
	w = write
	e = execute
	d = delete
	v = visible
	s = secure read
	g = grant
	To specify a privilege template that lists multiple users or roles in the LDAP directory server, the attribute must be multivalued with a single user or role for each value.

Attach Report InEmailAttr	LDAP attribute that specifies an Actuate user's preferred form of e-mail notification.
	`<AttachReportInEmailAttr>`
	`actuateEmailForm`
	`</AttachReportInEmailAttr>`
	The e-mail can contain either a copy of the document or a link to the document.

Attach Report InEmail Default	Value to use for AttachReportInEmailAttr when LDAP does not contain a value for that attribute. The value is either included or linked. If the value is included, the user receives the document as an attachment to the notice, if possible. If the value is linked, the user receives a link to the document. The default value in ldapconfig_<volume>.xml is linked.
	`<AttachReportInEmailDefault>`
	`linked`
	`</AttachReportInEmailDefault>`

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example

Email Address Attr	Name of the LDAP user attribute that specifies an Encyclopedia volume user's e-mail address that iServer uses to send e-mail. For some object classes, such as inetorgperson, an e-mail attribute exists in the standard LDAP schema.
	`<EmailAddressAttr>`
	`  mail`
	`</EmailAddressAttr>`

SendEmail Attr	LDAP user attribute that specifies when to send an e-mail notification message to notify an Actuate user of the completion of a job.
	`<SendEmailAttr>`
	`  actuateEmailWhen`
	`</SendEmailAttr>`

SendEmail Default	Value to use for SendEmailAttr when the LDAP directory server does not contain a value for that attribute.
	`<SendEmailDefault>`
	`  never`
	`</SendEmailDefault>`
	Use one of the following values: never, always, failures, or successes.
	never—Do not modify.
	always—Notify of failures and successes.
	failures—Notify of failures only.
	successes—Notify of successes only.
	The default value in ldapconfig_<volume>.xml is never.

(continues)

Failure Notice Expiration Attr	LDAP attribute that specifies how long iServer keeps a user's notices about failed jobs in the completed notice folder of the Encyclopedia volume. The value is a number of minutes. A value of 0 (zero) means that iServer does not keep notices about failed jobs. A value of -1 means that iServer keeps the notices indefinitely.
Failure Notice Expiration Default	Value to use for FailureNoticeExpirationAttr when LDAP does not contain a value for that attribute. The value is a number of minutes. The default value in ldapconfig_<volume>.xml is 0.
SendNotice Attr	LDAP user attribute that specifies when to notify a user about the completion of a job by placing a notice in the completed notice folder of the Encyclopedia volume.

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example
SendNotice Default	Value to use for SendNoticeAttr when LDAP does not contain a value for that attribute. Use one of the following values: never, always, successes, or failures. never—Do not modify. always—Notify of failures and successes. failures—Notify of failures only. successes—Notify of successes only. The default value in ldapconfig_<volume>.xml is always.

Home Folder Attr	LDAP attribute that specifies a user's home folder in the Encyclopedia volume.
	There is no default value.
	`<HomeFolderAttr>`
	`  actuateHomeFolder`
	`</HomeFolderAttr>`

MaxJob PriorityAttr	LDAP attribute that specifies a user's maximum request priority.
	The value is the maximum request priority that the user can set for a document print or generation request in the Encyclopedia volume.
	In LDAP, the value must be an integer between 0 and 1000.
	`<MaxJobPriorityAttr>`
	`  actuateMaxPriority`
	`</MaxJobPriorityAttr>`

MaxJob Priority Default	Value to use for MaxJobPriorityAttr when LDAP does not contain a value for that attribute.
	The value must be an integer between 0 and 1000.
	The default value in ldapconfig_<volume>.xml is 500.
	`<MaxJobPriorityDefault>`
	`  500`
	`</MaxJobPriorityDefault>`

UserObject	Name of the LDAP object class that the RSSE application uses to find Actuate user names.
	An example of an LDAP object class is inetorgperson.
	`<UserObject>`
	`  inetorgperson`
	`</UserObject>`

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example

Success Notice Expiration Attr	LDAP attribute that specifies how long to keep a user's success completion notices in the completed notice folder of the Encyclopedia volume.
	The value is a number of minutes. A value of 0 (zero) discards notices about successful jobs. A value of -1 keeps success notices indefinitely.
	`<SuccessNoticeExpirationAttr>`
	`  actuateSuccessNoticeExpiration`
	`</SuccessNoticeExpirationAttr>`

Success Notice Expiration Default	Value to use for SuccessNoticeExpirationAttr when LDAP does not contain a value for that attribute.
	The value is a number of minutes. The default value in ldapconfig_<volume>.xml is 0.
	`<SuccessNoticeExpirationDefault>`
	`  0`
	`</SuccessNoticeExpirationDefault>`

View Preference Attr	LDAP attribute that specifies the user's default viewing preference.
	Use one of the following values: default or dhtml.
	`<ViewPreferenceAttr>`
	`  actuateViewingPref`
	`</ViewPreferenceAttr>`

(continues)

View Preference Default	Value to use for ViewPreferenceAttr when LDAP does not contain a value for that attribute.
	Specify the default viewing mode using one of the following values: default or dhtml.
	The default value in ldapconfig_<volume>.xml is default.
	`<ViewPreferenceDefault>`
	`  default`
	`</ViewPreferenceDefault>`

Connection PropertyList	Values to use for information object pass-through security. When using pass-through security, iServer requires a database user name and password. The ConnectionPropertyList element contains two ConnectionProperty elements. Each ConnectionProperty element contains a Name and Value element. The values for the ConnectionProperty Name elements are username and password. The value for username is the LDAP user attribute that contains the database user name.

Table 11-2 ldapconfig_<volume>.xml parameters (continued)

Parameter	Description and example

	ConnectionProperty Value for password is the LDAP user attribute that contains the database password.
	`<ConnectionPropertyList>`
	`  <ConnectionProperty>`
	`  <Name>username</Name>`
	`  <Value>dbname</Value>`
	`</ConnectionProperty>`
	`<ConnectionProperty>`
	`  <Name>password</Name>`
	`  <Value>dbpassword</Value>`
	`  </ConnectionProperty>`
	`</ConnectionPropertyList>`

Understanding an LDAP directory structure

Figure 11-13 shows a simple LDAP directory structure that stores information about the following iServer objects:

- Encyclopedia volume users
- Security roles
- Notification groups

In the LDAP directory, Actuate is the LDAP domain that the RSSE application uses. The domain Actuate contains LDAP objects used by the RSSE application.

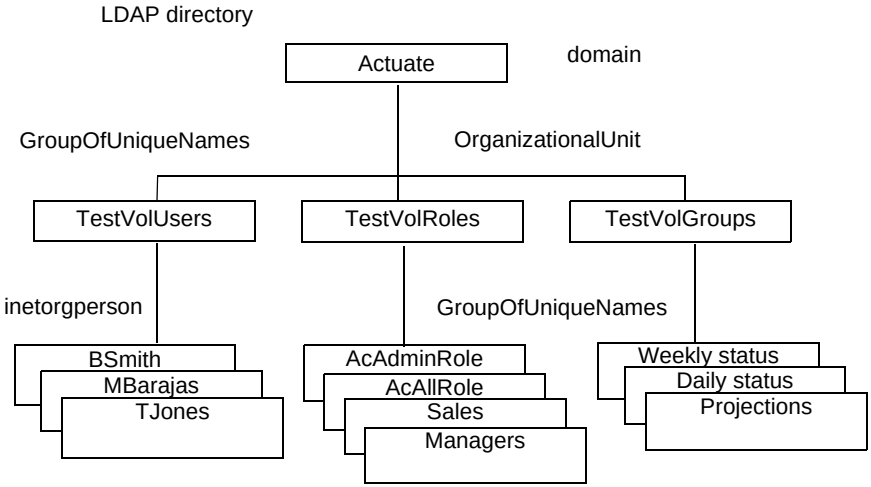

Figure 11-13 LDAP directory structure

TestVolUsers is an instance of groupofuniquenames. TestVolUsers contains a list of the Encyclopedia volume users. The users are instances of the inetorgperson LDAP object.

TestVolRoles and TestVolGroups are instances of the OrganizationalUnit LDAP object. Within TestVolRoles are instances of the GroupOfUniqueNames LDAP object with names AcAdminRole, AcAllRole, Sales, and Managers. Each GroupOfUniqueNames object contains references to Actuate users that are instances of the inetorgperson LDAP object.

Within TestVolGroups are instances of the GroupOfUniqueNames LDAP object with names Weekly status, Daily status, and Managers. Each GroupOfUniqueNames object contains references to Actuate users that are instances of the inetorgperson LDAP object.

Mapping LDAP objects to users

Use the following parameters in the LDAP RSSE application to map the LDAP objects to Encyclopedia volume users:

- UserObject parameter maps the LDAP object to the Encyclopedia volume user.

  ```
  <UserObject>inetorgperson</UserObject>
  ```

- UserBaseDN parameter maps the instance of the LDAP object to the Encyclopedia volume user.

  ```
  <UserBaseDN>cn=TestVolUsers, dc=actuate, dc=com</UserBaseDN>
  ```

In the configuration file, use the following properties to map the LDAP objects to Encyclopedia volume security roles:

- RoleObject parameter maps the LDAP object to the Encyclopedia volume security role.

  ```
  <RoleObject>groupofuniquenames</RoleObject>
  ```

- RoleBaseDN parameter maps the instance of the LDAP object to the Encyclopedia volume security role.

  ```
  <RoleBaseDN>ou=TestVolRoles, dc=actuate, dc=com</RoleBaseDN>
  ```

In the configuration file, use the following properties to map the LDAP objects to Encyclopedia volume notification groups:

- GroupObject parameter maps the LDAP object to the Encyclopedia volume notification group.

  ```
  <GroupObject>groupofuniquenames</GroupObject>
  ```

- GroupBaseDN parameter maps the instance of the LDAP object to the Encyclopedia volume notification group.

```
<GroupBaseDN>ou=TestVolGroups, dc=actuate, dc=com
  </GroupBaseDN>
```

In the configuration file, use the following properties to map the LDAP objects to the special Encyclopedia volume roles:

- AdminRole parameter maps the instance of the LDAP object to the Encyclopedia volume administrator security role.

  ```
  <AdminRole>AcAdminRole</AdminRole>
  ```

- AllRole parameter maps the instance of the LDAP object to the Encyclopedia volume administrator security role.

  ```
  <AllRole>AcAllRole</AllRole>
  ```

Converting an Encyclopedia volume to use an RSSE application

When you configure iServer to use the default Encyclopedia volume security, the Encyclopedia volume stores all security information. iServer uses an identifier ID and does not use the name of the user, role, or group when it assigns privileges and sets other administrative options. To use the RSSE application and an external security source, replace the internal IDs with the user, role, or group name. When the administrator enables open security as a web service and restarts the volume, iServer replaces references to Actuate user, role, and group IDs with their corresponding names in the following Encyclopedia volume management information:

- Privilege rules, or access control lists (ACLs) for files and folders in the volume
- Privilege rules in scheduled jobs
- Privilege rules for a volume's channels
- Job completion notification settings
- Scheduled jobs
- Completed jobs

The RSSE application matches the users, roles, and groups in the Encyclopedia volume to users, roles, and groups in the external security source. When the administrator disables the open security as a web service and restarts the volume, iServer modifies these references. For each user name reference in the Encyclopedia volume, iServer looks up the corresponding Encyclopedia volume ID and changes the reference to an ID. If there is no corresponding ID, iServer removes the reference.

Converting internal IDs to external names

To convert existing Encyclopedia volume management information to a form that the RSSE application can use, complete the following tasks in this order:

1. Open Properties for the selected Volume.
2. In Open Security, select “Enable as web service”. Choose OK.
3. Take the volume online.

When the volume is taken online, iServer converts the Encyclopedia volume references from internal users, roles, and groups to the corresponding external names.

Converting information from external to internal

To convert an Encyclopedia volume from using external information to using internal information, complete the following tasks in this order:

1. Open Properties for the selected Volume.
2. In Open Security, select “Do not enable”. Choose OK.
3. Take the selected volume offline.
4. Take the selected volume online.

When the volume is taken online, iServer converts the Encyclopedia volume references from external users, roles, and groups to the corresponding internal IDs.

Caching external security information

When using an external security application, iServer caches external security source information, including:

- External user properties
- Roles of a user under external user registration
- Groups and group memberships

Control the maximum time that the cache holds the information by setting Open Security parameters. Use the RSSE Cache Timeout parameter to control how long the cache stores security information. The default value is 60 minutes. The RSSE Cache Timeout setting can be set to -1 to specify that the RSSE cache never expires.

Working with RSSE page-level security

The following topics describe how to use the example RSSE page-level security application:

- About the RSSE page-level security example
- Installing the RSSE web service application
- Creating an access control list

About the RSSE page-level security example

Using the RSSE framework, a developer can create an RSSE service that manages page-level security by retrieving a user's access control list (ACL) externally. By default, when a secure design asks for the ACL of a user, the Encyclopedia volume returns a list that includes the user ID and the roles to which the user belongs. Frequently, the information in iServer security does not match the information in a database used by a secure design. An RSSE page security application can translate an iServer ACL to a design-specific ACL.

iServer Integration Technology contains an example of how external page-level security works using RSSE. This example is located under the Java Report Server Security Extension directory in the subdirectory, Page_Security_Example.

To use page-level security, license the BIRT Page Level Security Option if you are working with BIRT designs and documents, or the e.Report Page Level Security Option if you are working with Actuate Basic designs and documents. The example RSSE page-level security application uses a text file, users.acls, that maps Encyclopedia volume user names to other external security IDs. Use the sample application as a basis for a custom RSSE application.

Page-level security protects a report from unauthorized access by comparing the user ID and user membership in Encyclopedia volume roles to an access control list (ACL) for the report document. If the user's name appears in the ACL, the user can view the protected pages.

The RSSE application uses a file, users.acls, to translate an Encyclopedia volume user ID or role to one or more security IDs, which iServer uses to check against the Actuate report page ACL when a report uses page-level security.

For example, if a report shows information about the sales reps in the following city offices:

- NYC
- Boston
- Philadelphia

and the file, user.acls, contains the following access control list specifications:

```
user1=NYC
user2=Boston
user3=Philadelphia
```

then User1 has access to the pages with information about NYC office, user2 to the Boston office, and user3 to the Philadelphia office.

In this example, when user1 tries to read a report with page-level security enabled, the RSSE application returns a list of security IDs that contain user1 and user1's roles. iServer checks the user ID, roles, and RSSE list against the Actuate report page ACL. iServer lets user1 view any report page where a security ID that the RSSE application returns matches a report page security ID.

Creating an access control list

The example file, user.acls, stores a user's access control list (ACL) using the following format:

```
Username=acl1, acl2, ..
```

The username field matches the name of user in the Encyclopedia volume. An equal ('=') sign separates the user name from the ACL list. The ACL list can contain 0, 1 or more ACL specifications as shown in the following code example:

```
user1=acl1, acl2, acl3, acl4
user2=acl5, acl6,\
   acl7, acl8
user3=acl9
user4=acl10
```

If there is more than one ACL in the list, separate each ACL using a comma. The scanner reading the file, users.acls, eliminates any white space or backslash. All the username specifications in the example are legal. A list can contain users that do not appear in the Encyclopedia. The information for these users is ignored.

The default maximum length of an ACL is 64 kilobytes (KB). If you use an ACL longer than 64 KB, specify a longer maximum length using the RSSE Max ACL Buffer Size parameter on Volumes—Properties—Open Security in Configuration Console.

Chapter

12

Archiving files

This chapter contains the following topics:

- Understanding online archiving
- Using the online archive driver

Understanding online archiving

iServer ships with a configurable, Java-based Encyclopedia volume archive driver that you can use to archive files from an Encyclopedia volume. Developers can create custom online archive drivers using the Actuate Information Delivery API (IDAPI). The source code and build files for the online archive driver reside in the Online Archive Driver directory after installing IDAPI.

Using the online archive driver or your customized version requires purchasing the Online Archive Option.

About the online archive driver

The iServer online archive driver copies expired Encyclopedia volume files to a second Encyclopedia volume that serves as a file archive.

The online archive driver supports the following features:

- Preserving file attributes, such as description, version name, and security information
- Preserving file dependencies
- Deleting empty folders from the source Encyclopedia volume

Configure iServer to use the online archive driver from Configuration Console. Configure and perform archiving from Management Console.

About the online archive driver files

iServer installs the following online archive driver files:

- A script runs the driver and sets environment variables. On UNIX or Linux, the script is aconlinearchive.sh. On Windows, the script is aconlinearchive.bat. The script resides in AC_SERVER_HOME/bin.
- The Java JAR files aconlinearchive.jar and aconlinearchiveDEP.jar in AC_SERVER_HOME/drivers.
- An XML configuration file, onlinearchive.cfg, is a reference implementation. The default location for the configuration file is the AC_SERVER_HOME/etc directory.

Retaining file attributes during archiving

When configuring the online archive driver, specify whether the driver retains the following general file and security information:

- Time stamp

- Owner
- Permissions
- Dependencies

If the online archive driver does not retain the permissions information for an archived file, the owner of the file has full access. If the driver retains security information, including owner and privilege information about an archived file, the driver performs the following functions:

- The access control list (ACL) of the file refers to a user or role, and the driver attempts to match the reference to a name in the archive. When using an existing Encyclopedia volume user or role, the driver does not update the properties of the user or role with information from the original volume.
- If the archive uses RSSE external registration for user and role information, the online archive driver uses the RSSE information to update the user and role privileges to access the file.
  - If the original owner of the file is not defined in the archive, the user named in the configuration file of the driver becomes the owner of the file when it is added to the archive.
  - If a user or role is in the original ACL, but not in the archive, they do not get privileges to access the file when it is added to the archive.
- If the user or role is not present in the archive, the driver creates them, assigning the properties of the original user or role to them, except the driver does not enable login for a user or role it creates.

Preserving file dependencies during archiving

When the online archive driver archives a file that has a dependency on another file, the driver can archive both files and preserve the dependency information. When the driver retains file dependency, and an archived file depends on another file, the driver copies both files to the archive and creates the same relative file structure. The driver does not delete the file on which the archived file depends from the original Encyclopedia volume, unless the driver archives both files.

If more than one archived file depends on the same file, the driver retains the original file dependency and typically retains only one copy of the file on which the archived files depend. The driver archives two copies of the file on which the other files depend, one for each file that is archived in each session.

Using the online archive driver

To use the online archive driver, first create an Encyclopedia volume for storing archived files. Next, specify the configuration file that contains the configuration

parameters for the driver. Use the default configuration file, shown in Listing 12-1, or modify the file. Finally, specify the startup script file that sets the environment variables and launches the online archive driver.

Specifying the online archive configuration file

The online archive driver uses an XML configuration file. The default name of the configuration file is onlinearchive.cfg. The default location for the configuration file is the AC_SERVER_ HOME\ etc directory.

In UNIX, you can specify the configuration file using the -c command-line option. For example, specify the following command for the Encyclopedia volume parameter, Use archive service, on System Volumes—Properties—General:

```
aconlinearchive.sh -c /local/actuate11/archiveconfig.xml
```

In a production system, you can configure an Encyclopedia on a different volume or on a separate node in a cluster to function as the repository for the archive. You copy the reference implementation of onlinearchive.cfg in AC_SERVER_ HOME /etc, and rename it using the following format:

```
onlinearchive_<volume>.cfg
```

where <volume> is the name of the Encyclopedia volume that runs the archive driver.

Modifying the online archive configuration file

Listing 12-1 shows the default online archive configuration file. You can modify the parameters described in Table 12-1.

