

# Actuate e.Reporting Suite 4



e.Reporting Server Guide  
Release 4.1

Information in this document is subject to change without notice. Examples provided are fictitious. No part of this document may be reproduced or transmitted in any form, or by any means, electronic or mechanical, for any purpose, in whole or in part, without the express written permission of Actuate Corporation.

© 1995 - 2000 by Actuate Corporation. All rights reserved. Printed in the United States of America.

Contains information proprietary to:

Actuate Corporation  
701 Gateway Boulevard  
South San Francisco, CA 94080  
<http://www.actuate.com>

The software described in this manual is provided by Actuate Corporation under an Actuate License agreement. The software may be used only in accordance with the terms of the agreement.

Actuate Corporation trademarks and registered trademarks:  
Actuate, e.Report, e.Reporting, Live Report Document, Live Report Extension, ReportBlast, ReportCast, Report Encyclopedia, SmartSearch, Transporter, Virtual Report Distribution, and XML Reports.

Third party trademarks:

ActiveX, Microsoft, MSN, The Microsoft Network, Windows, Windows NT, Windows 95/98 and/or other Microsoft products are either trademarks or registered trademarks of Microsoft. CT-Library is a trademark of Sybase, Inc. Digital Creations and DCLC, are trademarks of Digital Creations, LC. Graphics Server is a trademark of Bits Per Second, Ltd. and Pinnacle Publishing, Inc. HDK is a registered trademark of Virtual Media Technology Pty Ltd. INFORMIX-ESQL/C is a trademark of Informix Software, Inc. InstallShield is a registered trademark of InstallShield Corporation. LEADTOOLS is a registered trademark of LEAD Technologies, Inc. Netscape, Netscape Communications, Netscape Communicator, Netscape Enterprise Server, Netscape FastTrack Server, and Netscape Navigator are either trademarks or registered trademarks of Netscape Communications Corporation. NobleNet and WinRPC are trademarks of NobleNet, Inc. ONC, Solaris, SPARC, Sun, and Sun Microsystems are trademarks of Sun Microsystems. Oracle Call Interface is a trademark of Oracle Corporation. Progress is a registered trademark of Progress Software Corporation. Sheridan Calendar Widgets is a trademark of Sheridan Software Systems, Inc. SmartHeap is a trademark of MicroQuill Software Publishing, Inc. Tools.h++ is a trademark of Rogue Wave Software, Inc. UNIX is a registered trademark of X/Open Company, Ltd. XPrinter is a trademark of Bristol Technology, Inc. XPAT, created by James Clark, is licenced under the Mozilla licence agreement. WinWidgets is a trademark of Simple Software, Inc.

All other brand or product names are trademarks or registered trademarks of their respective companies or organizations.

Document No. 000501-2-430301 April 28, 2000

# Contents

<b>Introduction</b> .....	<b>vii</b>
About Actuate e.Reporting Suite 4. ....	vii
About <i>Actuate e.Reporting Server Guide</i> .....	x
About Actuate e.Reporting Server product .....	xi
Online documentation. ....	xii
Using online manuals .....	xiii
Online help .....	xiii
Using context-sensitive online help .....	xiii
Using the Actuate online help system. ....	xiii
Using report-specific online help .....	xvi
Typographical conventions .....	xvii
Syntax conventions .....	xviii
<b>Chapter 1</b>	
<b>Overview of Actuate e.Reporting Server.</b> .....	<b>1</b>
Introducing the e.Reporting Server .....	2
Report server features .....	3
Report server architecture .....	4
Introducing the Report Encyclopedia .....	7
Performing administration tasks .....	8
Administrator Desktop tasks .....	8
Report server tasks .....	9
Upgrading the Report Encyclopedia .....	9
<b>Chapter 2</b>	
<b>Report server administration on UNIX</b> .....	<b>11</b>
Setting up the report server account .....	12
Maintaining security .....	12
Accessing resources. ....	12
Controlling UNIX file descriptors. ....	13
Specifying a configuration file. ....	14
Specifying an image search path .....	14
Specifying graph image quality .....	14
Starting the report server on reboot. ....	14
Automatic disk space checking .....	15
Setting up mail notification .....	15
Printing reports containing OLE objects. ....	16
Adding Postscript printer models .....	16

Using PostScript and PCL fonts . . . . .	16
Specifying and mapping fonts . . . . .	18
Font entry examples . . . . .	18
Indicating the report server's home directory. . . . .	19
Report server temporary files . . . . .	19
Displaying the report server version . . . . .	19
Adjusting Factory process timing . . . . .	20
Setting Factory process variables . . . . .	20
Setting Factory report generation status interval . . . . .	21
Adjusting system process restarts . . . . .	21
Adjusting report server startup . . . . .	21
Adjusting report server automatic restarts . . . . .	21
Adjusting object aging start times . . . . .	22
Setting View process variables . . . . .	22
Open server configuration . . . . .	25
Adjusting scheduled request removal . . . . .	25
Scheduled request removal environment variable . . . . .	26
Configuring locale-specific settings . . . . .	26
Language environment variables . . . . .	27
Configuring year conversion for Actuate . . . . .	28
View process locale configuration . . . . .	29
Configuring report server port numbers . . . . .	31
RPC and client communication port configuration . . . . .	31
View process socket port configuration. . . . .	32
Starting and stopping a report server . . . . .	33
The UNIX shutdown_srvr script . . . . .	34
Clearing items and queues . . . . .	35
Backing up and restoring the Report Encyclopedia . . . . .	36
Setting up mail notification . . . . .	38

## Chapter 3

### **Report server administration on Windows NT and Windows 2000. . . . . 39**

Setting up the report server account . . . . .	40
Maintaining security . . . . .	40
Accessing resources . . . . .	41
Specifying a configuration file . . . . .	41
Specifying an image search path . . . . .	42
Automatic disk space checking . . . . .	42
Configuring Factory process default locale . . . . .	43
Printing reports containing OLE objects . . . . .	43

Finding the report server's home directory . . . . .	.44
Report server temporary files . . . . .	.44
Displaying the report server version. . . . .	.44
Adjusting Factory process timing . . . . .	.45
Setting Factory timing Registry entries. . . . .	.45
Setting Factory report generation status interval . . . . .	.45
Adjusting system process starts and restarts . . . . .	.46
Adjusting report server startup . . . . .	.46
Adjusting automatic restarts. . . . .	.47
Adjusting object aging start times . . . . .	.47
Setting View process parameters. . . . .	.48
Open server configuration . . . . .	.50
Open server configuration for Scribe. . . . .	.51
Printing Scribe open server reports . . . . .	.51
Open server configuration for Crystal Reports . . . . .	.51
Printing Crystal Reports open server reports. . . . .	.52
Adjusting scheduled request removal . . . . .	.52
Scheduled request removal registry entry . . . . .	.52
Configuring year conversion for Actuate . . . . .	.53
View process locale configuration . . . . .	.54
Configuring report server port numbers . . . . .	.56
RPC and client communication port configuration . . . . .	.56
View process socket port configuration . . . . .	.58
Starting and stopping a report server . . . . .	.59
Configuring report server processes on Windows NT and Windows 2000 . . . . .	.59
Clearing items and queues . . . . .	.60
Backing up and restoring the Report Encyclopedia . . . . .	.62
Setting up mail notifications . . . . .	.65
RSMAIL.EXE parameters . . . . .	.67
Working with the Exchange Server and domains . . . . .	.68