Listing 12-1 Default online archive configuration file

```
<?xml version="1.0" encoding="UTF-8"?>
<archiveconfig>
   <
--TargetServer, TargetSOAPPort: [Required]              -->
   <
--    Name or IP of server and port for connecting to   -->
   <
--   the SOAP dispatcher service of the target volume  -->
   <TargetServer>localhost</TargetServer>
   <TargetSOAPPort>8000</TargetSOAPPort>

   <
--ArchiveVolume: [Required]                            -->
   <
--    Name of target volume to copy archived files to   -->
   <ArchiveVolume>DefaultVolume</ArchiveVolume>
```

```
   <
--AdminUser, AdminPassword: [Required]                    -->
   <
--    Name and password of a user in the target volume   -->
   <
--    that belongs to the Administrator role             -->
   <AdminUser>administrator</AdminUser>
   <AdminPassword></AdminPassword>

   <
--RetainTimestamp: [Optional, default: false]             -->
   <
--    Whether timestamp of archived file is preserved    -->
   <RetainTimestamp>false</RetainTimestamp>

   <
--RetainOwner: [Optional, default: false]                 -->
   <
--    Whether Owner of archived file is preserved        -->
   <RetainOwner>false</RetainOwner>

   <
--RetainPermission: [Optional, default: false]            -->
   <
--    Whether Permission (ACL) of archived file is       -->
   <
--    preserved                                          -->
   <RetainPermission>false</RetainPermission>

   <
--CopyDependOnFile: [Optional, default: true]             -->
   <
--    Whether files depended on by archived file are     -->
   <
--    copied                                             -->
   <CopyDependOnFile>true</CopyDependOnFile>

   <
--CreateUserRole: [Optional, default: true]               -->
   <
--    Whether to create missing user or roles in target  -->
   <
--    volume in order to retain Owner or Permissions     -->
   <CreateUserRole>true</CreateUserRole>
```

```
   <
--ArchiveRoot: [Optional, default: /]                    -->
   <
--    Root encyc folder for all archived files          -->
   <ArchiveRoot>/</ArchiveRoot>

   <
--CreateArchiveSubFolder: [Optional, default: true]     -->
   <
--    Whether to create a timestamp dependent subfolder -->
   <
--    under ArchiveRoot for each archive session        -->
   <CreateArchiveSubFolder>true</CreateArchiveSubFolder>

   <
-- LogLevel: [Optional, default: Summary]               -->
   <
--    Level of detail in log file. Valid values are: -->
   <
--    Summary, Detail and Trace  -->
   <LogLevel>Summary</LogLevel>

</archiveconfig>
```

Table 12-1 describes the parameters in the online archive configuration file for the Encyclopedia volume that stores the archive.

Table 12-1 Online archive configuration file parameters

Parameter	Description
TargetServer	Machine name or IP address to use to connect to the Encyclopedia volume that holds the archived files. Required.
TargetSOAPPort	Port number that the iServer SOAP Dispatch service uses to connect to the Encyclopedia volume. Required.
ArchiveVolume	Encyclopedia volume name. The default value is DefaultVolume. Required.
AdminUser	Encyclopedia volume user name. The user must belong to the Administrator role.
AdminPassword	Password of the user specified by the AdminUser parameter. Required.
RetainTimestamp	Flag for archiving the file's time stamp. The default value is false.

Table 12-1 Online archive configuration file parameters

Parameter	Description
RetainOwner	Flag for archiving the name of the file's owner. The default value is false.
RetainPermission	Flag for archiving the file's permissions. The default value is false.
CopyDependOnFile	Flag for archiving the file's dependency list. The default value is true.
CreateUserRole	Flag for creating missing user or roles in the archive Encyclopedia volume to retain the file's owner or permission information. The default value is true.
ArchiveRoot	Encyclopedia volume archive session root folder. The default value is /, the Encyclopedia volume root folder.
CreateArchiveSubFolder	Flag for creating a time stamp dependent subfolder under ArchiveRoot for each archive session. The default value is true.
LogLevel	The level of detail for information in the log. Valid values are Summary, Detail, and Trace. The default value is Summary.

The online archive driver applies changes to the configuration file when iServer runs the driver. Changes do not affect any archiving already in process.

Creating and specifying the startup script

Create a script to start your online archive driver, or use the implementation script and driver as shown in Listing 12-2.

Listing 12-2 Sample online archive driver startup script

```
@echo off
REM Actuate Online Archive Driver Startup Script
set ARCHIVE_DRIVER_JRE=%AC_JRE_HOME%
.
.
.
set DRIVER_JAR_PATH=%AC_SERVER_HOME%\drivers
set JAVA_EXE=%ARCHIVE_DRIVER_JRE%\bin\java.exe
if exist "%JAVA_EXE%" goto JavaOK
```

```
echo *** Java home not set correctly; trying Java.Exe in PATH ***
set JAVA_EXE=java.exe
:JavaOK
set DRIVER_JAR=%DRIVER_JAR_PATH%\aconlinearchive.jar
set DRIVER_LIB_JAR=%DRIVER_JAR_PATH%\aconlinearchiveDEP.jar
set DRIVER_CP=%DRIVER_JAR%;%DRIVER_LIB_JAR%
start "Online Archive Service" "%JAVA_EXE%" -cp "%DRIVER_CP%"
   com.actuate.onlinearchivedriver.Main %1 %2 %3 %4 %5 %6 %7 %8
```

To specify the script file, in the Advanced view of Configuration Console, choose Volumes. Choose Properties from the drop-down menu of a volume. For Volume archive service provider, in Use archive service on Properties—General, type the path and name of the script, as shown in Figure 12-1.

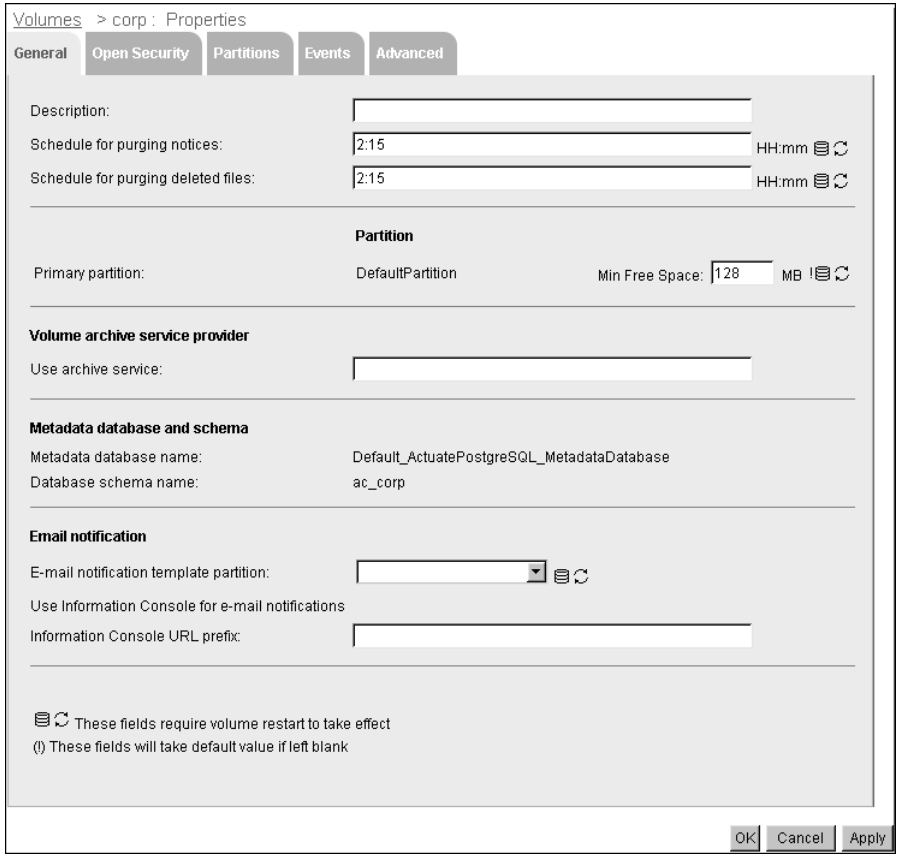

Figure 12-1 Volumes—Properties—General

The default Volume archive service uses the configuration batch file aconlinearchive.bat, shown in Figure 12-1, that is located in the AC_SERVER_HOME/bin directory.

Managing file purging

In Volumes—Properties—Advanced—Archiving and Purging, shown in Figure 12-2, you can specify expiration times for Encyclopedia volume items.

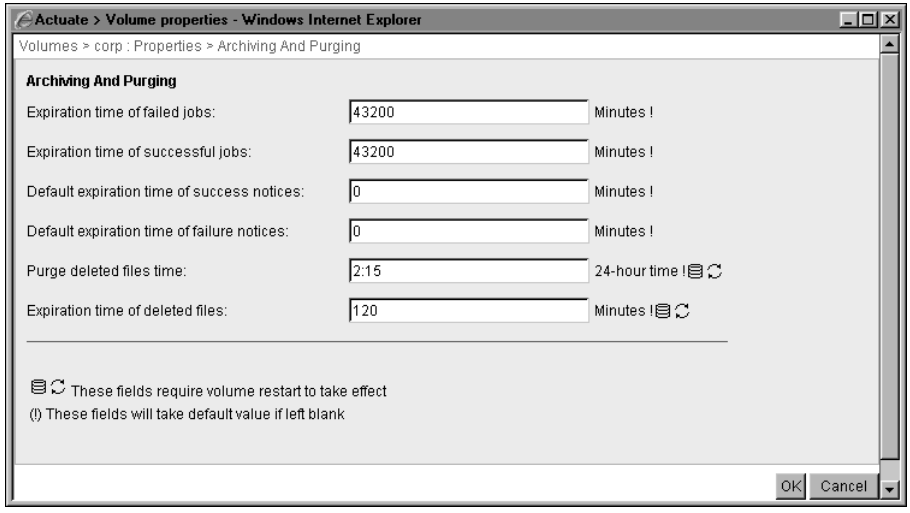

Figure 12-2 Specifying Encyclopedia volume item expiration times

Table 12-2 describes the parameters that appear on System Volumes—Properties—Advanced—Archiving and Purging. Purging notices reduces the size of the Encyclopedia volume and contributes to better performance.

Table 12-2 Archiving and purging parameters

Parameter	Description
Expiration time of failed jobs	Length of time before iServer can delete a job completion notice for a failed job from Jobs-Completed. iServer purges the notice during the scheduled completion notice purge time if no completion notice for the failed job exists on a volume channel, and the age of the notice exceeds the value of this parameter. The lower the value of this parameter, the better iServer performs. If an administrator specifies 0 for this parameter, iServer uses the default value for the volume.

(continues)

Table 12-2 Archiving and purging parameters (continued)

Parameter	Description
Expiration time of successful jobs	Length of time before iServer can delete a job completion notice for a successful job from Jobs-Completed. iServer purges the notice during the scheduled completion notice purge time if no completion notice for the successful job exists on a volume channel, and the age of the notice exceeds the value of this parameter. The lower the value of this parameter, the better iServer performs. If an administrator specifies 0 for this parameter, iServer uses the default value for the volume.
Default expiration time of success notices	The default value for the volume specifying the length of time before iServer can delete a job completion notice for a successful job from a user's personal channel. This is the same property as Purge success notices after n days n hours, in Management Console, Volume—Properties—Archiving and Purging. Setting this property in one console sets the property in the other console. iServer does not delete the notice if the value for this parameter is 0, the default value.
Default expiration time of failure notices	The default value for the volume specifying the length of time before iServer can delete a job completion notice for a failed job from a user's personal channel. This is the same property as Purge failure notices after n days n hours, in Management Console, Volume—Properties—Archiving and Purging. Setting this property in one console sets the property in the other console. iServer does not delete the notice if the value for this parameter is 0, the default value.

Locating the archived files

The online archive driver creates a folder in the archive Encyclopedia volume and places files from an online archive session in this folder. In the driver configuration file, you specify the name of a root folder that contains the directories for all the online archive sessions. The directory structure within the folder matches that of the original Encyclopedia volume. The subfolder name containing the files from the online archive session consists of the start date and time of the online archive session in the following format:

```
YYYY_mm_dd.hh_mm_ss
```

For example, if you specify the online archive content folder as /archive2009, for an autoarchive session starting on 8:25 pm June 2, 2009, the driver copies the file /documents/sales/commission.rptdocument to the archive Encyclopedia volume in the following location:

```
/archive2009/2009_06_02.20_25_14/documents/sales
   /commission.rptdocument
```

Chapter

13

Printing documents

This chapter contains the following topics:

- Understanding printing
- Sending a document to the printer
- Managing the printing environment
- Configuring fonts

Understanding printing

This chapter describes the requirements and operations necessary to set up document printing in the Windows, UNIX, and Linux environments.

Printing on Windows

The installation process on Windows sets up printers already configured on iServer as system printers. To connect additional system printers after installation, first connect them to client computers using Printers and Faxes in the Control Panel. Next, in Configuration Console—Advanced view, you add printers to the server using Printers—Add Printer, shown in Figure 13-1.

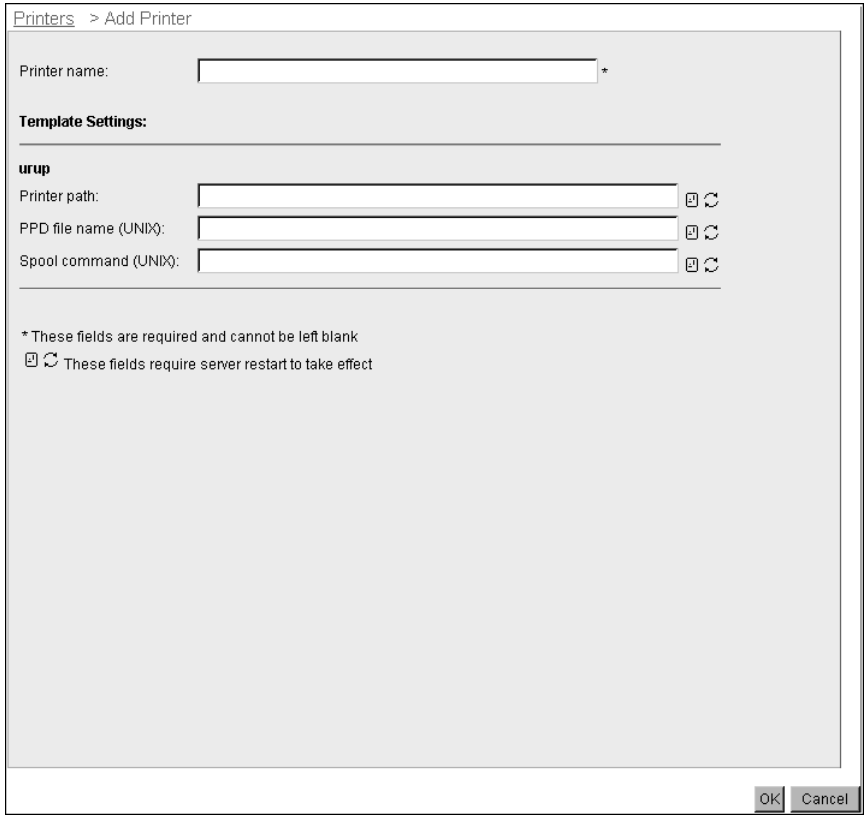

Figure 13-1 Adding a printer

To add an iServer printer, connect the printer to client computers using Printers and Faxes in the Control Panel. Next, add the printer as an iServer printer. Lastly, restart iServer.

How to add a printer as an iServer printer

1 From the Advanced view of Configuration Console, choose Printers. Printers lists the printers available to iServer, as shown in Figure 13-2. In Printers, choose Add Printer.

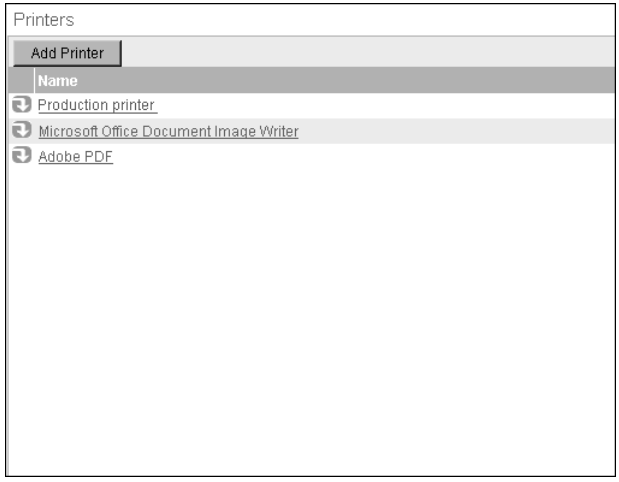

Figure 13-2 List of printers

In Printers—Add Printer, you can change the following server settings for each machine:

- PPD File Name
  PPD file name for UNIX and Linux operating systems. Do not specify a path.
- Printer name
  Name of the printer.
- Printer path
  Path to the printer from the machine.
- Spool command
  Spool command for UNIX or Linux. Actuate recommends using the copy before printing option. For example, on an AIX system, use the following command:

  ```
  lp -c -d
  ```

  On Solaris and Linux, use the following command:

  ```
  lpr -P
  ```

Choose OK.

2 Restart iServer.

In an iServer cluster, each machine maintains printer property information. When adding a printer to a cluster, you must specify the printer parameter values for each node in the cluster.

Printing on UNIX and Linux

Actuate sends output to any Windows printer and to PostScript Level 1 or 2 printers. If the PostScript font is not installed on the system, the PostScript file, generated at print time, contains only the references to the missing font name.

You must install required PostScript fonts on either the printer or iServer; otherwise, the printer substitutes a font or printing fails. If you install the PostScript font on iServer, font information embeds in the PostScript file, and the document prints with the correct font.

iServer uses Adobe Type-1 font technology and a font configuration file for printing. The configuration file is in the following directory:

```
AC_SERVER_HOME/operation/print/fonts/fonts.supported
```

Throughout this documentation, AC_SERVER_HOME refers to the folder that the installer chose as the location for binary files during the iServer installation. By default, this location is AC_SERVER_HOME/AcServer in a Linux or UNIX environment. To print documents, check that XVFBDISPLAY variables, start_srvr.sh and display_value, are set in pmd11.sh.

Adding a printer on UNIX and Linux

For iServer to recognize a printer in Solaris, HP, and IBM environments, the printer name must appear in the following locations:

- Solaris and Linux—a subdirectory of the /etc/lp/printers directory. On Solaris 9 and later, iServer also recognizes printers you can list using the following command:

  ```
  lpget list
  ```

- HP—the /etc/lp/member directory.
- IBM—the /etc/qconfig file.

How to add a printer on UNIX and Linux

You add a printer to the server in Configuration Console—Advanced view, using Printers—Add Printer, shown in Figure 13-3. Specify the following information:

- The printer path.
- Name of the PostScript printer definition (PPD) file. Do not include the path.
- The name of the spool command.

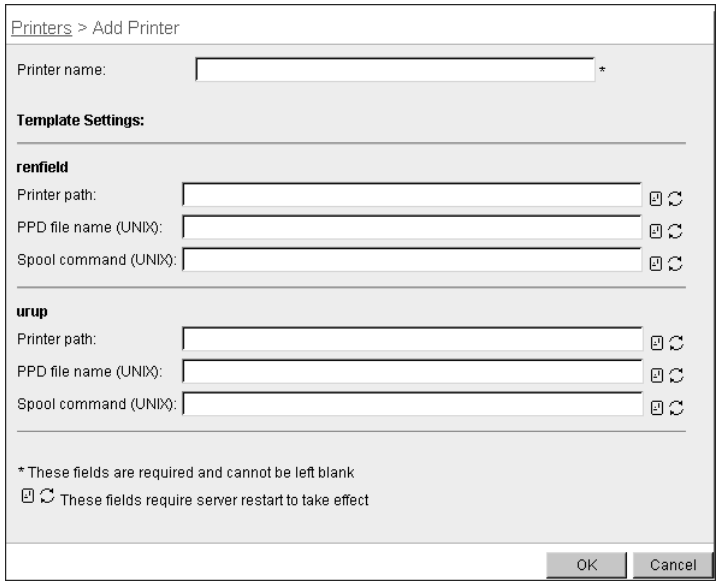

Figure 13-3 Adding a printer to iServer in UNIX

Restart iServer after adding printers.

About the PostScript font utility

The Actuate UNIX utility, fontutils, installs and uninstalls PostScript fonts to the machine. The utility is in the AC_SERVER_HOME/bin directory. Before using the fontutils utility, append the following path to the library path environment variable:

```
AC_SERVER_HOME/lib
```

The name of the environment variable depends on the platform, as follows:

- On SunOS and Linux, the environment variable is LD_LIBRARY_PATH.
- On AIX, the environment variable is LIBPATH.
- On HP-UX, the environment variable is SHLIB_PATH.

About Xprinter environment variables

The XPHOME and XPPATH environment variables for Xprinter on UNIX and Linux specify the location of the iServer Xprinter home and the path to Xprinter files, respectively.

Installing a PostScript font

Copy the font's .afm and .pfa files to the AC_SERVER_HOME/bin directory. Convert the .afm file to UNIX format using a utility such as dos2unix. Do not convert the .pfa file to UNIX format. Change the permissions to readable by all users and change the .pfa file's permissions so that it is writable by all users.

With the .afm and .pfa files in the AC_SERVER_HOME/bin, use fontutils and the following options from a command line:

```
fontutils -T 1  -1"<font name>" -2<font file>.afm -3 <font
   file>.pfa -4<CodePage>
```

where font name is the name of the font used in the document and CodePage is the font's code page. Enclose the font name in double quotes. Font file is the name of the .afm or .pfa font.

Uninstalling a PostScript font

To uninstall a font, use the fontutils utility and the following options from a command line:

```
fontutils -T 2 -1"<font name>" -4<CodePage>
```

where font name is the name of the font you want to uninstall and CodePage is the font's code page. Enclose the font name in double quotes.

Obtaining the PPD file

If the manufacturer of your printer does not provide a PPD, download the Windows PostScript drivers and extract the PPD file. Put the PPD file in the AC_SERVER_HOME/operation/print/ppds. List the file in operation/print /ppds/driver_mapping.

For example, to use a the PPD file hp9000.ps with the iServer printer HP LaserJet 9000, the PPD file must be in AC_SERVER_HOME/operation/print/ppds/ and in the same directory, the driver_mapping file must contain an entry similar to the following one:

```
hp9000.ps:  "HP LaserJet 9000"
```

After updating the UNIX or Linux system, restart the cluster node for the changes to take effect. Specify the name of the PPD in Configuration Console—Advanced view by choosing Printers—Add Printer.

Sending a document to the printer

A user can send a document to the printer from Management Console. You must schedule a job to perform server-side printing. On Files and Folders, select Schedule for the document you want to print. Choose Print, and specify the

printer and other options, as shown in Figure 13-4.

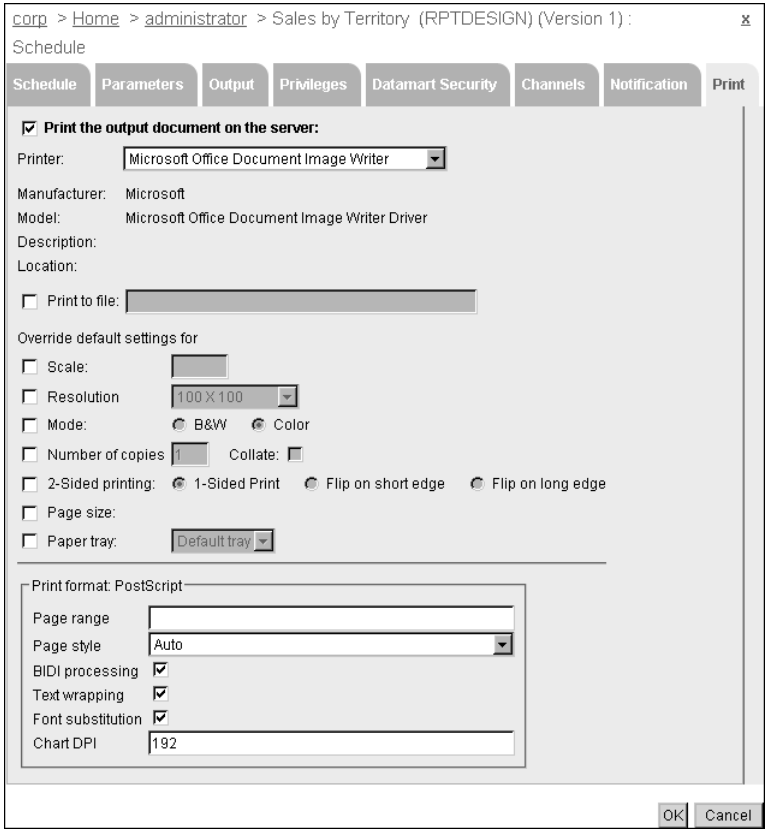

Figure 13-4 Scheduling a job to print a document

Managing the printing environment

The following sections describe how to manage the iServer document printing environment.

Changing a path to a printer

In Printers, the administrator can specify the path to a printer.

How to set the path to a printer

In the Advanced view of Configuration Console, set the path to a printer by performing the following steps:

1 Select Printers from the Advanced view.

2 Point to the arrow next to the printer name in the list, as shown in Figure 13-5.

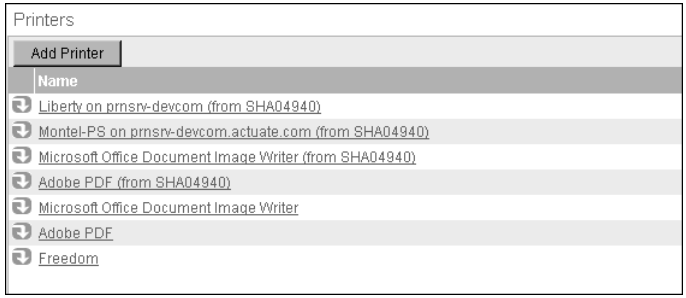

Figure 13-5 Choosing a printer name

3 Choose Template settings to change the path to a printer on Windows or to a PPD file name of a printer on UNIX or Linux.

4 On Properties, as shown in Figure 13-6, in Printer path, specify the path to the printer. For a UNIX system, specify the PPD file name and the Spool command.

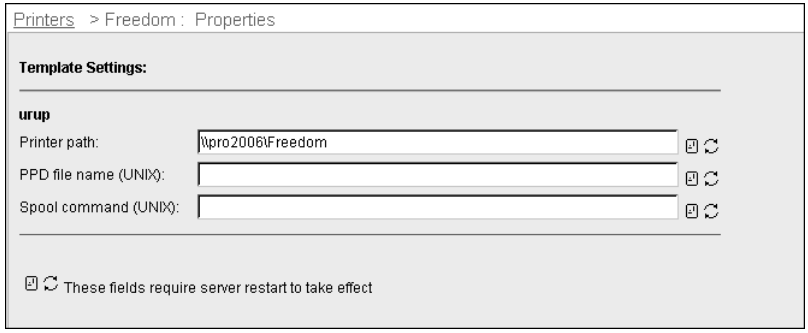

Figure 13-6 Specifying the path to a printer

Removing a printer

In Printers, the administrator can remove a printer.

How to remove a printer

To remove a printer from iServer, complete the following tasks:

1 If the printer is the default printer for an Encyclopedia volume, assign a new default printer.

2 On Printers, point to the arrow next to the printer name and choose Delete.

Configuring advanced printing parameters

You can configure iServer to limit the list of printers that users see. Configure a subset of printers in your enterprise as iServer printers. In a reasonably sized list, users can find a printer quickly to print a document. Displaying a long list of printers in Management Console can slow the application.

In System—Properties—Advanced—Printing, set Maximum number of printers to automatically configure, as shown in Figure 13-7. You can specify up to 50 printers. At startup, iServer configures the specified number of printers.

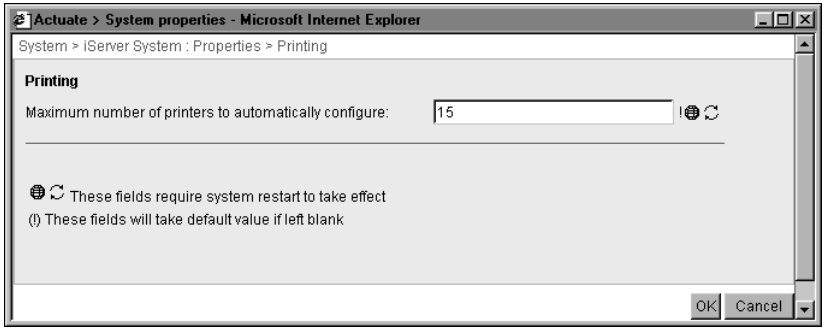

Figure 13-7 Specifying advanced printing property values

Disabling automatic printer configuration

During restart, iServer typically deletes printer information and adds the names of printers known to the operating system running iServer. Setting Maximum number of printers to automatically configure to 0 disables automatic printer configuration. iServer does not update the list of printers during restart. For example, set the value to 0 if iServer uses printers that are not detected by the operating system, such as a printer on a cluster node.

Configuring fonts

iServer supports TrueType and PostScript fonts. A report object executable (.rox) file stores information about the fonts used in an Actuate Basic report object instance (.roi) file.

iServer ships with the following Latin-1 fonts:

Arial Black	Arial Narrow Italic	Palatino Italic
Arial Black Italic	Arial Narrow Bold Italic	Times
Arial	Courier New	Times Bold

(continues)

Arial Bold	Courier New Bold	Times Italic
Arial Italic	Courier New Italic	Times Bold Italic
Arial Bold Italic	Courier New Bold Italic	Verdana
Arial Narrow	Frutiger-Roman	Verdana Bold
Arial Narrow Bold	Johnny-Bold	Verdana Italic
		Verdana Bold Italic

On UNIX systems, iServer installs Latin-1 fonts, AFM files, and at least one font set of each non-CJK language. Non-CJK languages are those languages other than Chinese, Japanese, and Korean. Each font set consists of the normal, italic, bold, and bold-italic variations of the font. For non-CJK languages, Actuate installs a localized version of a font type similar to the Arial font, including the corresponding AFM and PFA files.

Actuate works with, but does not supply the Chinese, Japanese, and Korean (CJK) fonts listed in Table 13-1.

Table 13-1 Supported CJK language fonts

Language	Font 1	Font 2
Simplified Chinese	SimSun	SimHei
Traditional Chinese	MingLiu	MS Hei
Japanese	Mincho	Gothic
Korean	Batang	Dotum

If you add a CJK font to the font properties file, the font name must be in the native language. The font's AFM file is required. iServer does not embed CJK fonts; therefore, you must print the PostScript file using a printer on which the font is installed.

iServer can embed non-CJK fonts. If a font is embedded in the output, the font's PFA file is required. For example, for a non-CJK font, the UNIX font name must be the same as the value for the FontName parameter in the PFA file.

Understanding PostScript font conversion issues

When converting a font using Fontographer 4.1 from Macromedia, choose File➛GenerateFontFile. In the dialog box, select Advanced and in the Encoding section, select Adobe Standard. Do not change any other options.

If you used Fontographer 4.1 from Macromedia, check the font's KernPair value. There is a problem in Fontographer 4.1 in which it uses incorrect KernPair values when converting some TTF fonts. After converting a font with Fontographer and

installing the font using fontutils, go to the font metrics directory and use the UNIX grep utility to search for the string KPX in the font's AFM file.

```
grep -c KPX mynewfont.afm
```

Open the .afm file in a text editor, search for the StartKernPairs variable, and check the value. If the StartKernPairs number is not the same as the number displayed using the grep utility, replace the number with the one displayed from the grep command.

Mapping fonts for charts in documents

A developer can specify fonts of text components in a chart, such as the title, data points, and axis label. To render charts properly, the fonts specified in the chart must be available to the iServer Java virtual machine (JVM).

The procedure for making fonts available differs by platform.

- In Windows, install fonts in the standard fonts folder for the operating system. For example, on Windows, from Control Panel, open Fonts. Choose Fonts and then choose File➤Install new font.
- In UNIX or Linux, copy fonts to the JAVA_HOME/jre/lib/fonts directory.

iServer uses a default font for chart elements that do not specify a font. The default font depends on the locale setting of the iServer that generates the chart. Table 13-2 lists default fonts by language. To achieve consistent appearance in output in UNIX and Windows, the default fonts must be available to the JVM.

Table 13-2 Default fonts by language

Language	Default fonts
Languages other than Chinese, Japanese, or Korean (non-CJK languages)	arial.ttf arialbd.ttf arialbi.ttf ariali.ttf times.ttf timesbd.ttf timesbi.ttf timesi.ttf
Simplified Chinese	simsun.ttc
Traditional Chinese	mingliu.ttc
Japanese	msgothic.ttc msmincho.ttc
Korean	gulim.ttc batang.ttc

Locating fonts when generating BIRT documents

To render or generate BIRT design output, iServer looks for font configuration information in the AC_SERVER_HOME/Jar/BIRT/platform/plugins /org.eclipse.birt.report.engine.fonts directory.

BIRT designs use five different types of font configuration files. The font configuration file naming convention includes information about the rendering format, the system platform, and the system locale, as shown in the following template:

```
fontsConfig_<Format>_<Platform>_<Locale>.xml
```

BIRT supports the following levels of font configuration files:

- For all rendering formats:
  These files have no format specifier in their names. The configuration files are divided into three sub-levels:
  - The default configuration file:

    ```
    fontsConfig.xml
    ```

  - Configuration files for a specific platform, for example:

    ```
    fontsConfig_Windows_XP.xml
    ```

  - Configuration files for a specific platform and locale, for example:

    ```
    fontsConfig_Windows_XP_zh.xml
    fontsConfig_Windows_XP_zh_CN.xml
    ```

- For certain formats only
  These files include the format specifier in their names. These configuration files are divided into three sub-levels:
  - The default configuration file for a format, for example:

    ```
    fontsConfig_pdf.xml
    ```

  - Configuration files for a format for a specific platform:

    ```
    fontsConfig_pdf_Windows_XP.xml
    ```

  - Configuration files for a format for a specific platform and locale:

    ```
    fontsConfig_pdf_Windows_XP_en_AU.xml
    ```

iServer first looks for the font location in the font configuration files. If the font is not found, iServer searches for the font in the system-defined font folder. An exception to this search order occurs if iServer encounters a <fonts-path> section in one of the font configuration files. iServer searches each path coded in the <fonts-path> section instead of searching the system-defined font folder.

Locating fonts when generating Actuate Basic documents

The default behavior is for iServer to use the font information in customized_fonts.rox and master_fonts.rox. If the font information is not in customized_fonts.rox or master_fonts.rox, iServer searches the report executable file. The master_fonts.rox file includes standard Windows fonts.

Design developers create a customized_fonts.rox to be used alone or with master_fonts.rox. When used alone, the customized_fonts.rox file includes only the fonts that are not included in report executable files. When used with master_fonts.rox, the customized_fonts.rox file includes fonts that are not included in report executable files and are not in the master_fonts.rox file. Actuate recommends creating a customized_fonts.rox file and not modifying master_fonts.rox. If the customized master fonts file and master_fonts.rox include the same font, the font metrics in the customized master fonts file have precedence. To create a customized_fonts.rox file, use Actuate e.Report Designer Professional.

Name the customized master fonts file customized_fonts.rox and place it in AC_SERVER_HOME/etc. In an iServer cluster, place customized_fonts.rox on every node that has the View service or Factory service enabled. If you are using a customized master fonts file that contains all the fonts you use, performance may improve slightly if you rename master_fonts.rox.

Configuring font searching

You can configure iServer to search for font information in the following situations:

- A View process displays an Actuate report object instance (.roi) file in DHTML format or generates a PDF, and the fonts are not in the ROX.
- iServer runs an ROX that contains a dynamic text control.
- An ROX generates Excel data, and the data uses fonts that are not in the ROX.

A dynamic text control contains variable-length text. The text can be in HTML, RTF, or plain text format. iServer can use the HTML or RTF information to format the data in a document. The HTML or RTF formatting information can contain font information that is not stored in the report object executable (.rox) file. If font information is not in the report object executable (.rox) file, iServer searches customized_fonts.rox and master_fonts.rox to locate the font information.

Filtering font searches to find metrics

In System—Properties—Advanced—Information Display—Fonts, the administrator can specify the search order that iServer uses when looking for font metric information, such as character width and height, by setting Use externalized font file, as shown in Figure 13-8. Use externalized font file sets the

UseExternalizedFonts parameter in iServer.

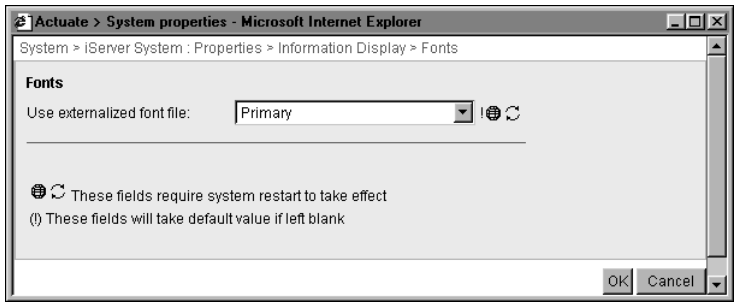

Figure 13-8 Specifying font metrics search order

Table 13-3 describes the values for the UseExternalizedFonts parameter.

Table 13-3 Values for the UseExternalizedFonts parameter

Value	Description
no	Do not use externalized fonts. The View process does not use externalized fonts. The View process uses the font information in the report object executable (.rox) file.
primary	Look for the font in customized_fonts.rox and master_fonts.rox first. If iServer cannot locate the font in customized_fonts.rox or master_fonts.rox, it looks for the font in the report object executable (.rox) file. Primary is the default value.
secondary	Look for the font in the ROX first. If iServer cannot locate the font in the ROX, it looks for the font in customized_fonts.rox and master_fonts.rox.

iServer uses the database client locale settings when it establishes a connection with the database. To use Actuate report object executable (.rox) files that access databases with data in multiple languages, you must configure your database clients to retrieve and view data in those languages.

Most databases support Unicode UTF-8 encoding. Actuate recommends setting the database client encoding to UTF-8. For UTF-8 encoding, Actuate supports only the characters in the UCS-2 character set.

Chapter

14

Connecting to data sources

This chapter contains the following topics:

- About data source connections
- Using a connection configuration file
- Defining environment variables
- Connecting to data sources
- Using a connection pool

About data source connections

Actuate Customer Support publishes the Supported Products and Obsolescence Policy document that describes the data sources, drivers, operating systems, and other software requirements for connecting iServer to data sources. Actuate Supported Products and Obsolescence Policy, available on the Actuate Support site at the following URL, also contains information about the required patches:

http://support.actuate.com/documentation/spm

iServer connects to data sources when generating documents and using the Actuate Caching service (ACS). The design developer specifies data source connection information in the design, or in an external connection configuration file. In most cases, iServer and the database run on different computers for load-balancing purposes, but this division is not mandatory. Running iServer on the database host can improve performance.