## Chapter 4

<b>Database connections . . . . .</b>	<b>71</b>
About database connections . . . . .	.72
Stored procedures . . . . .	.72
Defining environment variables . . . . .	.73
Connecting to DB2 databases. . . . .	.73
DB2 environment variables . . . . .	.74
HP-UX shared objects for DB2 clients . . . . .	.74
Sun shared objects for DB2 clients . . . . .	.74
Checking a connection to a DB2 instance . . . . .	.74
Installing Factory server software for DB2. . . . .	.75

DB2 version 5 fixpac . . . . .	75
DB2 version 6.1 connection issues. . . . .	75
Connecting to Informix databases . . . . .	76
Informix environment variables . . . . .	76
Informix database environment . . . . .	77
Informix account information . . . . .	77
Using UNIX Factory server software with Informix . . . . .	77
Determining Informix version on UNIX . . . . .	77
Accessing Informix libraries . . . . .	77
Overriding Informix DLLPath . . . . .	78
Determining Informix version on NT . . . . .	78
Setting the maximum column length . . . . .	78
Connecting to MS/SQL databases . . . . .	79
MS/SQL database server. . . . .	79
MS/SQL account information . . . . .	79
Accessing MS/SQL libraries. . . . .	79
Testing the MS/SQL connection. . . . .	80
Driver versions . . . . .	80
Db-library limitations. . . . .	80
7.0 Compatibility with 6.5 . . . . .	80
Connecting to databases through ODBC . . . . .	81
Setting the maximum column length . . . . .	82
Using the PeopleSoft ODBC driver . . . . .	82
Using Red Brick ODBC drivers . . . . .	83
Windows NT Red Brick configuration. . . . .	84
UNIX Red Brick configuration . . . . .	84
Checking the connection to Red Brick databases. . . . .	85
Connecting to Oracle databases . . . . .	85
Oracle java stored procedures . . . . .	85
Installing Actuate libraries for Oracle clients . . . . .	85
Using Actuate clients and Oracle clients . . . . .	86
Using the Actuate e.Reporting Server and Oracle clients . . . . .	86
Oracle connection strings. . . . .	87
SQL*Net version 1 . . . . .	87
SQL*Net version 2 . . . . .	87
Oracle environment variables . . . . .	87
Oracle listener processes . . . . .	88
Testing the Oracle connection . . . . .	88
Oracle 8 datatype support . . . . .	89
Setting the maximum column length . . . . .	89
Connecting to Progress 8 databases . . . . .	89
Actuate software . . . . .	90
Setting the maximum column length . . . . .	90

Progress software . . . . .	90
Preparing the Progress Client environment . . . . .	90
Preparing the environment on Windows 95/98 or NT machines . . . . .	90
Preparing the environment on UNIX machines . . . . .	91
Progress self-service client connection . . . . .	92
Starting a Progress 8 database and the Progress Open Interface Broker . . . . .	92
Connecting to a Progress 8 database from a Developer Workbench report . . . . .	93
Connecting to Progress 9.1 databases . . . . .	95
Actuate software . . . . .	96
Progress software . . . . .	96
Windows environment settings . . . . .	96
Actuate connection properties. . . . .	97
Progress 9.1 SQL92 connections. . . . .	97
Stored procedures and triggers . . . . .	98
Using the AcProgressConnection connection . . . . .	98
UNIX platforms . . . . .	99
SunOS. . . . .	99
HP-UX . . . . .	99
AIX . . . . .	100
Connecting to Sybase databases . . . . .	100
Sybase environment variables. . . . .	100
Sybase database server. . . . .	100
Sybase account information . . . . .	101
Accessing Sybase dynamic link libraries. . . . .	101
Testing the Sybase connection. . . . .	101
UNIX factsrvr database compatibility . . . . .	101

## Chapter 5

### **Report Encyclopedia utilities . . . . . 103**

Overview of the utilities and examples . . . . .	104
About the utilities. . . . .	104
Utility programs . . . . .	105
About the examples . . . . .	105
ACDEFRAG utility . . . . .	106
ACEXPORT utility . . . . .	107
ACIMPORT utility . . . . .	111
ACPIMPRT utility . . . . .	115
ACTOC utility . . . . .	124
UNIX PostScript font utility . . . . .	124
Installing a PostScript font. . . . .	125
Uninstalling a PostScript font . . . . .	125
PostScript font conversion issues . . . . .	125

About the archive driver. . . . .	126
Installing the example archive driver . . . . .	126
How the example archive driver works . . . . .	127
Logging example archive messages . . . . .	127
About the Report Server Security Extension . . . . .	127
Installing an RSSE application. . . . .	128
Using an MT-safe RSSE application. . . . .	128
About the example RSSE application . . . . .	129
Using the example RSSE application . . . . .	129
About USERSIDS.TXT file . . . . .	130
Logging example RSSE application messages . . . . .	130
<b>Index . . . . .</b>	<b>131</b>

---

## About Actuate e.Reporting Suite 4

e.Business customers use Actuate® e.Reporting Suite 4 to develop and deploy high resolution structured content to tens of thousands of users. Actuate takes web reporting to the next level by providing options for various needs as varied as seamless personalized web pages and traditional online and printed reports.

Customers building e.Business sites face challenges where Actuate e.Reporting Suite 4 offers the following solutions.

---

<b>Challenge</b>	<b>Actuate solution</b>
Developing custom content using HTML	DHTML provides a fast, no download option
Using plug-ins to view structured content	No need to distribute and install plug-ins for tens of thousands of users
Compromising information display due to lack of integrated tools	Provides template based design and display
Exploding use of web-based content delivery	Ability to support a million hits per day per CPU
Demand for personalized secure content	Page security for tens of thousands of users
Reuse of existing content	Open server provides access to content from other applications

<b>Challenge</b>	<b>Actuate solution</b>
Maintaining data integrity on hard copy	PDF provides high resolution printed copy
Portability of electronic data or content to other applications	XML output provides access to data across applications

Actuate tools and reports:

- Solve complex data access problems
- Solve formatting problems that go beyond the scope of other tools
- Scale to support thousands of users

The following summary describes the products in the Actuate e.Reporting Suite 4.

<b>Product name</b>	<b>Use</b>
Actuate Developer Workbench	An object-oriented application used by professional developers of structured content to design, build, and distribute report object designs and components throughout the enterprise. The Actuate Basic Language and Actuate Foundation Class Library support extensive customization capabilities.
Actuate e.Report Designer	An application that complements the Developer Workbench and is used by business users to design and distribute a variety of reports. These reports require no programming. This application supports both modification of complex reports and using sophisticated components from libraries.
Actuate End User Desktop	An application used by end users to request, generate, view, and print live report documents on networked client systems. The ReportQuery™ capabilities enable seamless transfer of data from an Actuate report to any productivity tool or analysis tool.
Actuate Viewer	Application for end users to find, view, and print report documents. The ReportQuery™ capabilities are also part of the Actuate Viewer.

Product name	Use
Actuate e.Reporting Server	<p>A server application that generates Live Report Documents, manages them in the Report Encyclopedia®, and makes them available to users.</p> <p>This product includes with Actuate Administrator Desktop, an application for system and network administrators to manage and control one or more Actuate report servers.</p> <p>This product also includes Actuate ReportCast™ that transforms the Report Encyclopedia into a dynamic, secure web site. ReportCast provides the foundation for Channels and seamless integration with other web sites.</p> <p>This product includes Actuate Live Reports Extension (LRX) that works with both Microsoft Internet Explorer and Netscape Navigator to support report viewing and printing.</p>
Actuate Advanced e.Reporting Server	<p>An application that adds two capabilities to the basic e.Reporting Server, open server and page security.</p> <ul style="list-style-type: none"> <li data-bbox="768 930 1290 1017">■ Open server supports the use of third-party report generators with the Actuate e.Reporting Server.</li> <li data-bbox="768 1025 1290 1107">■ Page security supports personalized viewing of parts of reports for different users</li> </ul>

Product name	Use
Actuate Software Development Kit	<p data-bbox="705 210 1198 296">Actuate ActiveX Controls embed Actuate reporting functionality into custom applications.</p> <p data-bbox="705 305 1233 510">Actuate Requester API accesses attributes and values of report parameters, changes the values of report parameters, controls how and when a report is generated, displays and prints reports, and configures report print setup. Access the Requester API using Actuate Basic, Visual Basic, C, or C++.</p> <p data-bbox="705 520 1183 635">Actuate search extension API supports developing search extensions to transfer data to any third-party productivity or analysis tool.</p> <p data-bbox="705 645 1210 878">Actuate report server API implements common Report Encyclopedia tasks, integrates report server features into existing corporate applications, automates routine or time-consuming tasks, and implements new feature groupings for custom business processes. Access the report server API using C++.</p> <p data-bbox="705 888 1198 975">Actuate Report Server Security Extension supports the use of third-party security tools.</p> <p data-bbox="705 986 1210 1065">Actuate archive driver supports the use of third-party archiving software and hardware.</p>

Actuate Viewer and Actuate Live Report Extension (LRX) are included with all products except Actuate Software Development Kit.

---

## About *Actuate e.Reporting Server Guide*

*Actuate e.Reporting Server Guide* provides information about managing the interaction between client and server to enable activities such as scheduling and running reports.

*Actuate e.Reporting Server Guide* includes the following chapters:

- *Introduction*. This chapter provides an overview of this guide, the Actuate e.Reporting Suite, and the typographical conventions used.

- *Chapter 1. Overview of Actuate e.Reporting Server.* This chapter introduces server architecture, the Report Encyclopedia, the Administrator Desktop, and the Navigator.
- *Chapter 2. Report server administration on UNIX.* This chapter contains discussions of starting and stopping a report server, clearing items and queues, and backing up and restoring the Report Encyclopedia.
- *Chapter 3. Report server administration on Windows NT and Windows 2000.* This chapter contains discussions of starting and stopping a report server, clearing items and queues, and backing up and restoring the Report Encyclopedia.
- *Chapter 4. Database connections.* This chapter contains information about connecting to Oracle, Sybase, MS/SQL, Informix, Progress, and DB2 databases. The chapter also contains information about connecting to databases using ODBC.
- *Chapter 5. Report Encyclopedia utilities.* This chapter contains information about Actuate command-line utilities to let you easily import and export all or part of the Report Encyclopedia to let you transfer data between Encyclopedias.

---

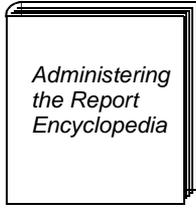
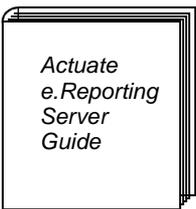
## About Actuate e.Reporting Server product

Actuate e.Reporting Server documentation includes a printed manual, an installation card, online help, user documentation in PDF files and in HTML format, and release notes. Information about the product that could not be included before the book printing deadline is in the release notes.

The Actuate web site, <http://www.actuate.com>, contains late-breaking news about the product and its features, as well as product update information. To obtain the password necessary to access the portion of the web site available only to customers, telephone Actuate Customer Support. The engineers in Actuate Customer Support can also help you with technical questions about the product according to your service contract. The Customer Support telephone number, fax number, and e-mail information can be found among the printed materials in the product box.

The printed documentation includes the following manuals.

<b>For information about</b>	<b>See the following</b>
Installation	Installation card
Late-breaking information about the software and documentation	Release notes

For information about	See the following
<p>Users, groups, privileges, and roles</p> <p>Printers and printing requests</p> <p>Process groups</p> <p>Building and managing your first Report Encyclopedia</p>	 <p><i>Administering the Report Encyclopedia</i></p>
<p>Viewing, running, finding, and printing reports</p>	 <p><i>Using e.Reports</i></p>
<p>Overview of the Actuate report server architecture</p> <p>Report server administration</p> <p>Database connections</p> <p>Report Encyclopedia utilities</p>	 <p><i>Actuate e.Reporting Server Guide</i></p>
<p>Actuate ReportCast, its templates, URL directives, scripting language, security and user authentication facilities.</p> <p>Actuate Live Report Extension (LRX)</p>	 <p><i>Building an e.Reporting Web Site</i></p>

## Online documentation

The information in the printed manuals is also available as online books in Adobe Acrobat PDF format and in the online help system for the Actuate products. For products without a Windows interface such as the Actuate e.Reporting Server and the Actuate ReportCast, we provide HTML help files.

## Using online manuals

The online manuals do not install automatically with the product. On the product CD, you find those files in the Manuals directory. Open the introductory PDF file to get an overview of the manuals. Copy this file and the files for each book you want to be able to use online to your local drive. The items in the table of contents and the page numbers in the index both contain links to the appropriate topics in the text. In the index, you access the link by positioning your cursor over the page number, not the topic.

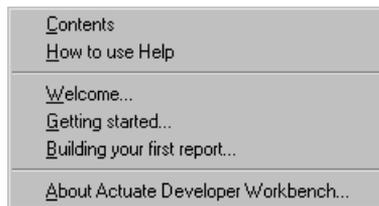
The HTML files install automatically with the product. These files can be viewed with standard browsers.