About drivers

The iServer installation process installs and configures DataDirect Connect for ODBC 5.3 drivers and JDBC 3.7 SP1 drivers. You can also use third-party drivers to connect to data sources, but you must license, install, and configure them.

To make a JDBC connection with an Oracle database, iServer uses the DataDirect JDBC driver that installs with iServer only.

About information object connections

You can use information objects in designs. Information objects make ODBC and JDBC connections, so you need to be able to connect underlying data sources to databases in addition to the obvious data sources required by your design.

Using a connection configuration file

A connection configuration file is an XML file, such as the one shown in Listing 14-1, in UTF-8 or ASCII encoding. The file specifies the data source connection properties to use when iServer runs a design. Having the data source connection information for a design in an external file makes it convenient to modify. You change the connection information without altering the design. You specify the location of the file using Configuration Console.

You can use an external connection configuration file to define a data source for the Actuate Caching service and for a data connection definition (.dcd) file, which contains information object connection properties for a data source. You can also

use an external connection configuration file for connecting data sources to designs.

You can create an external connection profile to a data source used by a design. Changes to the profile are automatically picked up by the design. The settings in a connection configuration file override any connection configuration properties in the connection profile. The sample connection configuration file in Listing 14-1 externalizes the file path to the Connection Profile, C:\SqlServer.profile.

Listing 14-1 BIRT connection configuration file example

```
<oda-data-source
   extensionID="org.eclipse.birt.report.data.oda.jdbc" name="JDBC
   Data Source - SQL Server" id="783">

   <property name="odaDriverClass">com.actuate.jdbc.sqlserver.
      SQLServerDriver
   </property>

   <property name="odaURL">jdbc:actuate:sqlserver://DBSRV1-W2K
   </property>
</oda-data-source>

<ConnectOptions Type=".eclipse.birt.report.data.oda.jdbc_ JDBC
   Data Source - SQL Server ">

   <Property PropName="OdaConnProfileStorePath">C:\Mypath
   </Property>

</ConnectOptions>
```

In a BIRT design, the configuration key used to specify a data source is the unique ID of the ODA data source extension and data source name defined in the BIRT design or library. You must concatenate the string as follows:

```
extensionID + "_" + data source name
```

For example, the key is org.eclipse.birt.report.data.oda.jdbc_SQL Server.

Changing a configuration file

The Factory process reads the configuration file for the configuration key values when the process starts. After changing a configuration file, you must restart Factory processes for changes to take effect. Only Factory processes that start after changes in the configuration file use the new information. To ensure that design files use updated configuration file information, confirm that no document generation jobs are running and stop Factory processes that are running before you change the configuration file. After changing the file, iServer starts a Factory process for the next document generation job.

Specifying the location of the connection configuration file

There is no default location for the connection configuration file. To use a connection configuration file, create the file and then specify its name and location using the ConnConfigFile parameter in Configuration Console.

From Server Configuration Templates—Settings, expand iServer, then choose Database Connection Configuration File. Specify the location of the file using the Configuration file for database connections and search path parameter shown in Figure 14-1.

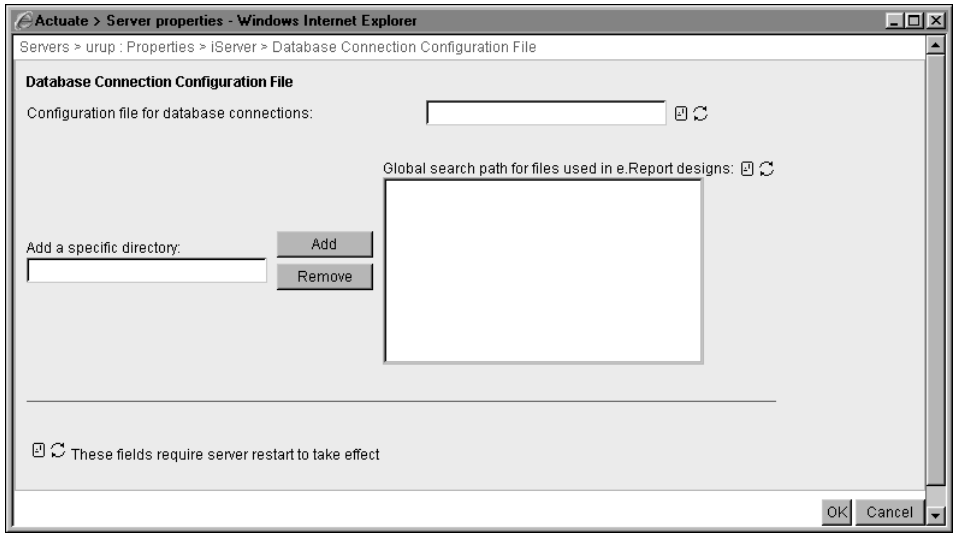

Figure 14-1 Specifying the location of a connection file

On UNIX and Linux, the value of the parameter can be a path and file name only. On Windows, it can be either a path and file name or a URL. For example:

```
\\server1\configs\serverconfig.xml
```

or

```
http://myserver/configs/testconfig.xml
```

If you do not specify a value for the configuration file parameter, iServer uses the data source connection properties in the design.

Configuring a cluster to use a connection configuration file

In a cluster of iServers, each node must have access to the connection configuration file. The path can be a local absolute path on each machine and must be specified for each iServer. If you use a single copy of the file for a cluster,

put it in a shared location, then specify the path to that shared location for all iServers in the cluster.

Defining environment variables

The installation program defines environment variables for data sources that you specify during installation, but if you want to connect to data sources later, you need to set these variables.

On UNIX and Linux, insert the library path environment definition in the file pmd11.sh in AC_SERVER_HOME/bin. The library path variable names are:

- LD_LIBRARY_PATH for Sun Solaris and Linux
- LIBPATH for IBM AIX
- SHLIB_PATH for HP-UX

On Windows, you define system, not user, environment variables. iServer processes are system processes, and cannot access the user environment. Windows uses PATH to search for libraries and executable files.

In some cases, environment variables or registry entries affect the capability to display documents in different languages.

Table 14-1 lists data source-related Actuate registry keys on Windows and environment variables on UNIX, Linux, and Windows.

Table 14-1 Environment variables

Key or variable name	Description
AC_DBMS_INFORMIX_MAXVARLEN	Maximum column length Actuate uses with Informix data sources. Default value is 4000.
AC_DBMS_ODBC_MAXVARLEN	Maximum column length Actuate uses with ODBC data sources. Default value is 8000.
AC_DBMS_ORACLE_MAXVARLEN	Maximum column length Actuate uses with Oracle data sources. Default value is 4000.
AC_DBMS_PROGRESS_MAXVARLEN	Maximum column length Actuate uses with Progress data sources. Default value is 4000.
DB2COMM	DB2 information.

(continues)

Table 14-1 Environment variables (continued)

Key or variable name	Description
DB2DIR	The path to the DB2 client installation.
DB2INSTANCE	Specifies the DB2 instance name.
DISPLAY	UNIX and Linux. Specifies the X Windows server used by the machine.
DLC	Progress 9.1 installation directory.
INFORMIXDIR	The directory where the Informix product is installed.
INFORMIXSERVER	Specifies the name of the Informix data source.
INSTHOME	Inherited from shell and used by third-party processes.
ORACLE_HOME	The path to the Oracle installation.
SYBASE	The path to the Sybase installation.
SYBASE_OCS	The path to the Sybase OpenClient installation.

Connecting to data sources

The following sections describe how to connect to various external data sources from iServer system.

Connecting to a data source through ODBC

The iServer data source connection interface uses DataDirect Connect ODBC 6.0 drivers to connect to data sources through ODBC on UNIX, Linux, and Microsoft Windows. During installation on UNIX and Linux, you can choose to have ODBC configured. If your designs use information objects to access data sources, you need ODBC. For information about configuring DataDirect ODBC drivers, iServer installs DataDirect help files. The DataDirect help files are in the following Actuate ODBC directory:

Windows: \Program files\Common Files\Actuate\11.0\odbc\help

UNIX and Linux: AC_SERVER_HOME/odbc/help

About the ODBC initialization file

To connect to a data source through ODBC, iServer has the following requirements:

- Access to ODBC data source connection information.

  iServer installs and configures the ODBC initialization file in the /odbc directory, partially shown in Listing 14-2. To modify the initialization file on Windows, use the ODBC Data Source Administrator.

  iServer sometimes requires additional information to access the data from the data source. For example, when iServer connects to a data source through ODBC, it uses a user name and password specified in the design to access data. The design can also contain other information iServer needs to connect to an ODBC data source such as a value for an environment variable, a set of connection parameters, or a software library name.

- Access to the ODBC data source itself.

  The iServer account must have sufficient privileges to access the data source. For example, a DBMS managing a database is on a different machine from the machine on which the data source resides. iServer must have access to the machine, and must be able to connect to the data source.

- To access ODBC data sources, iServer on Microsoft Windows must be configured to run as a user account, not as the system account.

- On UNIX and Linux, to make a DataDirect ODBC connection, perform the following tasks:

  - In an Actuate Basic design, set DLLPath to ODBC32 in the Component Editor for report design.
  - Confirm that a symbolic link exists between the ODBC32 file under AC_SERVER_HOME/lib and the DataDirect ODBC manager library file. For example, AC_SERVER_HOME/lib/odbc32.so -> libodbc.so for the Solaris platform.
  - When connecting to a Microsoft SQL Server data source, set the UNIX or Linux environment variable LC_MESSAGES to en_US to control message formatting between the data source and iServer.

Listing 14-2 shows portions of the odbc.ini installed with iServer on Linux.

Listing 14-2 Example of odbc.ini

```
[ODBC Data Sources]
dBASE=ActuateDD 5.3 dBASEFile (*.dbf)
DB2 Wire Protocol=ActuateDD 5.3 DB2 Wire Protocol
Informix Wire Protocol=ActuateDD 5.3 Informix Wire Protocol
Oracle Wire Protocol=ActuateDD 5.3 Oracle Wire Protocol
```

```
Sybase Wire Protocol=ActuateDD 5.3 Sybase Wire Protocol
Text=ActuateDD 5.3 TextFile (*.*)
Teradata=ActuateDD 5.3 Teradata
SQLServer Wire Protocol=ActuateDD 5.3 SQL Server Wire Protocol
Progress OpenEdge=ActuateDD 5.3 Progress OpenEdge
Progress SQL92=ActuateDD 5.3 Progress SQL92
MySQL Wire Protocol=ActuateDD 5.3 MySQL Wire Protocol
FoxPro3=ActuateDD 5.3 dBASEFile (*.dbf)
_
[Oracle Wire Protocol]
Driver=/home/Actuate/Actuate11/AcServer/odbc/lib/N_ora23.so
Description=ActuateDD 5.3 Oracle Wire Protocol
AlternateServers=
ApplicationUsingThreads=1
ArraySize=60000
AuthenticationMethod=1
CachedCursorLimit=32
CachedDescLimit=0
CatalogIncludesSynonyms=1
CatalogOptions=0...
```

Understanding language encoding

If you use third-party drivers instead of the DataDirect Connect for ODBC drivers installed with iServer, you need to consider encoding requirements. iServer can use either ODBC Unicode drivers or ODBC ANSI drivers. Unicode drivers support wide-character versions of ODBC interfaces, such as SQLConnectW( ). ANSI drivers support the non-wide versions, such as SQLConnect( ). For data sources that return Unicode, non-ASCII data, iServer requires a Unicode ODBC driver. For data sources that return only ASCII data, iServer can use either a Unicode driver or an ANSI driver.

Most ODBC driver managers, such as DataDirect, try to detect the code page or the Unicode encoding in which the database communicates with the driver. Use the environment variable AC_DBC_ENCODING if the driver manager fails to detect the appropriate code page or the Unicode encoding. The driver manager can use an incorrect default value and send or receive garbled data.

The environment variable AC_DBC_ENCODING specifies the code page or the Unicode encoding used by the ODBC driver manager while interpreting data from the driver. The AC_DBC_ENCODING value can be UCS-2, UTF-8, or ANSI. The ANSI value indicates the currently set code page of the OS locale.

Setting the environment variable AC_DBC_ENCODING to UTF-8 when using DataDirect ODBC 5.3 drivers is not required and can actually cause problems.

Actuate's ODBC interface eliminates the need for a driver manager. The ODBC interface detects whether the driver supports Unicode by making a call to SQLConnectW( ). If the driver supports this method, the ODBC interface uses the

driver's Unicode methods with a W suffix. If the driver does not have the Unicode version of the methods, the Actuate ODBC interface assumes the driver is a non-Unicode ANSI driver and calls the ODBC API methods without the W suffix.

Setting the maximum column length

The default maximum column length Actuate permits with ODBC data sources is 8000 characters. Problems can occur when generating an Actuate document if the design uses an ODBC data source column with a column length greater than 8000 characters.

To change the maximum character length, set the environment variable AC_DBMS_ODBC_MAXVARLEN.

Connecting to a DB2 data source

iServer can connect from Actuate Basic designs to IBM DB2 data sources on UNIX and Windows using run-time client version 9.1, 9.5, and 9.7. To connect to DB2 data sources, you must perform the following tasks:

- Define appropriate environment variables.
- Specify the database environment.
- Supply an account name and password.
- Specify the protocol your site uses.
- Configure the DB2 client's locale setting to match the locale of the machine.

The Actuate interface to DB2 clients supports DB2 stored procedures. From the Stored Procedure Data Source Builder in Actuate e.Report Designer Professional, developers can use DB2 stored procedures to retrieve data to generate documents. The following limitations of DB2 stored procedures exist:

- Due to a problem with the DB2 CLI API, the Stored Procedure Data Source Builder does not support DB2 overloaded stored procedures.
- The Actuate Stored Procedure Browser does not list DB2 user-defined functions. DB2 user-defined functions do not return result sets and are not a data source.

Other DB2 issues are:

- DB2 does not support large object (LOB) data types between versions. See the DB2 documentation for information about DB2 support for LOB data types.
- You must bind the DB2 CLI packages from the client system to the DB2 server using db2cli.lst in some situations. For example, after you apply a Fix Pak to a client or server, or on a system where an Actuate design uses DB2 9.1 client software with a DB2 9.1 server.

Defining DB2 environment variables

For UNIX and Microsoft Windows operating systems, you must define the following environment variables:

- DB2INSTANCE and DB2DIR

  Define these variables to connect to a DB2 instance. DB2INSTANCE specifies the instance name, and DB2DIR is the path to the DB2 client installation.

- DB2CODEPAGE

  DB2CODEPAGE is a DB2-specific environment variable. For Windows and UNIX, Actuate uses DB2CODEPAGE to determine the DB2 database's client locale. On Windows systems, DB2CODEPAGE is a registry setting. On UNIX systems, DB2CODEPAGE is an environment variable. At execution time when Actuate connects to a DB2 data source, the active code page is in effect for the duration of the connection. All data is interpreted based on this code page. If this variable is not set, Actuate's DBMS module determines the client locale setting from the operating system locale.

Use db2set to set DB2CODEPAGE. For example, the following command sets the DB2 database client to retrieve data in UTF-8 format:

```
db2set DB2CODEPAGE=1208
```

When using the db2set command, add the location of db2set to the environment variable PATH. For example, if db2set is in $DB2DIR/adm and you use db2set in the pmd11.sh shell script, add the location of db2set to the environment variable PATH in pmd11.sh.

About the AIX DB2 library path

The DB2 Factory server on AIX uses the DB2 library libdb2.a. The library libdb2a is part of the DB2 client installation. Ensure that the library path DB2DIR/lib is part of the environment variable LIBPATH. DB2DIR is the path to the DB2 client installation.

About the HP-UX11i Version 1 library path

The definition of the HP-UX11i in the environment variable SHLIB_PATH must put the path to ODBC and DB2 libraries before the AC_SERVER_HOME path. For example, if ODBC/lib and DB2DIR/lib are paths to the ODBC and DB2 libraries, use the following SHLIB_PATH definition:

```
SHLIB_PATH=$ODBC/lib:$DB2DIR/lib:$AC_SERVER_HOME/lib:usr/local
   /bin:
```

About the Solaris library path

The path to ODBC and DB2 libraries must precede the AC_SERVER_HOME path in the Solaris LD_LIBRARY_PATH environment variable. For example, if

ODBC/lib and DB2DIR/lib are paths to the ODBC and DB2 libraries, use an LD_LIBRARY_PATH as shown in the following example:

```
LD_LIBRARY_PATH=$ODBC/lib:$DB2DIR/lib:$AC_SERVER_HOME/lib:/usr
   /local/bin:
```

Because of the requirements for the LD_LIBRARY_PATH environment variable, the following path order does not work:

```
LD_LIBRARY_PATH=$AC_SERVER_HOME/lib:$ODBC/lib:$DB2DIR/lib:/usr
   /local/bin
```

Using DB2 libraries on AIX, HP-UX, and SunOS

If you did not specify DB2 data source information during installation of iServer, and you now want to use DB2 data source connections from the machine, you must create a symbolic link to the DB2 library in the DB2 client installation directory.

Linking to a DB2 library on HP-UX or SunOS

On Solaris, create a symbolic link DB2CLI.so in AC_SERVER_HOME/lib to the DB2 library libdb2.so.

On HP-UX, create a symbolic link DB2CLI.sl in AC_SERVER_HOME/lib to the DB2 library libdb2.sl.

Linking to a DB2 library on AIX

On AIX, perform the following procedure:

1. Extract shr.o from $DB2DIR/lib/libdb2.a by using the following command:

   ```
   ar -x libdb2.a
   ```

2. Rename the extracted file to db2.o using:

   ```
   mv shr.o db2.o
   ```

3. On AIX, create a symbolic link DB2CLI.sl in AC_SERVER_HOME/lib to the DB2 library ln -s db2.o.

Checking a connection to a DB2 instance

To check that a connection exists between the machine and the DB2 instance, use the command-line utility DB2. This utility comes with the DB2 software and is available for UNIX, Linux, and Windows.

To start the DB2 utility, open a command-line window and enter DB2 at the command prompt. At the DB2 prompt, enter the command to connect to a DB2 database.

```
connect to <database> user <user name>
```

The <database> is the name of the DB2 database, and <user name> is the DB2 database user. You are prompted for a password. Enter the password for the user. DB2 displays the connection information in the command window when a connection is made. Enter quit to terminate the session.

About using XML Extender

The XML Extender component of DB2 provides data types to store XML documents and DTDs in DB2 databases as either an XML column or XML collection. Actuate retrieves the XML column data as string data. DB2 stores an XML collection as a set of tables. Actuate retrieves data from the set of DB2 tables.

Connecting to an Informix data source

When accessing data from Informix using the included DataDirect ODBC Informix data driver, set the UseDelimitedIdentifier property to 1. To specify the property value for connection-string based designs, add the property UseDelimitedIdentifier=1 to the end of the connection string. To specify the property value for DSN-based connections, perform the appropriate task for your operating system.

On a Windows system, go to the Windows ODBC registry entry:

```
HKLM\Software\ODBC\ODBC.ini\<DSN Name>
```

and set the value of UseDelimitedIdentifier to 1.

On a UNIX or Linux system, edit the odbc.ini file, and add UseDelimitedIdentifier=1 to the DSN entry.

Connecting to an Oracle data source

To connect with Oracle data sources, you must perform the following tasks:

- Install the proper Oracle client software on the server running the design.
- Supply a connection string.
- Define the appropriate environment variables.
- Ensure that a listener process is running on the database host.

When configuring the Oracle client, you must configure the client's locale setting to match the locale of the machine. In an Actuate report object executable (.rox) file, the default DBInterface for AcOracleConnection is acorcl111.