## Online help

Actuate products provide both context-sensitive online help about the product and report-specific online help about the report you are viewing. The Actuate e.Reporting Suite makes it possible for developers to create customized report-specific online help. Report-specific online help is only available when you view a report using an Actuate desktop product, such as Actuate End User Desktop, Actuate Viewer, or one of the Actuate LRXs.

## Using context-sensitive online help

Sections from the printed manuals have been linked directly to the software interface to make relevant information available while you work. Dialog boxes that need additional explanation about how to use them have a help button. To access online help, use the Help menu in the menu bar.



An alternative method is to use the F1 key.

## Using the Actuate online help system

Use two windows to access and view information in the e.Reporting Suite help system. The window on the right displays the contents of the online help

topics. The window on the left displays the table of contents or the index of the online help system.

Use the Sort button to view an alphabetical listing of help topics

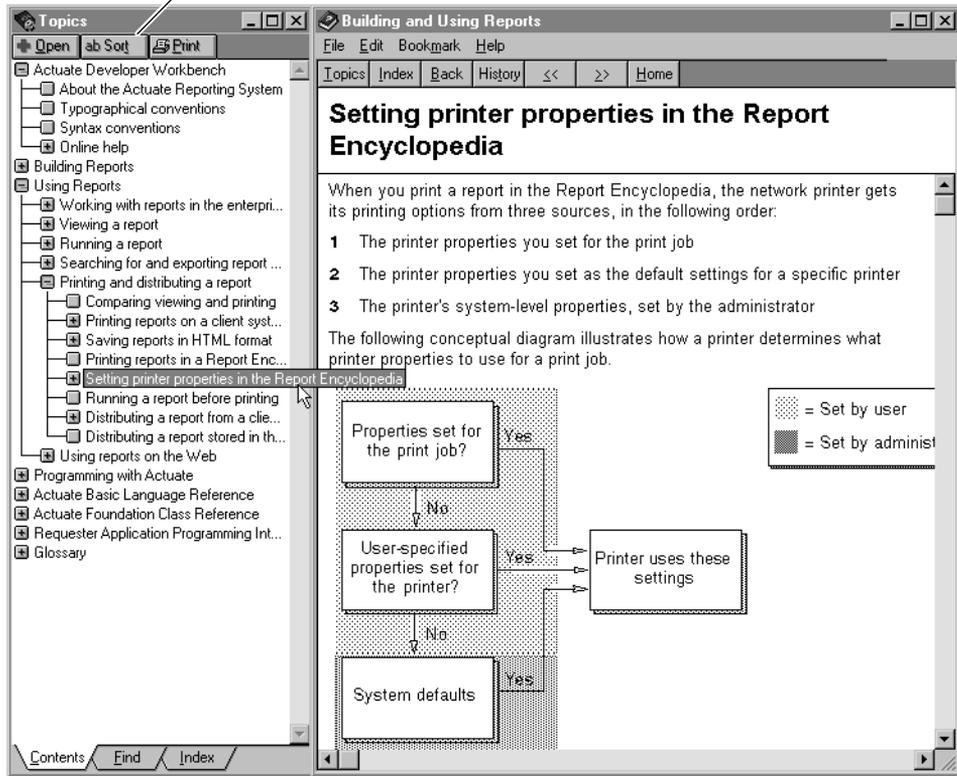


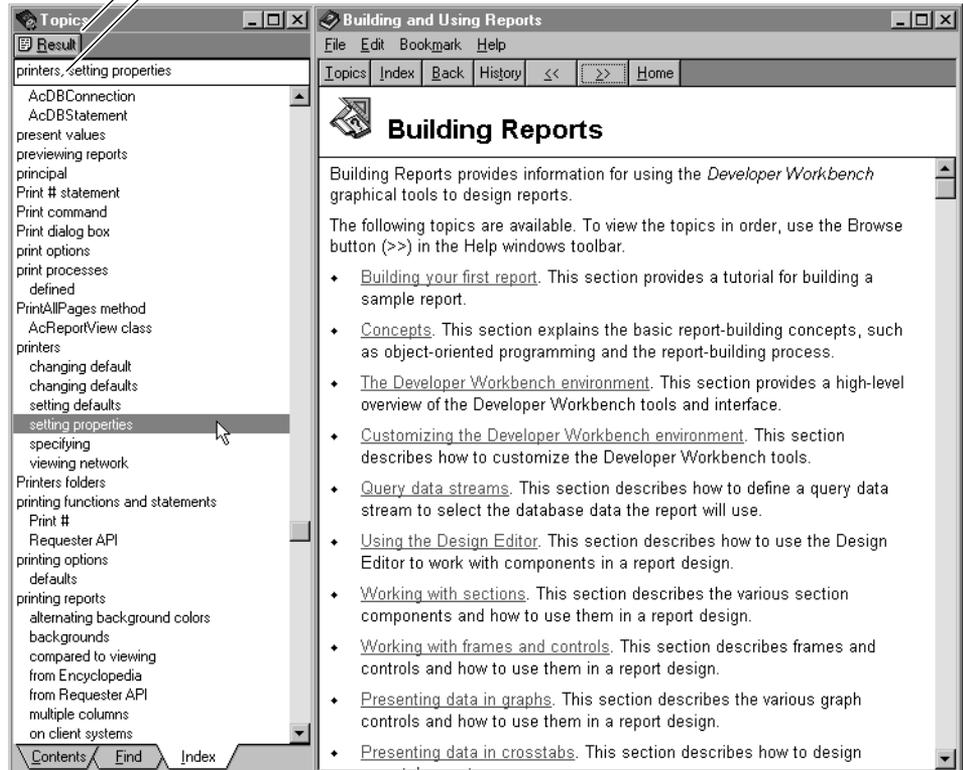
Table of Contents, Find, and Index tabs

The tabs at the bottom of the left window access different views. Use these tabs to switch views among the Table of Contents, Find, and Index. The buttons at the top of the window change according to the tab you select at the bottom.

The following illustrations show an example of the Index and the result of an Index search.

Use the Results button to view the topics

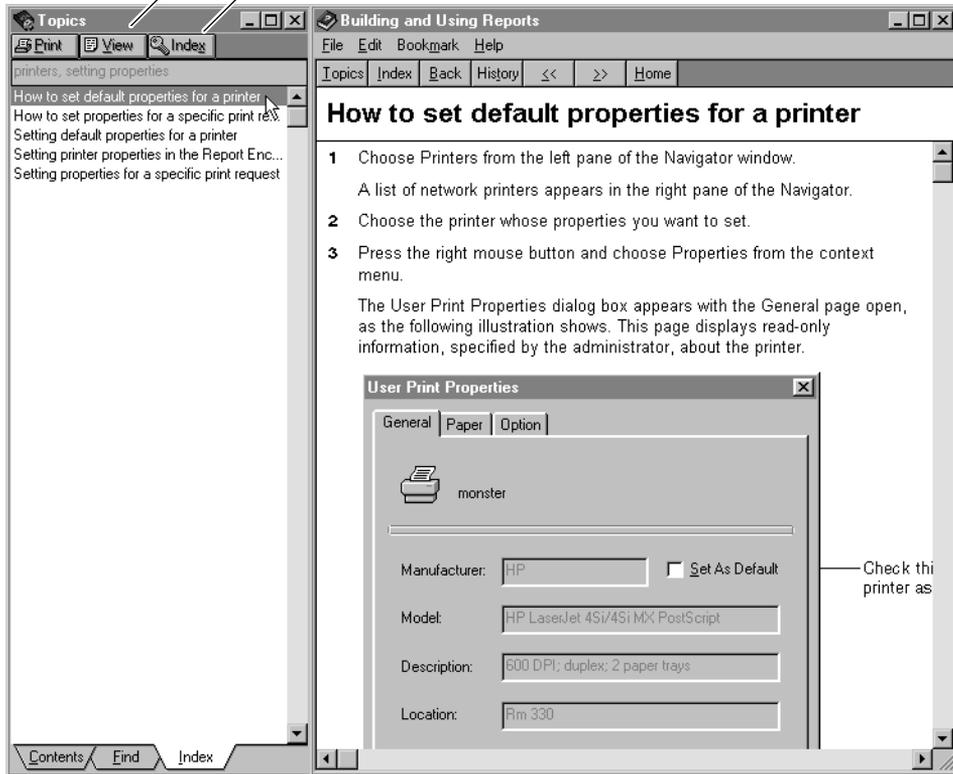
Enter the keyword to search the Index



The following illustration shows the result of the search as it appears in the window on the left. To view the topic in the right window double-click the topic from the list. The topic displays in the window on the right.

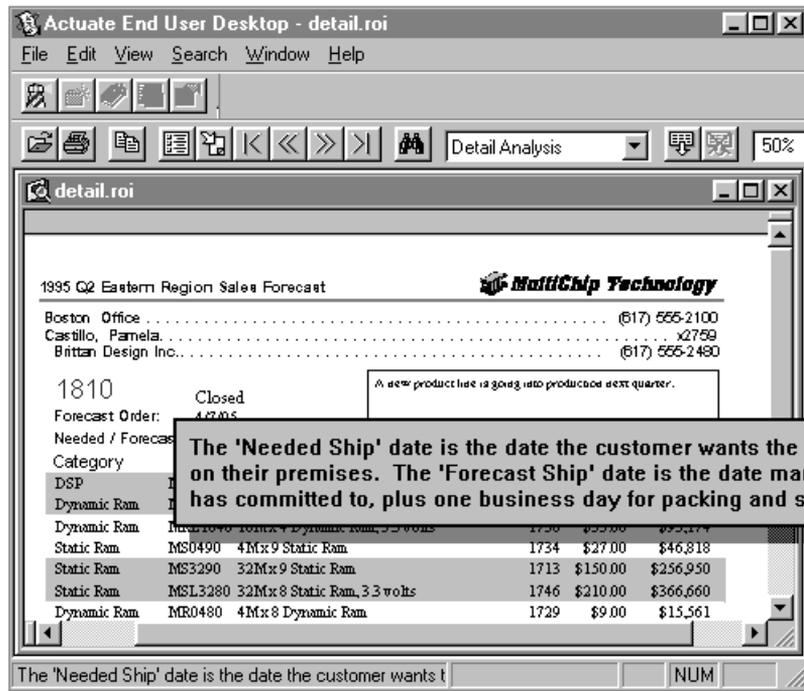
Use the View button to view the topic in the window on the right

Use the Index button to return to the keyword index



## Using report-specific online help

During the design phase, report developers have the option to include report-specific online help. For example, the report developer can add comments to give further detail about specific report objects or to explain how calculations were created.



For detailed information about report-specific online help see Chapter 3, “Viewing a report from the desktop Navigator,” in *Using e.Reports*.

## Typographical conventions

The following table describes the typographical conventions.

Item	Convention	Example
Code examples	Sans serif	Dim As String
File names	All uppercase letters	DETAIL.ROI
Key combination	A + sign between the keys means to press both keys at the same time	Ctrl+Shift
Menu items	Capitalized. No bold	File

<b>Item</b>	<b>Convention</b>	<b>Example</b>
Submenu items	Separated from main menu item with small arrow	File→New
User input or user response	Enclosed in quotation marks	“M*16*”

---

## Syntax conventions

The following table describes the symbols used to present the syntax of Actuate Basic language elements.

<b>Symbol</b>	<b>Description</b>	<b>Example</b>
[]	Optional item	[Alias<alias name>]
<>	Argument you must supply	<expression to format>
{}	Groups two or more mutually exclusive options or arguments	{While   Until}
	Separates mutually exclusive options or arguments in a group	Exit {Do   For   Function   Sub}

# Overview of Actuate e.Reporting Server

This chapter contains the following topics:

- Introducing the e.Reporting Server
- Introducing the Report Encyclopedia
- Performing administration tasks
- Upgrading the Report Encyclopedia

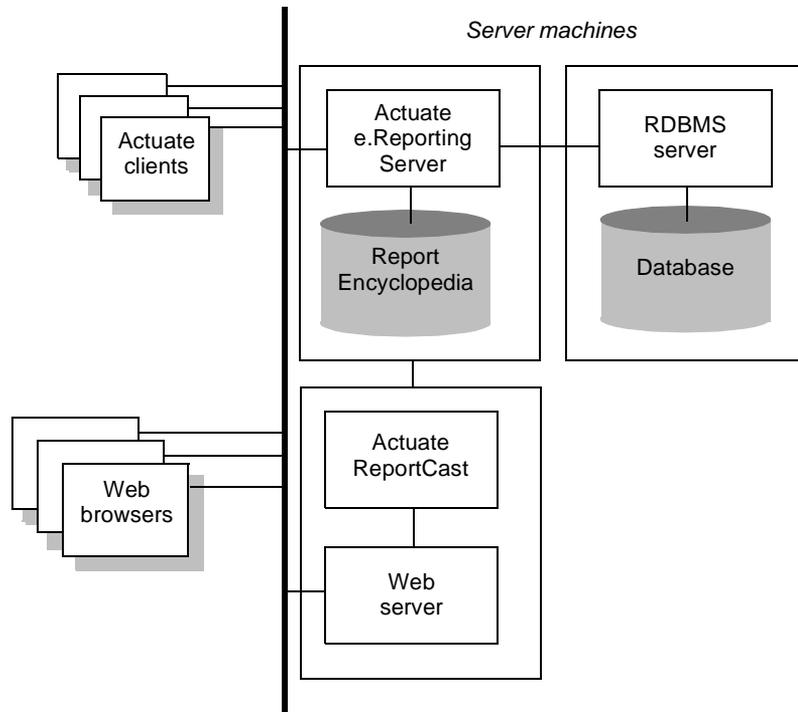
---

## Introducing the e.Reporting Server

In the Actuate Reporting System, developers and users perform tasks such as designing, generating, distributing, viewing, and printing reports. In the process, they work with a variety of report items. These items, along with related data such as access privileges and request queues, are stored in the Report Encyclopedia. The processes that manage the information and service the requests are known collectively as the Actuate e.Reporting Server.