Defining Oracle environment variables

You must set the following environment variables when connecting to an Oracle data source:

- ORACLE_HOME
- NLS_LANG

About ORACLE_HOME

For Oracle databases on UNIX and Linux platforms, the account running iServer processes must have a definition for the standard Oracle environment variable ORACLE_HOME. You can provide this definition in a login script such as .cshrc or .profile, or you can include it in the scripts that start the iServer processes.

For Oracle databases on Windows, also ensure that the definition of the environment variable PATH indicates the location of the dynamic link library that selects the proper database.

About NLS_LANG

The Oracle UNIX and Linux environment variable and Windows registry setting NLS_LANG specifies the Oracle locale, that consists of the language, territory, and character set. The default value for NLS_LANG is American_America.US7ASCII. The administrator must ensure the NLS_LANG setting is correct for the information in the Oracle database.

On UNIX and Linux systems, add the environment variable NLS_LANG to the pmd11.sh script, the iServer request server startup script. On Windows servers, the Oracle installer configures NLS_LANG.

The following example sets NLS_LANG for simplified Chinese on UNIX and Linux:

```
export NLS_LANG
NLS_LANG="Simplified Chinese_China.ZHS16GBK"
```

Double quotes are required when setting a value that contains spaces.

On Microsoft Windows server operating systems, set the NLS_LANG registry value in the key HKEY_LOCAL_MACHINE \Software\Oracle\Home.

For information about the NLS_LANG values, see the Oracle documentation.

About the Oracle listener processes

iServer interacts with an Oracle data source through the Oracle listener process. If iServer is having trouble communicating with the Oracle data source, the link to the listener process possibly failed. You can frequently solve such problems by stopping and starting the listener process.

Setting the maximum column length

The default maximum column length Actuate uses with Oracle data sources is 4000 characters. Problems can occur when generating an Actuate document if the design uses an Oracle data source column with a large column length. You can

use a registry setting on Windows or an environment variable on UNIX and Linux to set a smaller maximum column length used by Actuate.

To change the maximum character length:

- On Windows, change the value of the string value name MaxVarLen. The value name is in the registry key.

  ```
  HKEY_CURRENT_USER\Software\Actuate\DBMS\Oracle
  ```

- On UNIX and Linux, set the environment variable AC_DBMS_ORACLE_MAXVARLEN.

Connecting to custom data sources

iServer runs designs using third-party software to connect to data sources. The following sections describe how to install custom data source connection software:

- Installing a custom open data access (ODA) driver
- Installing a custom Eclipse DTP ODA driver plug-in
- Using custom Java-based data source connections
- Using custom relational data sources with the Integration service

Installing a custom open data access (ODA) driver

iServer supports open data access (ODA) drivers that a design uses to retrieve data from a data source. To install an ODA driver, place the ODA driver files in the AC_SERVER_HOME/oda directory. AC_SERVER_HOME refers to the folder that the iServer installer chose as the location for binary files during the iServer installation. By default, this location is /Program Files/Actuate11SP4/iServer in a Windows environment, and <installation directory>/AcServer in a Linux or UNIX environment.

Each ODA driver must be in a separate directory. For example, if you have two ODA drivers, the installation directories would be similar to the following directories:

- AC_SERVER_HOME/oda/oda-driver1
- AC_SERVER_HOME/oda/oda-driver2

iServer does not support installing ODA drivers in a directory below the oda directory. For example, if you have two ODA drivers, Driver1 and Driver2, iServer does not support the following ODA directory structure for the two drivers:

- AC_SERVER_HOME/oda/CustomDrivers/Driver1
- AC_SERVER_HOME/oda/CustomDrivers/Driver2

The directory name for an ODA driver in the AC_SERVER_HOME/oda directory must match the driver name specified in an Actuate Basic report (.rox) or Actuate Basic information object (.dox).

When installing an ODA driver, you must also install and configure any software the ODA driver requires to access a data source. For example, you must install and configure any database connection software the ODA driver uses to connect to a database. You must also ensure the ODA driver can access the required software.

Install the configuration file and files required for the ODA driver on the machine that runs the design. Each ODA driver requires a run-time configuration file, odaconfig.xml. The configuration file must be in the ODA driver directory. You can locate the ODA library files outside of the Actuate installation directory. Specify the location in odaconfig.xml.

The iServer installation process puts the Actuate Data Integration service driver in the AC_SERVER_HOME/oda directory. The installation also provides an example of an ODA flat file driver that operates with Actuate e.Report Designer Professional and iServer in ACTUATE_HOME/oda/examples/FlatFileExample.

Installing a custom Eclipse DTP ODA driver plug-in

A design can use a custom Eclipse Data Tools Platform (DTP) open data access (ODA) driver to retrieve data from a data source. The DTP ODA API supports building a custom Eclipse plug-in that accesses data from standard and custom data sources.

Install a plug-in in the AC_SERVER_HOME/MyClasses/eclipse/plugins directory. You may need to manually create the eclipse/plugins subdirectory if it does not already exist in MyClasses. In the UNIX and Linux installation of iServer, there is a common folder, acshared/MyClasses, for installing a customized DTP ODA driver.

Do not place a custom ODA plug-in in the AC_SERVER_HOME/oda/eclipse /plugins directory. This directory is reserved only for plug-ins provided by the Actuate product installation. Installing a plug-in in the AC_SERVER_HOME /MyClasses/eclipse/plugins directory ensures that the Actuate install and uninstall processes do not remove the custom plug-in.

About using an ODA driver

When running a design using an ODA driver, the Factory process loads the driver during the document generation. If the Factory process cannot load the specified driver, iServer logs an error message and document generation terminates.

An ODA driver cannot share a library file with another ODA driver or data source connection software. Each ODA driver must have a separate copy of the library file for dedicated use. For example, on a Windows system, if a database

library uses a DLL to connect to a data source, and an ODA driver uses the same DLL, you create a copy of the DLL file and use the copy with the ODA driver.

If you change an ODA driver's configuration such as a setting in the configuration file or an ODA driver library, the Factory process uses the updated configuration information and the updated library during the next document generation.

iServer can cache an ODA driver. If a Factory process uses a cached ODA driver, the Factory process checks the last modified time of the configuration file and the cached driver's run-time libraries before document generation. If you modify the configuration file or the driver since last loading the driver, the Factory process releases the cached driver and reloads it.

Using custom Java-based data source connections

To connect spreadsheet designs to JDBC data sources, place the Java archive (.jar) files for the custom database driver in the following location:

```
AC_SERVER_HOME\iServer\reportengines\engines\ess\lib
```

To connect BIRT designs to JDBC data sources, place the .jar files for the custom database driver in:

```
AC_SERVER_HOME\Jar\birt\platform\plugins\
   org.eclipse.birt.report.data.oda.jdbc<version>\drivers
```

The iServer Integration service also looks in this location for drivers to access JDBC data sources.

Using custom relational data sources with the Integration service

iServer connects to Actuate information object data sources using the Actuate iServer Integration service (AIS). AIS works with JDBC drivers that are compliant with DataDirect Connect for JDBC version 3.7 Service Pack 1and later.

AIS uses configuration files that define the data source connections.

- AC_SERVER_HOME/etc/data_integration/datasources.xml contains a list of custom relational data sources.
- AC_SERVER_HOME/etc/data_integration/<database>/mappings.xml contains the data source mappings for a custom relational data source.

Specifying connection types

iServer installation provides an example of a datasources.xml configuration file that specifies a MySQL Enterprise connection type as an example. The datasources.xml file specifies the JDBC connection type as shown in Listing 14-3.

Listing 14-3 Specifying JDBC connection types in datasources.xml

```
<DataSourceConfig>

   <DataSourceHosts>
   </DataSourceHosts>

   <ConnectionTypes>

      <
-- Example: MySQL Enterprise connection type
        (requires MySQL driver) -->

      <ConnectionType Name="MySQL_41">
         <JDBCDriver DriverName="com.mysql.jdbc.Driver">
            <ConnectionString>
               jdbc:mysql://%server%:%port%/%database%?
            </ConnectionString>
            <ConnectionProperties>
                  <Property Name="user">%username%</Property>
                  <Property Name="password">%password%</Property>
            </ConnectionProperties>
            <LibraryPath>
               <
-- Fill in JAR location below -->
               <Location></Location>
            </LibraryPath>
         </JDBCDriver>
         <ConnectionParams>
            <ConnectionParam Name="server"
                  Display="Server"
                  Type="string"
                  ValueIsCaseSensitive="false"/>
            <ConnectionParam Name="database"
                  Display="Database"
                  Type="string"
                  ValueIsCaseSensitive="false"/>
            <ConnectionParam Name="username"
                  Display="User name"
                  Type="string"/>
            <ConnectionParam Name="password"
                  Display="Password"
                  Type="masked"/>
            <ConnectionParam Name="port"
                  Display="Port"
                  Type="integer"
                  Optional="true"
                  DefaultValue="3306"/>
```

```
         </ConnectionParams>
      </ConnectionType>

   </ConnectionTypes>

   <DatabaseTypes>
      <
-- Example: MySQL data base type -->
      <
--
      <DatabaseType Name="MySQL_41" DisplayName="MySQL 4.1"
         ConnectionType="MySQL_41"
         DataSourceMapper="BaseMySQLMapper"
         DataSourceMapping="MySQL"/>
      -->
   </DatabaseTypes>
</DataSourceConfig>
```

Specify the location of the JDBC driver in <Library Path>, as shown in Listing 14-3. Place the driver classes or Java archive (.jar) files accessing JDBC data sources in the /drivers directory, as described earlier in this document.

Accessing data sources using Actuate Analytics

iServer supports the following types of data sources when using Actuate Analytics:

- ODBC data sources
- Text files
- XML files
- Executable files

Actuate Analytics Option licenses you to run cube profiles to generate multidimensional data cubes in an Encyclopedia volume. Generating a cube requires an Actuate iServer license with the Actuate Analytics Option enabled.

Generating a cube using ODBC

On UNIX, to a access a data source using ODBC for cube generation, you must configure iServer to use the proper software as follows:

- Use the Actuate Analytics ODBC open server driver.
- Set the ODBC manager library in the iServer library path.

The default Actuate Analytics open server driver named AcDBDrv is in AC_SERVER_HOME/drivers. The Actuate Analytics ODBC open server driver, acdbdrv.odbc, supports all Actuate Analytics supported data source access

including ODBC. To use the ODBC driver, save and move or rename the original acdbdrv file, and rename the file acdbdrv.odbc to acdbdrv. These changes must be done while iServer is not generating a cube. Ensure the ODBC manager library is named libodbc.so on Solaris, libodbc.sl on HP-UX, and libodbc.a on AIX, and ensure that this file is the Actuate iServer library path.

Accessing data using Microsoft Analysis Services

The Actuate Analytics Option supports accessing data from a multidimensional cube stored in a Microsoft SQL database using Microsoft Analysis Services. iServer and the MS SQL Server must run on machines that use the same locale to access the Microsoft SQL cube data.

Using a connection pool

Using the ODA Java interface, a developer can create an ODA driver that pools connections. For example, when a Factory process uses an ODA driver, the driver can create a connection to the data source. When the Factory process requests another connection to the data source, the ODA driver can return a new connection or reuse the previous connection instance. Connection pooling optimizes application performance and improves scalability.

You can use an application server, such as a J2EE application server, to implement a connection pool to a data source that supplies data to iServer. In a typical J2EE application server environment, the connection pool uses a data access object (DAO) to provide a common interface between iServer and the data storage system. A DAO separates the application logic from the data access logic to provide a re-usable, persistent connection.

Figure 14-2 shows a J2EE application server configuration that supports multiple Information Console client sessions. Each viewed document contains data extracted by an iServer from a data source using a shared connection.

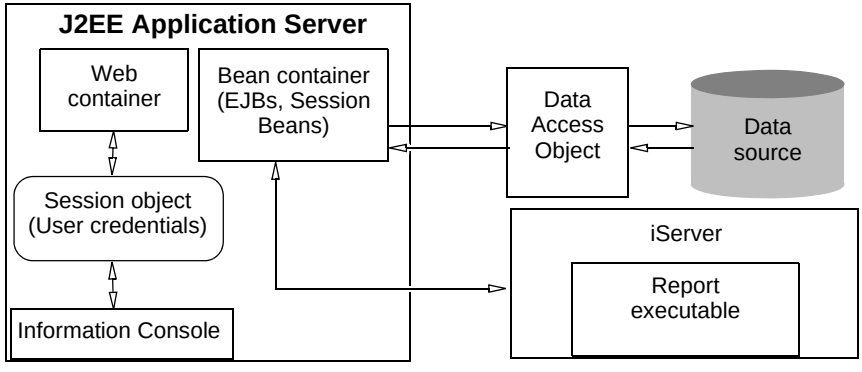

Figure 14-2 Implementing a connection pool using a J2EE application server

You need the Deployment Kit option to enable connection pooling for Actuate Java Components.

ODBC/JDBC connection pooling

Connection pooling allows applications to share a connection for greater efficiency. Each Actuate product connects to an external data source using a specified data access mechanism. Table 14-2 lists the type of connections available for Actuate report types, such as an out-of-the-box (OOTB) JDBC driver.

Table 14-2 Actuate connection types available by product

Report Type	OOTB ODBC	OOTB JDBC	ODBC	JDBC	Information Object
BIRT	No	Yes	No	Yes	Yes
e.Spreadsheet	No	No	No	No	Yes
e.Report	Yes	No	No	No	Yes
Analytics	No	No	No	No	No

An OOTB ODBC connection is available only on Windows in versions prior to Actuate Release 11.

Configuring ODBC connection pooling

After iServer creates an ODBC connection to a data source, it stores the connection in a pool of connections, and reuses the connection as needed. Connection pooling is available with ODBC 3.0 and later.

The DataDirect 5.3 ODBC drivers that ship with iServer are thread-safe, which connection pooling requires. You enable ODBC connection pooling by performing the following tasks:

- In the design, set up a DataDirect ODBC DSN-less connection. For example, using e.Report Designer Professional, in DBConnection Properties, set the ConnectionString instead of the DataSource report property.
- In the Advanced view of Configuration Console, on Server Configuration Templates—Settings—Factory Service—e.Reports—Data Access using ODBC, select Enable ODBC connection pooling for Actuate ODBC drivers (Windows Only), as shown in Figure 14-3.

You can start connection pooling using one of the following properties:

- Enable ODBC connection pooling for non Actuate ODBC drivers—Starts ODBC connection pooling for ODBC drivers other than Actuate DataDirect drivers specified in the ConnectionString property. Default is false.

- Enable ODBC connection pooling for Actuate ODBC drivers (Windows Only) only)— Starts ODBC connection pooling for Actuate DataDirect drivers specified in the ConnectionString property on Windows. Default is true.

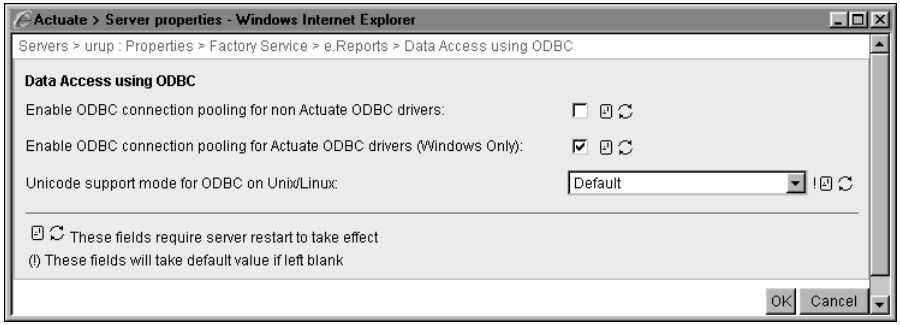

Figure 14-3 Starting ODBC connection pooling

Configuring BIRT JDBC connection pooling

In Server Configuration Templates➤Settings➤Factory Service and View Service, the administrator configures the BIRT JDBC connection pool by configuring the following properties:

- BIRT JDBC Connection Pool Size

  Caches concurrent connections, so the Factory can reuse them. The default is 10. Increase the pool size to establish more connections. To disable the connection pool, set BIRT JDBC Connection Pool Size to 0. The larger the pool size, the better performance in a highly concurrent system and the greater the memory consumption.

- BIRT JDBC Connection Pool Timeout

  Specifies the time-out interval for cached connections. The default is 3600 seconds. Increasing the available connections increases performance, but also increases memory consumption.

- BIRT Connection Validation Interval

  Controls the frequency of BIRT connection validation. Connection validation prevents a query from attempting to use a broken connection. The default value of -1 disables connection validation.

  The interval is expressed in seconds. If the value is positive, iServer first determines whether the last validation occurred in the specified interval.

  If not, iServer performs validation by calling java.sql.Connection.isValid( ). All DataDirect JDBC drivers support java.sql.Connection.isValid( ). Because some JDBC driver do not support java.sql.Connection.isValid( ), the administrator must restart iServer after rebooting the metadata database to refresh the JDBC connection pool even if the interval has not changed.

Figure 14-4 shows Server Configuration Templates➛Settings➛Factory Service➛BIRT JDBC Connection Pool.

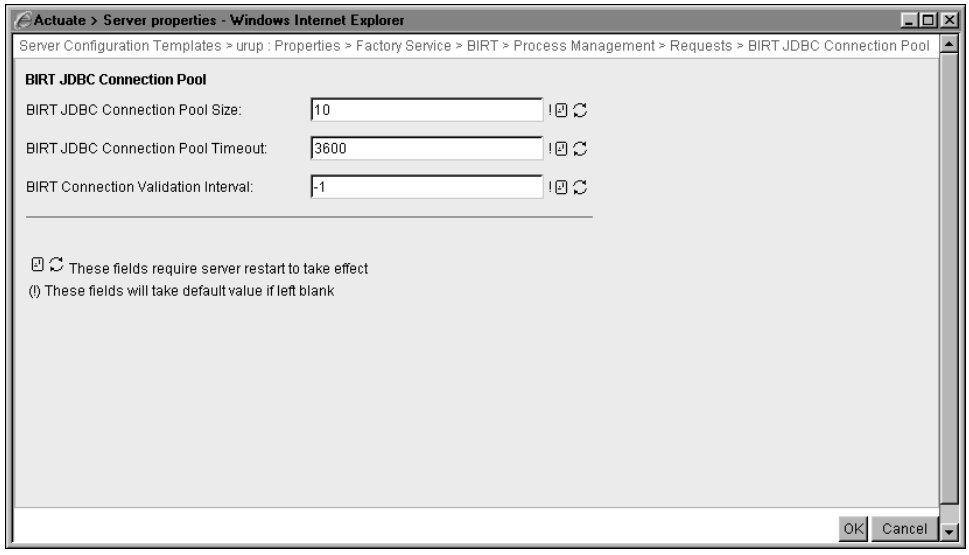

Figure 14-4 Specifying BIRT JDBC Connection Pool properties

How to configure BIRT JDBC Connection Pool properties

To configure BIRT JDBC Connection Pool properties, perform the following tasks:

1. Log in to Configuration Console, choose Advanced view, then choose Server Configuration Templates.
2. On Server Configuration Templates, choose a template.
3. On Settings, expand Factory Service or Viewing Service, then, expand BIRT, Process Management, and Requests, as shown in Figure 14-5 or Figure 14-6. Choose BIRT JDBC Connection Pool.
4. In BIRT JDBC Connection Pool Size, accept the default value, 10 connections. Alternatively, decrease or increase the pool size by typing a number not less than 0 or greater than 65535.
5. In BIRT JDBC Connection Pool Timeout, accept the default value, 3600 seconds. Alternatively, increase or decrease the time-out period by typing a different number.
6. In BIRT Connection Validation Interval, accept the default value, -1, to leave connection validation disabled. Alternatively, enable connection validation by specifying a period of time in seconds.

   Figure 14-4 shows BIRT JDBC Connection Pool set to default values.
7. Restart iServer after setting any of these values.

Figure 14-5 shows BIRT JDBC Connection Pool in Server Configuration Templates➛Settings➛Factory Service.

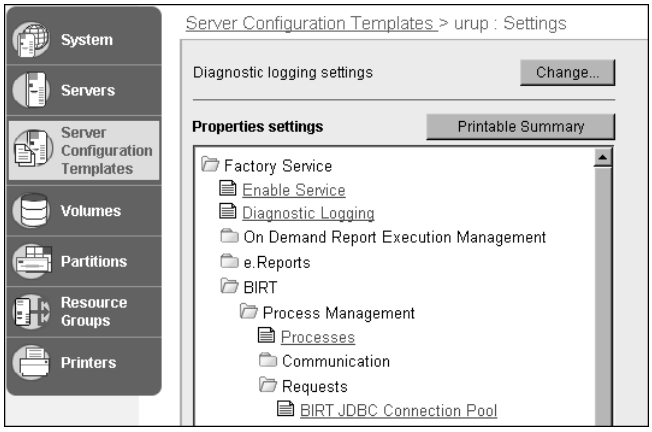

Figure 14-5 Choosing BIRT JDBC Connection Pool for Factory Service

Figure 14-6 shows BIRT JDBC Connection Pool in Server Configuration Templates➛Settings➛View Service.

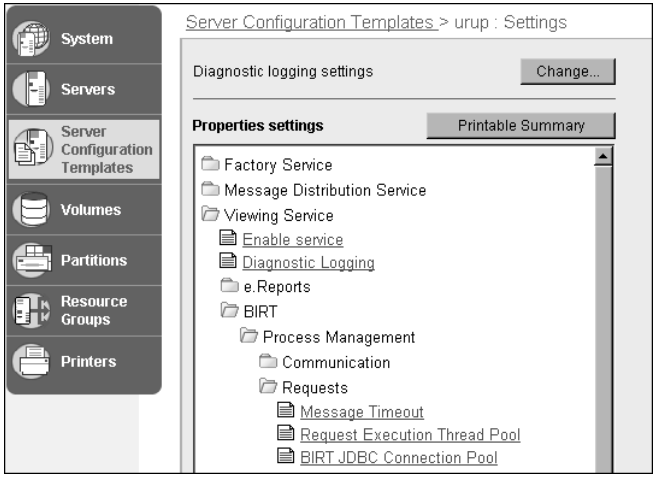

Figure 14-6 Choosing BIRT JDBC Connection Pool for View Service

Setting Encyclopedia Engine connection pool reap interval

The connection pool reap interval specifies the time, in seconds, between runs of the maintenance thread. When the maintenance thread runs, it discards any connections that remain unused for longer than the specified time.

The administrator sets the default connection pool reap interval by performing the following tasks:

1. Log in to Configuration Console and choose Advanced view.
2. Choose Server Configuration Templates. On Settings, expand the iServer folder. Choose Encyclopedia Engine.
3. In Default connection pool reap interval in seconds, accept the default value, 600, as shown in Figure 14-7. Alternatively, type a new value, from 0 to 86400. To disable the pool maintenance thread, set the interval to 0.

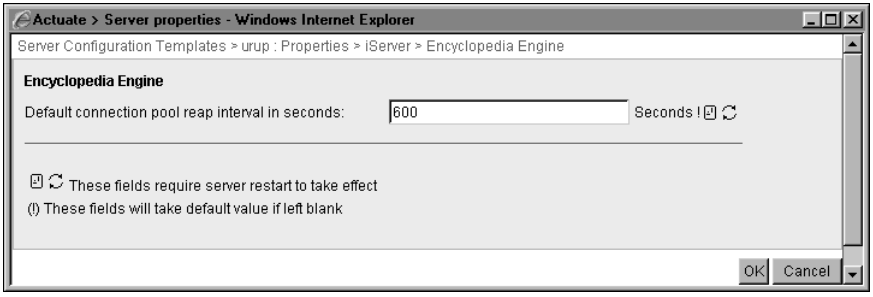

Figure 14-7 Specifying default connection reap interval

Chapter

15

Setting miscellaneous properties

This chapter contains the following topics:

- Changing locale, encoding, and time zone
- Modifying general volume properties
- Changing ports used by iServer
- Viewing and modifying general server properties
- Changing iServer system start-up parameters
- Configuring general system properties
- About General
- Setting start-up arguments for the Encyclopedia server JVM
- Starting and stopping iServer
- Setting miscellaneous server configuration template properties for iServer
- Configuring RSAPI sockets for RPC
- Setting Conversion Queue and E-mail Queue Resource Management properties
- Using Printable Summary to view system properties
- Using Printable Summary to view template properties
- Setting Filetype driver properties

Changing locale, encoding, and time zone

The language and regional settings of your computer operating system must be compatible with the default locale settings that you select. Incompatibilities can cause problems with the character set used for entries from your keyboard and the language used in the iServer graphical user interface.

You can specify a time zone:

- When you log in. This value overrides the settings in Options—General.
- On System—Properties—Regional Settings.
- On Options—General. This value sets the locale and time zone of the browser and workstation. The machine that runs the browser stores this setting locally.

Figure 15-1 shows the default regional settings for iServer or a cluster in System—Properties—Regional Settings.

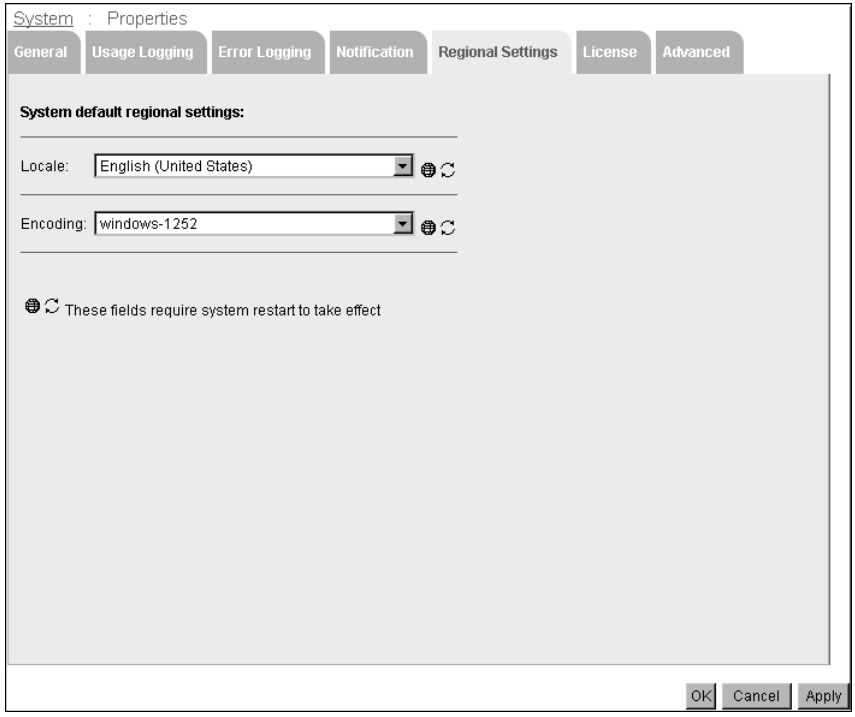

Figure 15-1 Configuring default regional settings

Figure 15-2 shows Options—General in the Advanced view of Configuration Console, where you specify the locale and time zone to use for iServer.

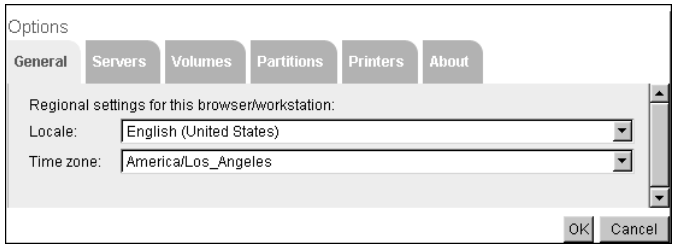

Figure 15-2 Specifying locale and time zone in Options—General

The following parameters appear in Options—General:

- Locale
  The user selects a locale during login. This locale appears in the Configuration and Management Consoles. If a user does not specify a value for this parameter during login, iServer reads the locale from the user's web browser cookie.
- Time zone
  The user selects a time zone during login. This time zone appears in the Configuration and Management Consoles. If a user does not specify a value for this parameter during login, iServer reads the time zone from the user's web browser cookie.

The TimeZones.xml file in the following directory stores information about time zones:

```
\Program Files\Actuate11SP4\iServer\servletcontainer\mgmtconsole
   \WEB-INF
```

TimeZones.xml stores time zones in the format that the tz database, or zoneinfo database prescribes.

The LANG environment variable on UNIX and Linux specifies the machine's language code.

Modifying general volume properties

You can view or modify the general property values for an Encyclopedia volume in Volumes—Properties—General using the following procedure.

How to modify general property values for an Encyclopedia volume

1. Log in to Configuration Console and choose Advanced view.
2. From the side menu, choose Volumes.
3. Choose the arrow next to the volume name, then choose Properties.

4 In Volumes—Properties—General, shown in Figure 15-3, you can modify the schedule for purging notices, purging deleted files, the volume archive service provider, and the partition for the e-mail notification template.

Choose OK.

The properties you can set in Volumes—Properties—General are the same whether you are modifying an existing volume or creating a new volume, with the exception that you can specify the volume name when creating a volume.

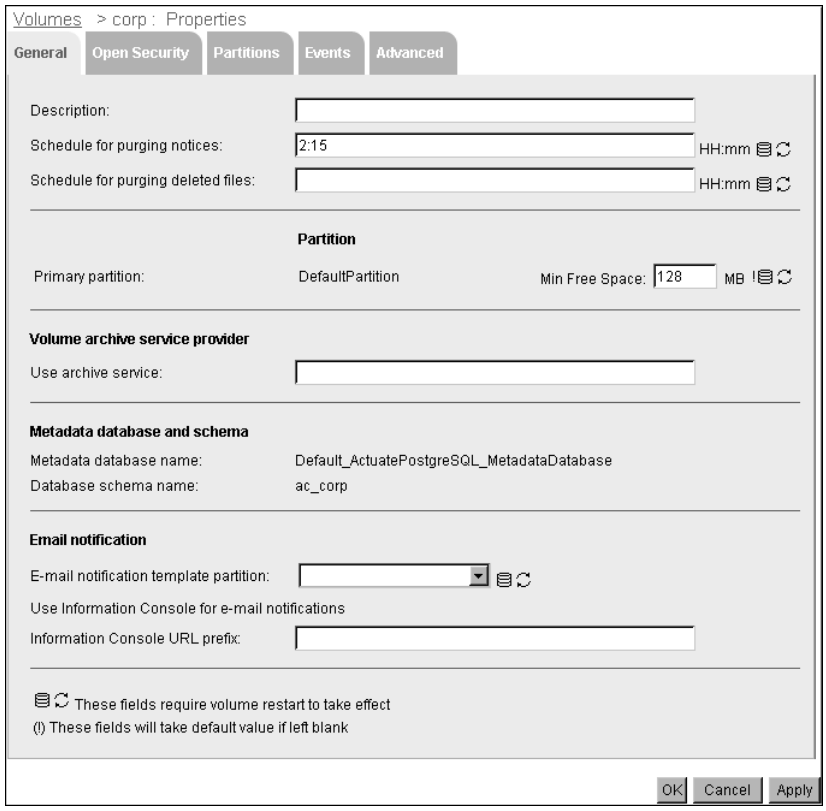

Figure 15-3 Modifying general property values for an Encyclopedia volume

Changing ports used by iServer

Table 15-1 lists a number of key ports. You can view the port numbers appearing on Servers—Properties—General. To change these port numbers, you must edit

acserverconfig.xml. You can view or change the port numbers that you access from Server Configuration Templates—Settings.

Table 15-1 Setting iServer ports

Port	Configuration Console location
Daemon listen port	Servers—Properties—General
The application container process listen port for the Management and Information Consoles	Servers—Properties—General
Server port	Servers—Properties—General
Server port base	Servers—Properties—General
Message distribution, which the Message Distribution service uses	Server Configuration Templates—Settings—Message Distribution Service—Process Management—Communication
Chart server port for generating charts in Actuate Basic reports	Server Configuration Templates—Settings—iServer—Chart Server for e.Reports
Base port for Java Factory server	Server Configuration Templates—Settings—Factory Service—BIRT—Process Management—Communication—Sockets
Base port for Java View server	Server Configuration Templates—Settings—Viewing Service—BIRT—Process Management—Sockets

Table 15-2 contains a list of iServer ports set during installation, default port numbers, and ranges of values.

Table 15-2 iServer Release 11 ports

Name	Display name	Default	Range	Changeable
AppContainerPort	Application container process listen port	8900	1 - 65535	Yes
CustomEvent ServicePort	Custom Event Service Port	8900	1 - 65535	Yes
NWPPort	Network process port for integration server queries	14000	1024 - 65535	Yes

(continues)

Table 15-2 iServer Release 11 ports (continued)

Name	Display name	Default	Range	Changeable
PMDPort	Daemon listen port	8100	None	No
ReportEngine HeartBeatPort	Port number for receiving Factory server heartbeat	11101	None	No
RSSESOAPPort	RSSE service port	8900	1 - 65535	Yes
ServerSOAPPort	Port number for iServer internal SOAP endpoint	11100	None	No
ServerSOAPPort Base	Base port number for iServer internal SOAP endpoint	13500	None	No
SMTPPort	Listen port	25	1 - 65535	Yes
SOAPDispatch SOAPPort	Message Distribution service port	8000	1 - 65535	Yes
SOAPPort	Port for caching server operations	11550	1024 - 65535	Yes
SOAPPort	Port for integration server operations	12100	1024 - 65535	Yes
SocketBaseForJava Processes	Base port for Java Factory server	21500	1025 - 65535	Yes
SocketBaseFor Processes	Base port number for processes	18500	1025 - 65535	Yes
SocketBaseForJava Processes	Base port for Java View server	21000	1025 - 65535	Yes

Viewing and modifying general server properties

Choose a machine name from the list of servers on Servers in the Advanced view of Configuration Console. In Servers—Properties—General, shown in Figure 15-4, you view general property values for a machine.

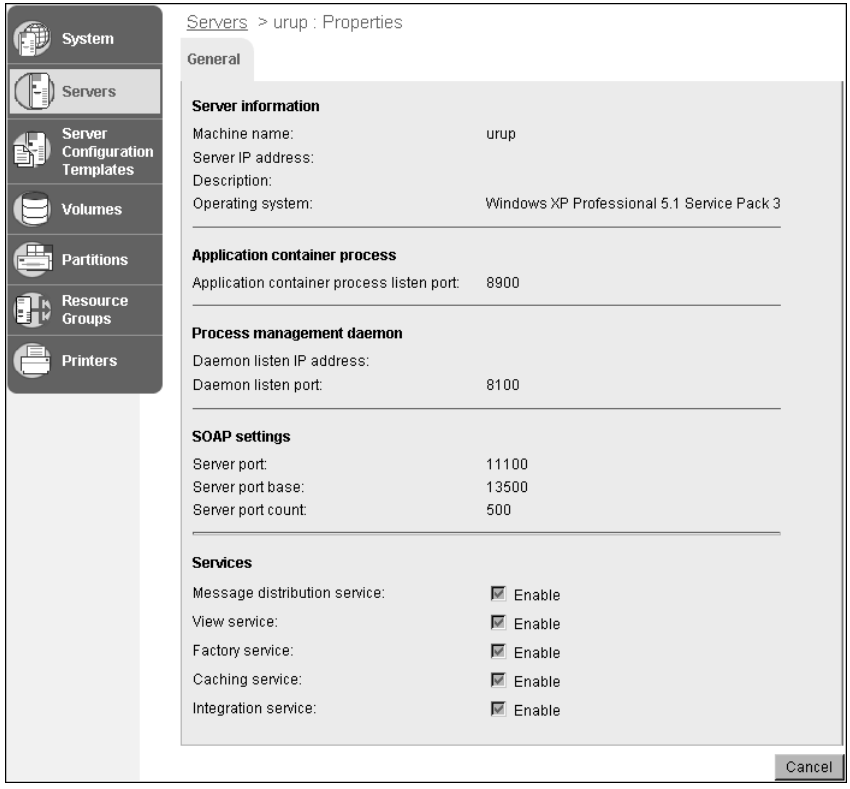

Figure 15-4 Viewing general properties for a server

The administrator can modify the following general server properties from Server Configuration Templates—Settings:

- Application container process listen port
- Server port
- Server port base
- Server port count

Services are enabled by default in the Template element in acserverconfig.xml.

The administrator modifies general server properties by expanding Server Configuration Templates—Settings—iServer—Process Management—Communication, and choosing the Application Container and Internal SOAP Endpoint options, as shown in Figure 15-5.

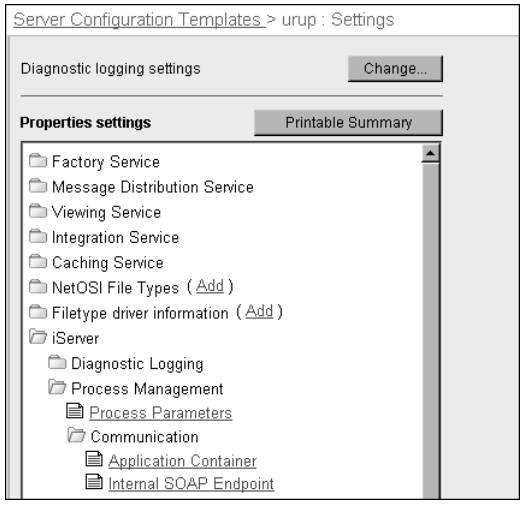

Figure 15-5 Choosing to view or modify general server properties

About application container process

This property specifies the application container port number. iServer uses an application container to host web services applications. Start and stop the application container process using the StartMC and StopMC scripts in AC_SERVER_HOME/bin.

How to set the application container process listen port

1. On Server Configuration Templates—Settings, expand iServer, Process Management, and Communication, then choose Application Container, as shown in Figure 15-5.
2. In Application container process listen port, accept the default, 8900, as shown in Figure 15-6. Alternatively, specify a different value.

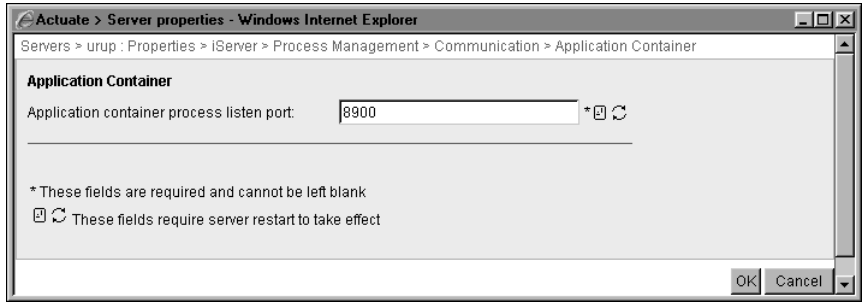

Figure 15-6 Viewing or changing application container process listen port

About SOAP settings

On Server Configuration Templates—Settings—iServer—Process Management—Communication—Internal SOAP Endpoint, the administrator can specify the following properties:

- Port number for iServer internal SOAP endpoint
  Web service API (IDAPI) port and internal server port for communication with other nodes in a cluster. iServer uses this port to receive information, such as heartbeat messages.
- Base port number for iServer internal SOAP endpoint
  Beginning of a range of port numbers that iServer attempts to use for internal SOAP messages. Used with server port count.
- Port range (from base) for iServer internal SOAP endpoint
  The range of port numbers of ports used for exchanging internal SOAP messages starting from the server port base.
- Port number for iServer Encyclopedia engine internal SOAP endpoint

How to set the SOAP settings properties

1. On Server Configuration Templates—Settings—iServer—Process Management—Communication, choose Internal SOAP Endpoint, as shown in Figure 15-5.
2. On Internal SOAP Endpoint, as shown in Figure 15-7, perform the following tasks:
   1. In Port number for iServer internal SOAP endpoint, accept the default, 11100. Alternatively, specify a different value.
   2. In Base port number for iServer internal SOAP endpoint, accept the default, 13500. Alternatively, specify a different value.
   3. In Port range (from base) for iServer internal SOAP endpoint, accept the default, 500. Alternatively, specify a different value.
   4. In Port number for iServer encyclopedia engine internal SOAP endpoint, accept the default, 14100. Alternatively, specify a different value.
   5. In Port number for iServer encyclopedia engine internal SOAP endpoint for servicing iServer components requests, accept the default, 14200. Alternatively, specify a different value.
   6. In Java encyc server purge thread limit, accept the default, 2. Alternatively, specify a different value.