Each report server on a network provides viewing, scheduling, execution, printing, and notification services to its clients. Some of these services, such as scheduling and notification, can be performed only by a report server. Other services augment the basic client facilities and allow clients to use the resources of the server's host machine to reduce the load on client computers. Thus the report server adds capabilities to a reporting system and improves its performance.

When a user attempts to view a report or creates a request to print or generate a report document, the report server processes verify that the request is legal, connects to the data source or database, and schedules and executes the request. The following illustration shows the relationships of the Actuate e.Reporting Server with its Encyclopedia, its clients, and an RDBMS server.



## Report server features

The report server provides a secure and consistent method of managing access to Encyclopedia items:

- The report server provides a hierarchical folder system. Report item names are independent of the syntax used in the underlying file system. As a result, users can organize report items in ways that are familiar and meaningful.
- The report server validates, schedules, and queues requests for report generation. When reports and print jobs are completed, the report server sends a notification message to designated users or groups. When users want to view reports, the report server sends only the requested pages over the network. This set of features, known as Virtual Report Distribution, reduces network traffic and disk-space requirements, and improves security as compared to the physical distribution of printed reports.
- The report server controls access to the Encyclopedia according to user names and passwords. As server administrator, you assign the appropriate levels of privilege to users, and you can group sets of privileges into roles for convenience of administration. With Release 4, Actuate has added the

secure read privilege to the Report Encyclopedia. Secure read lets users view objects in the Report Encyclopedia, but prohibits the downloading of objects. Secure read is used with Release 4 Actuate report documents viewed over the Web and allows the user to view the Actuate report document only as Dynamic HTML (DHTML) over the Web. With the Advanced e.Reporting Server, secure read also enables Actuate report page security.

- Actuate can age and automatically archive aged reports. Using the Report Encyclopedia auto-archive features, administrators and users specify parameters to automatically delete files such as Actuate reports, open server reports, and other documents stored in the Report Encyclopedia. As part of auto-archive, you can specify that a file is archived before deletion.
- In combination with Actuate ReportCast, the Actuate e.Reporting Server lets users access the Report Encyclopedia over the Web and view Release 4 Actuate report documents, .ROIs, in DHTML format. Users can still view report documents in the native .ROI format using the Actuate LRX if they have the appropriate privileges.
- The Release 4 Advanced e.Reporting Server has additional features not available with the standard e.Reporting Server, including open server report generation, report page security, and the Report Server Security Extension (RSSE) application programming interface.

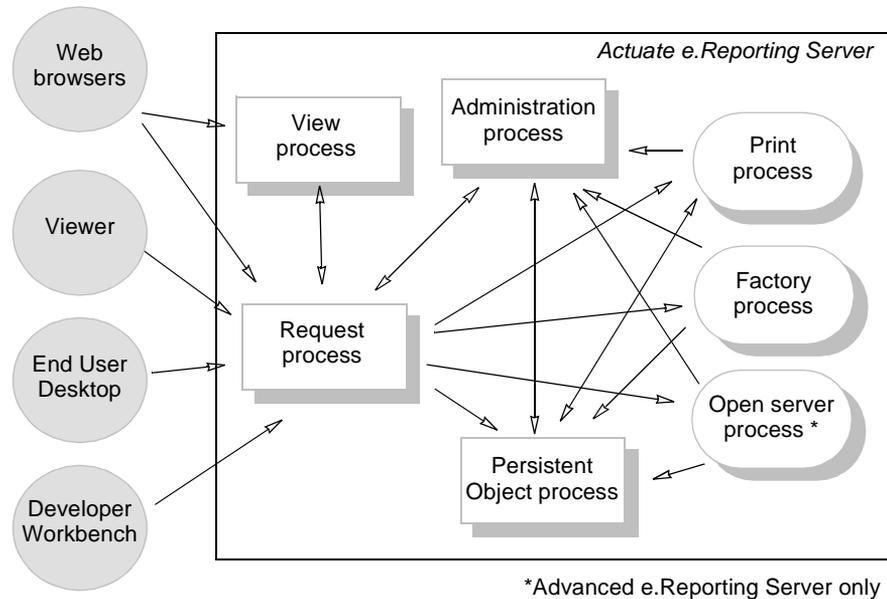
With open server report generation, users can configure the report server to print or run a third party report, called an open server report from the Report Encyclopedia. For example, you can schedule to run or print a Seagate Crystal Reports report document from within the Report Encyclopedia as well as view the report.

With report page security, Actuate report developers create a report document with security rules that determine which pages a user can view. In the Report Encyclopedia, page security is enabled for a user when the user's privilege is set to secure read.

Using the RSSE, developers can create applications that enhance the existing report page security, including integrating report page security functionality with other security tools and applications.

## Report server architecture

The report server manages interaction between clients and the Encyclopedia, enabling activities such as scheduling and running reports, browsing the folder hierarchy, and printing or viewing items. The report server processes execute such client requests. The following illustration shows the relationships among the report server's clients and its processes.



The report server includes the following processes:

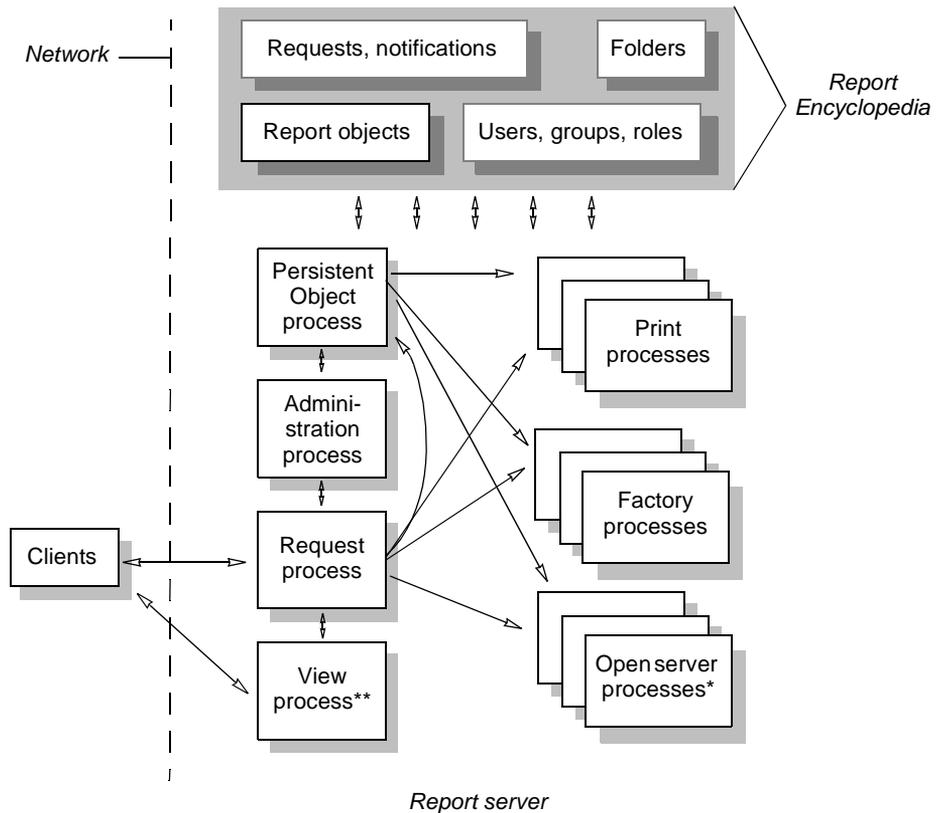
- The Request process receives, prioritizes, and routes incoming requests from clients. If validation is required, as with log-in requests, the Request process passes the request to the Administration process before routing it. After prioritizing the request, the Request process passes it to the appropriate agent such as a Factory, Print, or Persistent Object process. Each report server has one multithreaded Request process capable of handling many client requests concurrently.
- The Administration process validates users and user requests, and enables configuration and management of users, roles, groups, servers, privileges, and events. Each report server has one multithreaded Administration process.
- The Persistent Object process manages persistent objects in the local Encyclopedia volume. Persistent objects include report items and folders. Each report server has one multithreaded Persistent Object process.
- A View process lets a user view and search through an Actuate report in DHTML format when viewing the report using a web browser. The user is logged into a Report Encyclopedia through a web server and Actuate ReportCast. The report server starts a View process when the user selects an Actuate report.

The View process can also convert Actuate report document data to XML for use with other Web applications and PDF for printing.

Once a report run or a print job has been requested, validated, and scheduled, it is executed by the following operations processes:

- Factory processes, which generate reports from report object executables.
- Print processes, which, as their name implies, manage the printing of reports.
- Open server processes, which are used to generate or print third-party, open server reports. Open server report generation and printing is available with the Advanced e.Reporting Server.

The following illustration shows how client requests are handled.



\*Advanced e.Reporting Server only  
 \*\*View process is used with web browsers and ReportCast

---

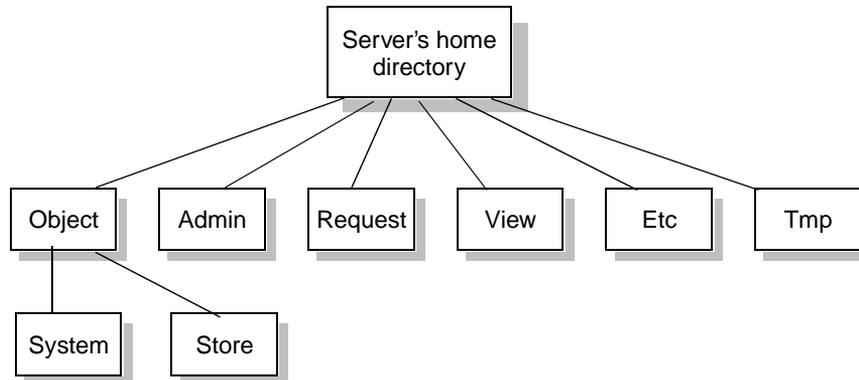
# Introducing the Report Encyclopedia

In the Actuate Reporting System, the Encyclopedia is the central repository for Live Report Documents. The Encyclopedia organizes six kinds of items:

- Report items such as files of report object executables, report object instances, report object designs, and report object values
- Folders
- Users, privileges, and roles
- Requests, schedules, notifications, and notification groups
- Printers and fonts
- Process groups and their schedules

Each Report Encyclopedia has a home folder or directory with a set of subdirectories for storing various item types:

- Persistent objects such as folders and report objects are in the subdirectory Object/Store.
- System-specific data such as file locking, object mapping, and cache information is in the subdirectory Object/System.
- Administration data such as users, roles, groups, server parameters, events, and privileges is in the subdirectory Admin.
- Request and schedule information, such as requests for reports and process group schedules, is in the subdirectory Request.
- View process files are in the subdirectory View.
- System configuration files are in the subdirectory Etc.
- Report server temporary files are in the subdirectory Tmp.



---

## Performing administration tasks

The tasks involved in administering a report server fall into two categories, one concerned with reports and the other with connections to resources such as databases and networks. Each category has a set of tools and procedures appropriate to its tasks.

### Administrator Desktop tasks

Most administration tasks are related to generating and distributing reports:

- Managing the server:
  - Starting and stopping processes
  - Setting default values for resources such as printers
  - Identifying Process Groups and setting their schedules
  - Managing the execution queues
  - Configuring and maintaining auto-archive rules for report objects
- Managing user access and server security:
  - Adding, modifying, and deleting users and their properties
  - Defining levels of access, or roles, and assigning them to appropriate users
  - Defining notification groups (lists of users) to be told when requests are completed
  - Administering Actuate ReportCast channels

If you have the Advanced e.Reporting Server, other tasks might include:

- Maintaining open server reports, including:
  - Creating open server file types to run open server reports
  - Importing open server reports and assigning parameter values to the reports as needed
- Configuring the Report Encyclopedia for Actuate reports that use page security, including:
  - Configuring the Report Encyclopedia ReportCast URL prefix setting
  - Setting up roles for use with reports using page security
  - Giving roles and users secure read privilege on reports using page security

You perform these tasks with the Administrator Desktop, a Windows application. You can use the Administrator Desktop on any client connected to the network. For information about the Administrator Desktop see *Administering the Report Encyclopedia*.

## Report server tasks

A few administrative tasks related to system-level issues must be performed in the UNIX, Windows NT, or Windows 2000 environment:

- Starting and stopping a server
- Checking the server process status
- Clearing the log, request queue, notifications, object store, or administration databases
- Backing up the Encyclopedia
- Configuring and maintaining the optional archive driver and archive software if it is installed

If you have the Advanced e.Reporting Server, other tasks might include:

- Maintaining the open server drivers and third party report generation software
- Installing and configuring the Report Server Security Extension (RSSE) application and any third party software if needed

---

## Upgrading the Report Encyclopedia

You can use the Actuate e.Reporting Server with a Release 3 Report Encyclopedia. Once the Report Encyclopedia is used with a Release 4 report

server, you cannot reinstall the Report Encyclopedia for use with a Release 3 report server. Also, Release 3 Actuate client software cannot connect to a Release 4 Report Encyclopedia.