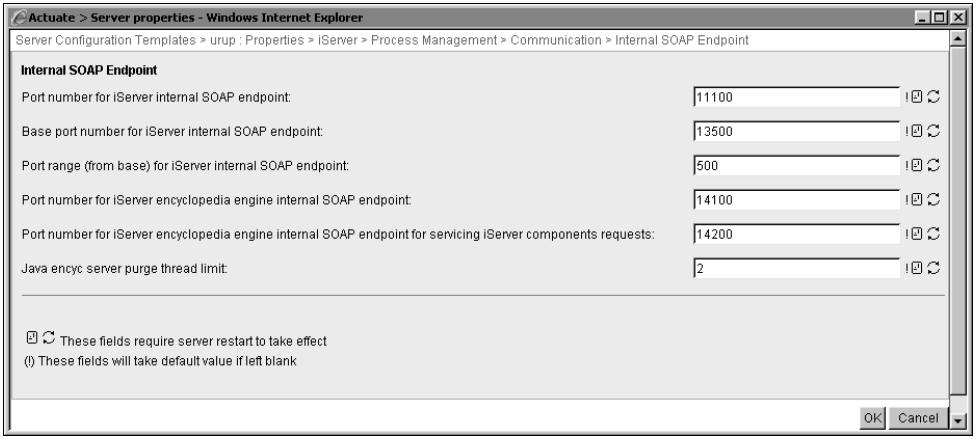

Figure 15-7 Specifying SOAP settings

3 If you change any property values, restart iServer.

Changing iServer system start-up parameters

Before starting the iServer system, you can specify an alternate configuration file as well as an alternate template for this iServer node to use. When you start the iServer system, iServer will use these resources.

How to modify start-up parameters for iServer

1 Put iServer offline by choosing Stop on System—Status, as shown in Figure 15-8.

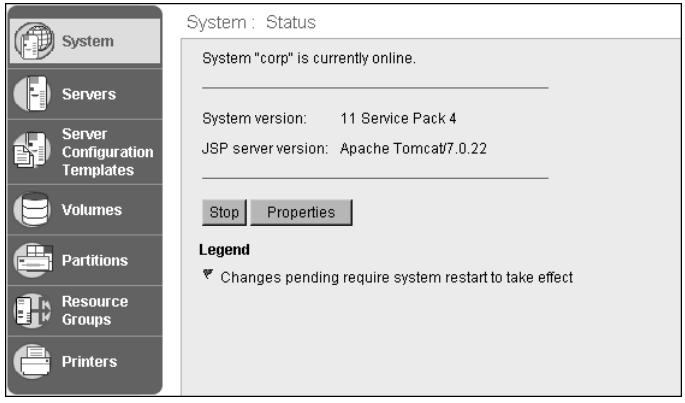

Figure 15-8 Putting iServer offline

2 On System—Status, choose Modify Start Parameters, as shown in Figure 15-9.

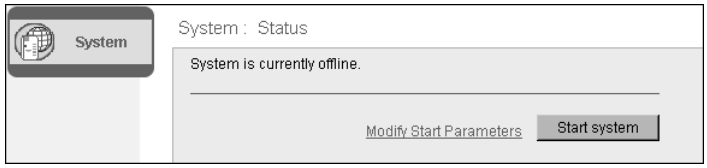

Figure 15-9 Choosing Modify Start Parameters

3 On System—Modify Start Parameters, set start-up parameters by performing the following tasks:

1 In Configuration home, type the path of the folder containing an alternate acserverconfig.xml. This folder must also contain acserverlicense.xml. By default, the location of acserverlicense.xml is AC_DATA_HOME/config/11SP4.

2 In Template name, type the name of the template that you want iServer to use, as shown in Figure 15-10.

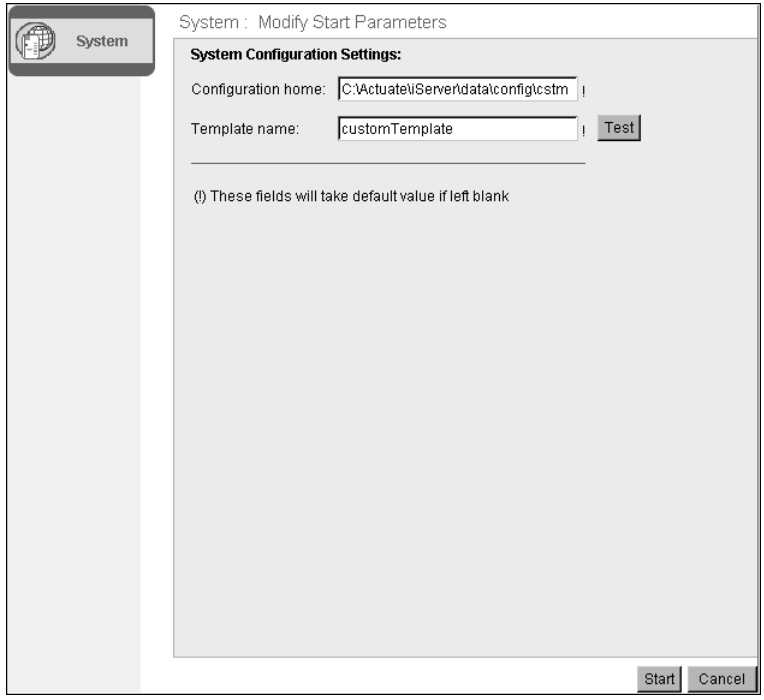

Figure 15-10 System—Modify Start Parameters

3 Choose Test to verify that iServer recognizes the parameters. A message box appears, confirming that the test is successful. Choose OK, as shown in Figure 15-11.

Figure 15-11 Confirming that the parameters are valid

4 On System—Modify Start Parameters, choose Start.

Configuring general system properties

In System—Properties—General, the administrator can view or change the following general system properties, as shown in Figure 15-12:

- System name of the cluster node or stand-alone iServer
- The password for logging in to Configuration Console

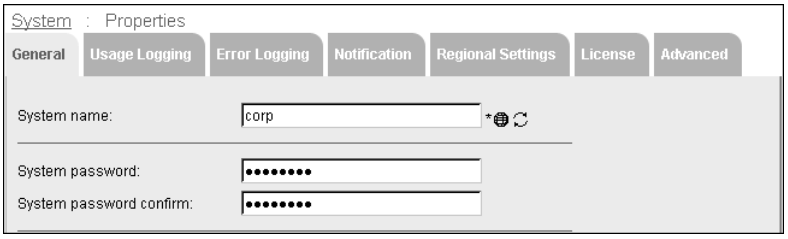

Figure 15-12 Changing general system properties

If you change the System name, restart iServer.

About General

In System—Properties—Advanced, the administrator can choose General to view or set values for the following properties:

- Cluster schema
  Name of the iServer system schema
- Maximum number of user properties cache entries
  Tunable property for holding a user's expanded role information in a cache
- Enable Multi-system mode
  Supports joining multiple clusters together

- Startup arguments for the cluster JVM
  Arguments passed to the JVM that communicates with the system schema

How to set general properties

1 On System—Properties—Advanced, choose General, as shown in Figure 15-13.

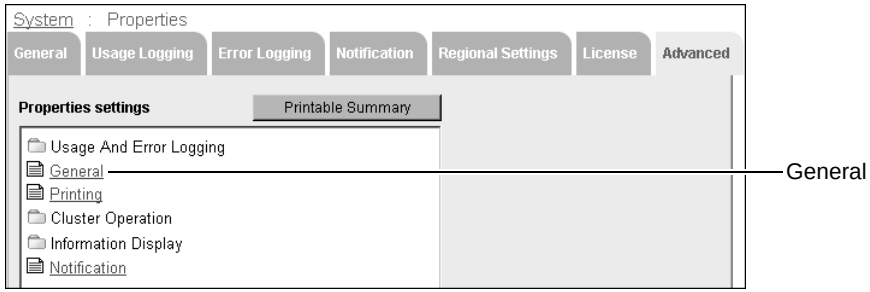

Figure 15-13 Viewing System—Properties—Advanced

2 On General, perform the following tasks:

1. Accept the default values for Cluster Schema, as shown on Figure 15-14. Alternatively, specify a different schema.
2. In Maximum number of user properties cache entries, accept the default value, 500, or alternatively, specify a different value.
3. For Enable Multi-system mode, accept the default value of unchecked, or alternatively, select the property.
4. In Startup arguments for the cluster JVM, accept the default arguments. Alternatively, modify these arguments.

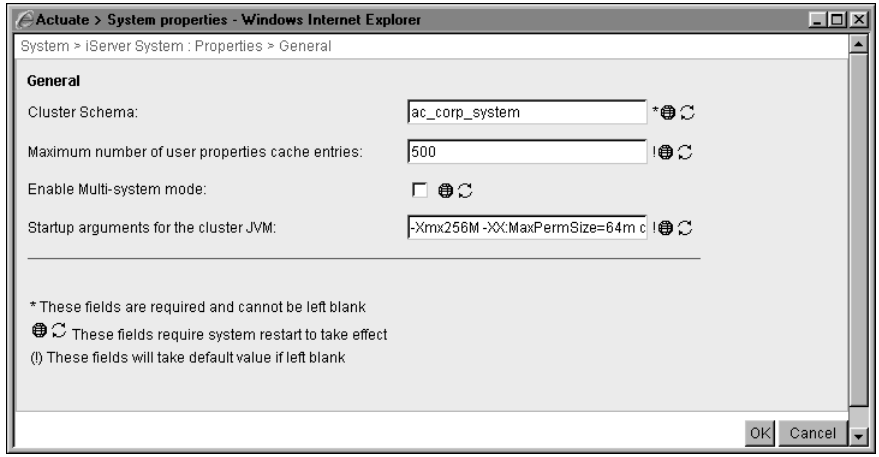

Figure 15-14 Configuring General

Setting start-up arguments for the Encyclopedia server JVM

In Server Configuration Templates—Settings—iServer, the administrator can choose to view or edit the start-up arguments for the Encyclopedia server JVM.

How to set server configuration properties

1 Expand iServer and Process Management, then choose Process Parameters, as shown in Figure 15-15.

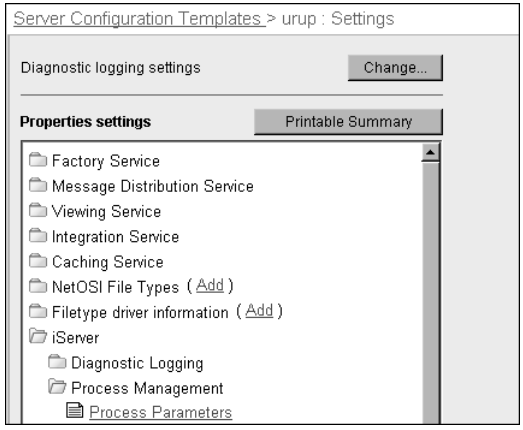

Figure 15-15 Choosing Process Management properties

2 In Start parameters for iServer encyclopedia engine, as shown in Figure 15-16, accept the default parameters. Alternatively, modify these parameters.

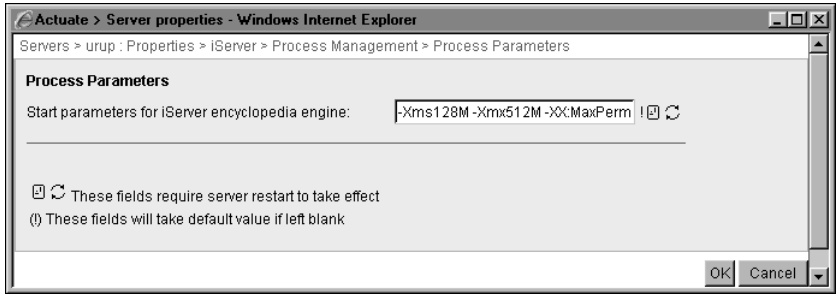

Figure 15-16 Viewing or setting start parameters for the Encyclopedia engine

If you change the property, restart iServer.

Starting and stopping iServer

System—Status displays iServer status and provides controls for starting and stopping iServer, cluster-level management, and cluster creation, as shown in Figure 15-17.

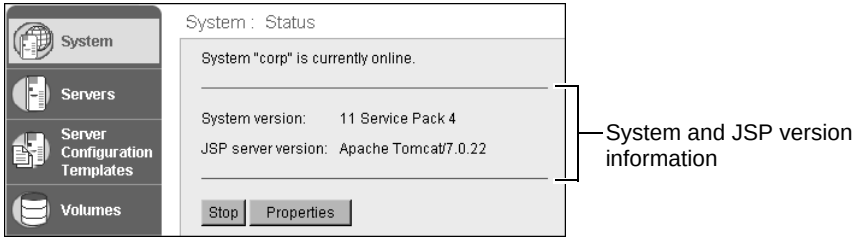

Figure 15-17 Viewing System—Status

System—Status displays the following information about iServer:

- iServer name and its current status
- Release version of the iServer
- Name and version of the application server

For example, in Figure 15-17, System—Status indicates that a machine is online. Actuate Release 11 is running.

Earlier in this book you learned about the possible iServer states that appear in the Simple view of Configuration Console. The same states can appear in System—Status. Two of these states are online and offline.

Stopping iServer

When iServer is online, you can access the following functionality in System—Status:

- Choose Stop to stop iServer. Choosing Stop for a node stops this node only.
- Choose Properties to view System—Properties and modify the following property values, which are described later:
  - General properties
  - System usage and error logging
  - E-mail notification
  - Default regional settings
  - License information

Starting iServer

When iServer is offline, you can start iServer from System—Status by choosing Start system, as shown in Figure 15-18.

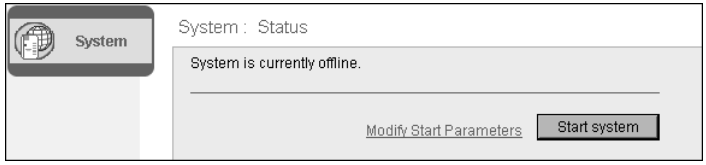

Figure 15-18 Starting the system

Setting miscellaneous server configuration template properties for iServer

The administrator can set several miscellaneous iServer properties appearing in Server Configuration Template—Settings—iServer, as shown in Figure 15-19.

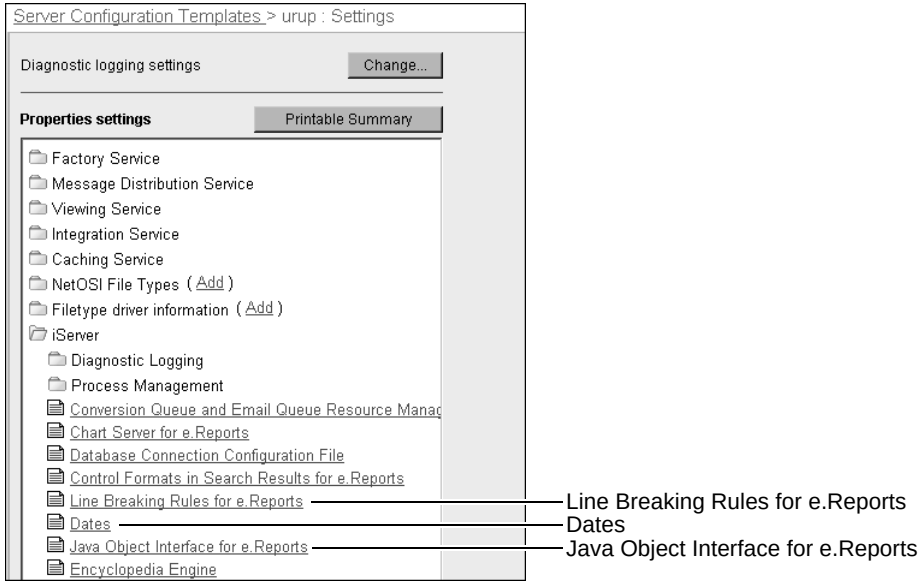

Figure 15-19 Viewing iServer properties

Configuring the line breaking rule

In Server Configuration Templates—Settings—iServer, the administrator can choose Line Breaking Rules for e.Reports to specify property values that control text wrapping in Actuate Basic documents.

How to set line breaking rules for Actuate Basic documents

1 Choose iServer—Line Breaking Rules for e.Reports, as shown in Figure 15-19.

2 Specify values for the properties appearing on Line Breaking Rules for e.Reports, as shown in Figure 15-20. Alternatively, leave these properties blank. Table 15-3 describes these properties.

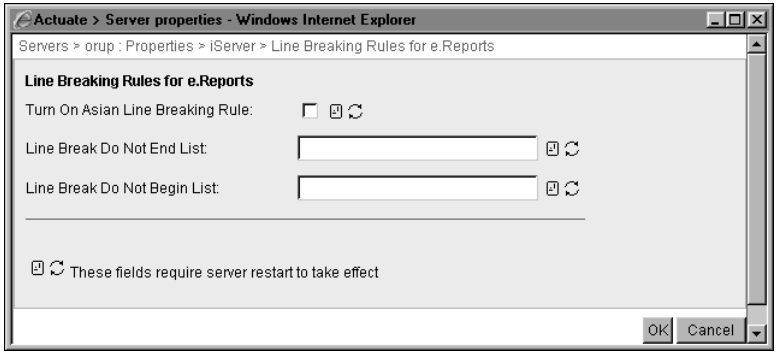

Figure 15-20 Specifying iServer text line breaking property value

Table 15-3 iServer text line breaking properties

Property	Description
Turn On Asian Line Breaking Rule	Supports controlling text wrapping in Actuate Basic text controls. iServer uses a list of characters that do not begin or end a line of text. If set to true, iServer also uses characters listed in DoNotBegin and DoNotEnd. iServer does not start a line with characters listed in DoNotBegin and does not end a line with characters listed in DoNotEnd. The default value is false. Do not use the additional characters specified in DoNotBegin and DoNotEnd. Set the value to true and specify additional characters if line wrapping occurs at incorrect locations in the text. For example, specify characters if the text wraps at the wrong location for punctuation when displaying text that uses double-byte characters.
Line Break Do Not End List	Controls text wrapping in Actuate Basic Text controls. If the property TurnOnAsianLineBreakingRule is true, iServer does not end a line with characters listed in DoNotEnd. Specify multibyte characters in hexadecimal format. The following example lists three double-byte characters: \x3001\x3002\xff0c

(continues)

Table 15-3 iServer text line breaking properties (continued)

Property	Description
Line Break Do Not Begin List	Controls text wrapping in Actuate Basic text controls. If the property TurnOnAsianLineBreakingRule is true, iServer does not start a line with characters listed in DoNotBegin. Specify multibyte characters in hexadecimal format. The following example lists three double-byte characters: \x3001\x3002\xff0c

Configuring dates

In Server Configuration Templates—Settings—iServer, the administrator can choose Dates, to specify how iServer formats dates.

How to set date formatting for iServer

1. Choose iServer—Dates, as shown in Figure 15-19.
2. Specify values for the properties appearing on Dates, as shown in Figure 15-21. Alternatively, accept the default values. Table 15-4 describes these properties.

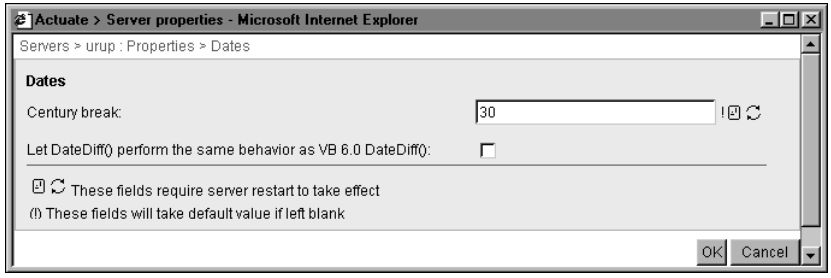

Figure 15-21 Specifying iServer date property values

Table 15-4 Date properties

Property	Description
Century break	Indicator of how iServer converts 2-digit year values into 4-digit years. Using the default value 30, iServer converts a date using a 2-digit year in the following manner: ■ A value less than 30 becomes 20xx. ■ A value of 30 and greater becomes 19xx. For example, the date 3-5-19 becomes 3-5-2019, and 3-5-57 converts to 3-5-1957.

Table 15-4 Date properties

Property	Description
Let DateDiff( ) perform the same behavior as VB 6.0 DateDiff()	Controls the DateDiff( ) function in Actuate Basic reports. Use the functionality based on Microsoft Visual Basic .Net, or on Microsoft Visual Basic 6.0. Set the value to true if you view or run Actuate Basic reports built with Actuate Release 7 or earlier software that use DateDiff( ) with a ww or www date part indicator.

Configuring the Java Object Interface

In Server Configuration Templates—Settings—iServer, the administrator can choose Java Object Interface for e.Reports, to configure the maximum heap size of the Java Virtual Machine (JVM) when loading the JVM into a Factory Server or View Server process if Java object access is needed.

How to configure Java Object Interface for e.Reports

1. Choose iServer—Java Object Interface for e.Reports, as shown in Figure 15-19.
2. To improve JVM performance and prevent running out of memory when the Java object needs more memory, set Maximum heap size for Java virtual machine to a larger maximum heap size. Alternatively, accept the default value of 0, as shown in Figure 15-22. Changing Maximum heap size for Java virtual machine to a larger value causes Factory Server and View Server processes to use more memory.

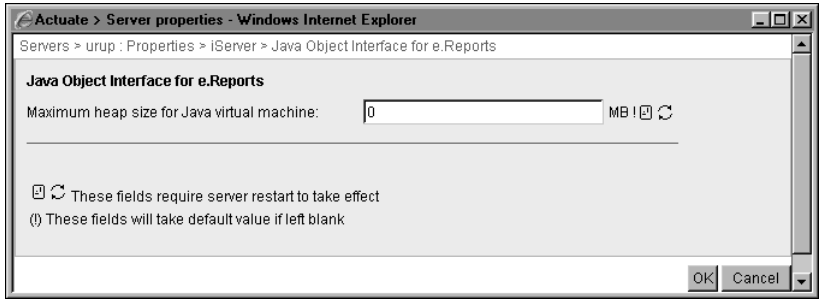

Figure 15-22 Configuring the Java Object Interface

Configuring RSAPI sockets for RPC

In Server Configuration Templates—Settings—iServer, the administrator can choose to specify socket information for use with RSAPI.

How to configure RSAPI sockets for RPC

1 Expand iServer, Process Management, Communication, and RSAPI, then choose Sockets, as shown in Figure 15-23.

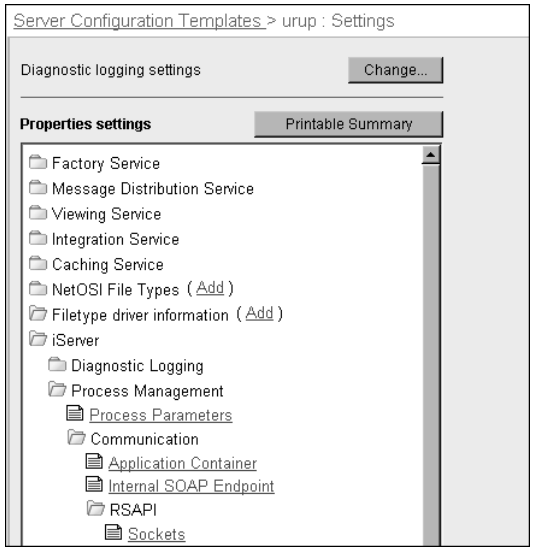

Figure 15-23 Choosing Sockets

2 Specify socket information for use with RSAPI by setting the property values on Sockets, as shown in Figure 15-24. Alternatively, accept the default values. Table 15-5 describes these properties.

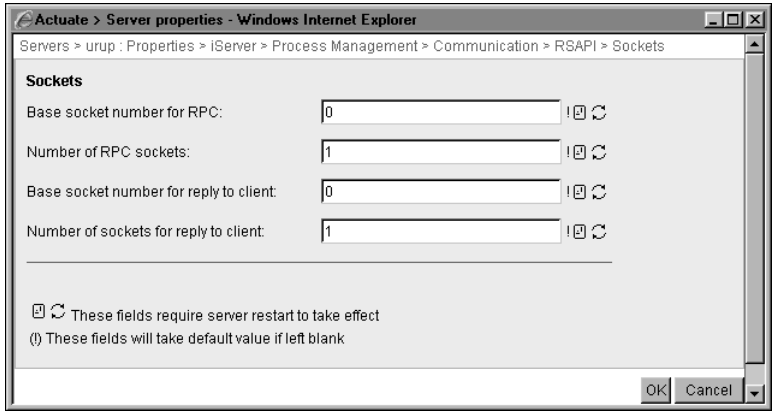

Figure 15-24 Specifying iServer RSAPI socket information

Table 15-5 iServer RSAPI socket properties

Property	Description
Base socket number for RPC	Beginning of a range of port numbers that iServer attempts to use for the request server RPC ports. Required for RSAPI. iServer starts at the BASE port and attempts to use up to BASE + COUNT - 1 port to find a single port.
Number of RPC sockets	Range of port numbers that iServer attempts to use for the request server RPC ports. Required for RSAPI. If you do not specify the beginning of the range in the Base socket number for RPC property value, iServer ignores the range setting. If you specify the beginning of the range, but you do not specify the range, iServer uses the default range of 1.
Base socket number for reply to client	Beginning of a range of port numbers for communication between iServer and clients. Required for RSAPI.
Number of sockets for reply to client	The range of port numbers for ports used for communication between iServer and clients. Required for RSAPI. If you do not specify the beginning of the range in the Base socket number for reply to client property value, iServer ignores the range setting. If you specify the beginning of the range, but you do not specify the range, iServer uses the default range of 1.

Table 15-6 lists the property names that appear in Configuration Console with the corresponding parameter names in AC_SERVER_HOME/etc /acmetadescription.xml, indicating default settings, ranges, and when a property change takes effect.

Table 15-6 RSAPI RPC communication parameters

Property name	Parameter name	Default	Range	Takes effect
Base socket number for RPC	BaseSocketNumForRPC	0	0 - 32767	Server Restart
Number of RPC sockets	NumSocketsForRPC	1	0 - 32767	Server Restart
Base socket number for reply to client	BaseSocketNumForReply	0	0 - 32767	Server Restart

Setting Conversion Queue and E-mail Queue Resource Management properties

Conversion Queue and E-mail Queue Resource Management controls thread and queue length allocations for the following processes:

- Conversion Queue
  - Number of conversion threads

    The number of threads available for e.Report document conversion tasks, such as conversion to PDF. Specify a number between 1 and 50 inclusive.

    Actuate recommends setting this number to a value that is equivalent to the number of view service processes plus 2. The number of view service processes is the value specified in Maximum number of processes in Server Configuration Templates➤Properties➤Viewing Service ➤e.Reports➤ Process Management ➤Processes, as shown in Figure 15-25.

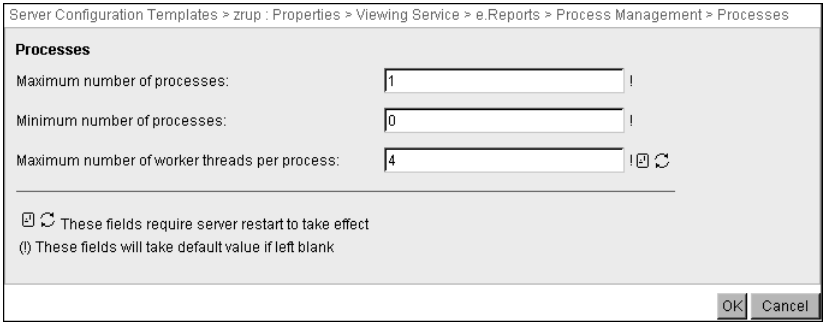

Figure 15-25 Specifying maximum number of View Services processes

  - Max conversion queue length

    The maximum number of document conversion tasks in the queue. Specify a number between 1000 and 1000000 inclusive.

- E-mail Queue
  - Number of e-mail notification threads

    The number of threads available for e-mail notification tasks. Specify a number between 1 and 50 inclusive.

  - Max e-mail notification queue length

    The maximum number of e-mail notification tasks allowed to wait for execution. Specify a number between 1000 and 1000000 inclusive.

Restart iServer after setting any of these values.

How to configure Conversion Queue and E-mail Queue Resource Management properties

To configure Conversion Queue and E-mail Queue Resource Management properties, perform the following tasks:

1. Log in to Configuration Console, choose Advanced view, then choose Server Configuration Templates.
2. On Server Configuration Templates, choose a template.
3. On Settings, expand iServer folder. Choose Conversion Queue and E-mail Queue Resource Management, shown in Figure 15-26.

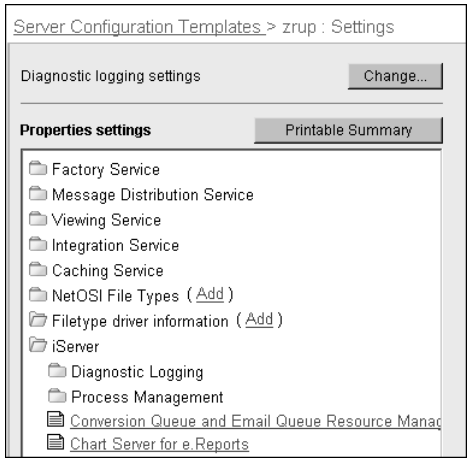

Figure 15-26 Choosing Conversion Queue and E-mail Queue Resource Management properties

4. In Number of conversion threads, accept the default value, 1. Alternatively, increase the number of threads available for document conversion by typing a number not greater than 50.
5. In Max conversion queue length, accept the default value, 10000. Alternatively, decrease or increase the queue length by typing a number not less than 1000 or greater than 1000000.
6. In Number of e-mail notification threads, accept the default value, 1. Alternatively, increase the number of threads available for e-mail notification by typing a number not greater than 50.
7. In Max e-mail notification queue length, accept the default value, 10000. Alternatively, decrease or increase the queue length by typing a number not less than 1000 or greater than 1000000.

   Figure 15-27 shows Conversion Queue and E-mail Queue Resource Management set to default values.

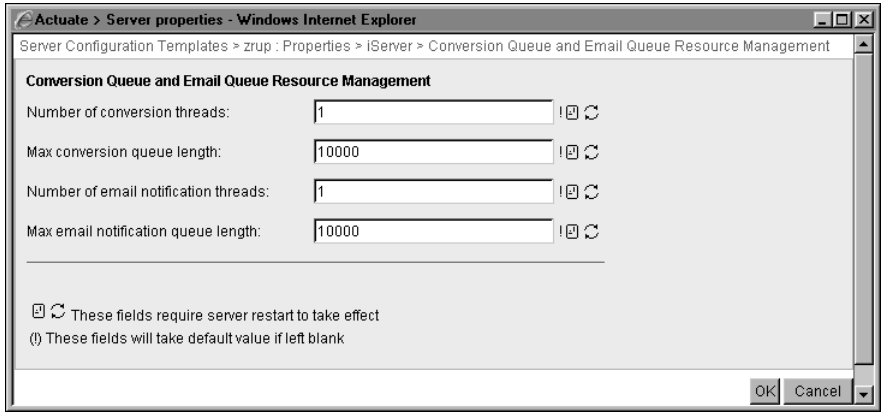

Figure 15-27 Specifying Conversion Queue and E-mail Queue Resource Management properties

8 Restart iServer after setting any of these values.

Using Printable Summary to view system properties

System➤Properties➤Advanced➤Printable Summary provides an HTML-formatted document containing a text summary of system property settings for an administrator to view or print. This summary contains the settings for the following system property groups:

- System, including General, Locale, Printing, Cluster Operations, Information Display, and Notification
- Usage and Error Logging

How to use Printable Summary to view system properties

The administrator can view this summary of system property settings by performing the following tasks:

1 Log in to Configuration Console, choose Advanced view, then choose System.

2 On System, choose Properties.

3 On Properties, choose Advanced.

4 Choose Printable Summary

An HTML-formatted document appears, displaying a detailed list of system properties, as shown in Figure 15-28.

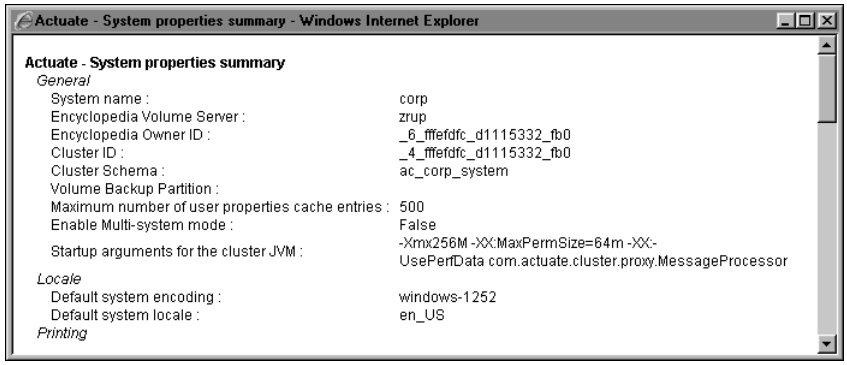

Figure 15-28 Viewing the system properties summary

Using Printable Summary to view template properties

Server Configuration Templates➤Settings➤Printable Summary provides an HTML-formatted document containing a text summary of template property settings for an administrator to view or print. This summary contains the settings for the following template property groups:

- Template, including Diagnostic Logging, Process Management, Queue Resource Management, and other miscellaneous properties
- View, factory, message distribution, caching, and integration services

How to use Printable Summary to view template properties

The administrator can view this summary of template property settings by performing the following tasks:

1. Log in to Configuration Console, choose Advanced view, then choose Server Configuration Templates.
2. On Server Configuration Templates, choose a template.
3. On Settings, choose Printable Summary.

   An HTML-formatted document appears, displaying a detailed list of the template and view, factory, message distribution, caching, and integration service properties, as shown in Figure 15-29.

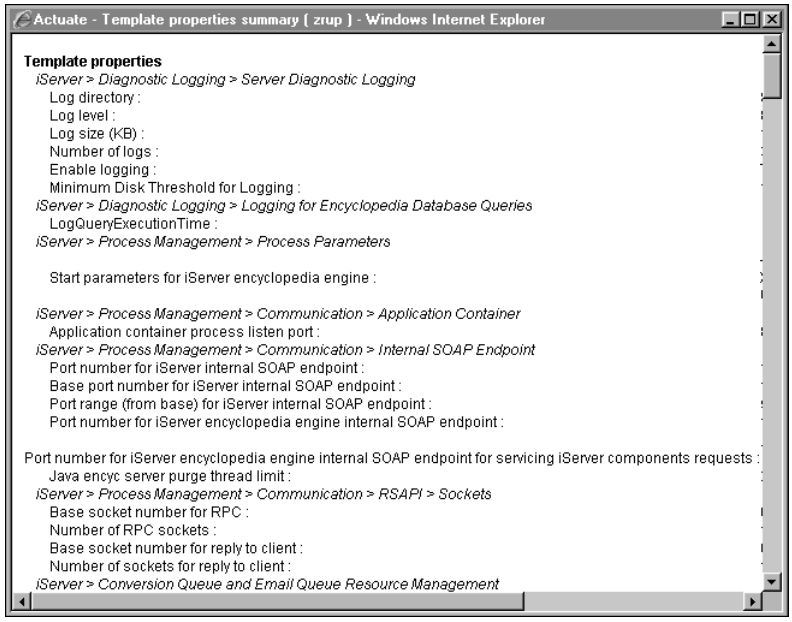

Figure 15-29 Viewing the template properties summary

Setting Filetype driver properties

The administrator specifies the following properties for a file-type driver:

- Name

  Open Systems Interconnection (OSI) file-type driver name.

- OSI driver path

  Specifies the network link used to access the OSI driver.

- OSI parameter file location

  Specifies the location of the OSI driver parameter file used to configure network settings.

How to set Filetype driver properties

To set Filetype driver properties, perform the following tasks:

1. Log in to Configuration Console, choose Advanced view, then choose Server Configuration Templates.
2. On Server Configuration Templates, choose a template.

3 On Settings, next to Filetype driver information, choose Add, as shown in Figure 15-30.

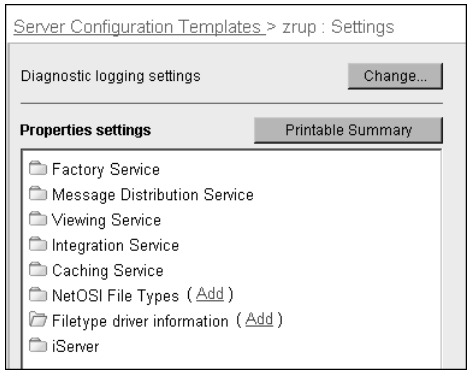

Figure 15-30 Choosing to add File type driver information

Figure 15-31 shows Add.

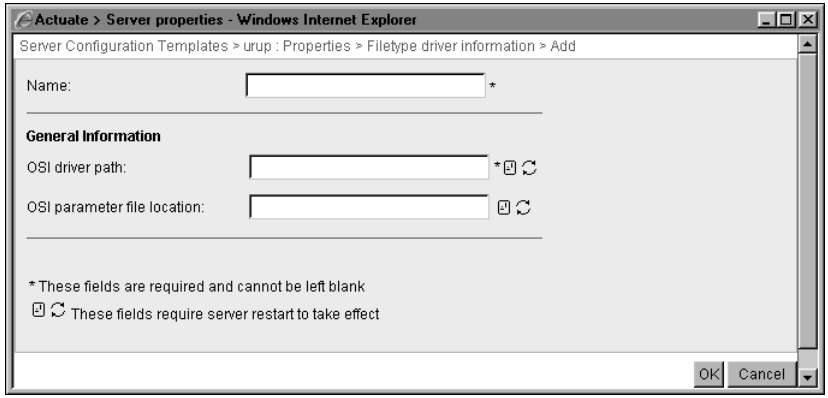

Figure 15-31 Specifying File type driver information

4 In Name, type the name of the OSI file-type driver.

5 In OSI driver path, type the full path to the OSI driver.

In OSI parameter file location, type the full path to the OSI driver parameter file.

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y