You must upgrade Actuate reports to Release 4 to use the new Actuate features such as DHTML report viewing and report page security. For information about migration issues see *Upgrading to Actuate e.Reporting Suite 4*. The latest version is on the Actuate web site at [www.actuate.com](http://www.actuate.com).

If you have Release 2 Report Encyclopedia, you must convert the Report Encyclopedia when you upgrade the report server software to use the new Actuate features such as ReportCast channels and DHTML reports.

You convert your Encyclopedia to the current format using the Actuate utilities ACEXPORT and ACIMPORT. For more information about the Actuate utilities, see Chapter 5, "Report Encyclopedia utilities."

Before you use the Actuate command-line utilities ACEXPORT or ACIMPORT, you should back up your Report Encyclopedia.

# Report server administration on UNIX

This chapter contains the following topics:

- Setting up the report server account
- Printing reports containing OLE objects
- Adding Postscript printer models
- Using PostScript and PCL fonts
- Indicating the report server's home directory
- Adjusting Factory process timing
- Adjusting object aging start times
- Adjusting scheduled request removal
- Configuring locale-specific settings
- Configuring year conversion for Actuate
- Configuring report server port numbers
- Starting and stopping a report server
- Clearing items and queues
- Backing up and restoring the Report Encyclopedia

---

## Setting up the report server account

Like all UNIX processes, the processes that perform report server tasks run under a specific account. You must set up this account appropriately to ensure that it functions correctly and preserves the level of security you expect.

You should work with your UNIX system administrator to set up the account for the report server. The account should have the appropriate UNIX privileges and access to the required software, such as database connectivity software, and hardware such as remote database servers and printers.

### Maintaining security

Your choice of an account to run the report server directly affects the security of your system and your report server. To maintain the optimum balance of security and convenience, you should consider this choice carefully when deciding how to apply your site's account conventions.

To run the report server processes, an account must have the appropriate access privileges to the server's directory hierarchy, which contains all the report server objects. As a result, you must control access to the account to maintain a secure server. One way to do this is to install the server under the root account. However, many sites restrict or prohibit the use of root processes to run externally obtained software for security reasons.

In most cases, the best option is to create a special account for the report server. This simplifies server administration by ensuring that the server account is associated only with server-specific issues and events rather than the variety of items that are directed to root accounts. If you exert control over the server account at a level equivalent to the control your site exercises on root accounts, you can maintain an equivalent level of security for the server.

### Accessing resources

For printing and displaying reports, the account running a report server must have access to a machine running X Windows system, an X Windows server, and the report server account's DISPLAY environment variable must point to that X Windows server. To start, the report server must have access to an X Windows server. For example, if the report server machine is running X Windows, you can set DISPLAY to the local machine:

```
# setenv DISPLAY :0.0
```

If you are using another machine as the X Windows server, specify the machine name. This example sets DISPLAY using the machine Bermuda as the X Windows server:

```
# setenv DISPLAY bermuda:0.0
```

Actuate supports X Server version X11R4 and higher. For information on the X Windows system, see the X.Org web site at [www.x.org](http://www.x.org).

The account that runs the report server must also have access to resources such as data sources and hardware devices in order to provide this access to report server clients. For instance, if a network printer is available and accessible to the server account, it can be made accessible to clients; if the printer is not accessible to the server account, it is not accessible to clients.

For a printer to be recognized by the report server:

- On Solaris systems, the printer's name must appear as a subdirectory of the `/etc/lp/printers` directory. On Solaris 2.6 and later, the report server also uses printers listed using the `lpget list` command.
- On HP systems, the printer's name must appear as an entry in the `/etc/lp/member` directory.
- On IBM systems, the printer's name must appear as an entry of the `/etc/qconfig` file.

For the latest information about supported printers and printing, see the latest version of Actuate Supported Products Matrix which can be found on the Actuate web site at the following URL <http://www.actuate.com>.

## Controlling UNIX file descriptors

Setting the UNIX environment variable `AC_SERVER_UNIX_FD_MAX` controls the maximum number of file descriptors the report server or ReportCast process can have open. If the environment variable is not set, the number of file descriptors the ReportCast or report process can have open is the lower of either 32768 (32 K) file descriptors or the system hard limit on `nfiles` (descriptors). As the default, the environment variable is not set.

If `AC_SERVER_UNIX_FD_MAX` is set, the maximum number of file descriptors the process can have open is the lower of either the system hard limit or the value of `AC_SERVER_UNIX_FD_MAX`. For example, if the system hard limit is 4096 and `AC_SERVER_UNIX_FD_MAX` is not set, the descriptor limit for the process is 4096. If the system hard limit is unlimited, and `AC_SERVER_UNIX_FD_MAX` is not set, the maximum number of file descriptors for the process is 32768. If the system hard limit is unlimited, and `AC_SERVER_UNIX_FD_MAX` is set to 65536, the maximum number of file descriptors is 65536. If the system hard limit is 4096 and `AC_SERVER_UNIX_FD_MAX` is set to 2048, the effective descriptor limit for the process is 2048.

Use the UNIX command `ulimit -aH` to find the system hard limit on `nfiles`. For information on file descriptors and the `ulimit` command, see your UNIX system documentation.

## Specifying a configuration file

An Actuate configuration file provides database connections and data sources that are used in Actuate reports. To specify an Actuate configuration file with the e.Reporting Server, use the environment variable `AC_SERVER_CONFIG_FILE` to specify a path and file name. If `AC_SERVER_CONFIG_FILE` is not defined, Actuate uses the database connection properties that are stored with the `.ROX`. For information on the configuration use and syntax, see *Designing e.Reports*.

## Specifying an image search path

For Actuate reports that include images when viewing or printing the report, use the environment variable `AC_IMAGE_SEARCH_PATH` to specify the directories used to search for report images. The value of `AC_IMAGE_SEARCH_PATH` is a semi-colon separated list of directory paths.

Add the environment variable `AC_IMAGE_SEARCH_PATH` to the request server startup script `reqsrvr.sh` in `$AC_SERVER_HOME/bin` directory. For example, add these two lines to set the image search path to the directories `/web/report/images` and `/web/images`:

```
export AC_IMAGE_SEARCH_PATH
AC_IMAGE_SEARCH_PATH=/web/report/images;/web/images
```

## Specifying graph image quality

On UNIX report servers, use the environment variable `AC_HIGH_QUALITY_GRAPH` to control the quality of graph images in Actuate reports. To specify a better quality graph image, set the environment variable to 1 or `TRUE`. If you set the environment variable, the report server takes more time to generate a graph image, but the image is of better quality.

For example, use these two lines to specify better quality graph images:

```
export AC_HIGH_QUALITY_GRAPH
AC_HIGH_QUALITY_GRAPH=TRUE
```

## Starting the report server on reboot

If you want the Actuate e.Reporting Server to start automatically as part of system initialization, you must access the system startup script directory to add a script:

- For Sun Solaris, the script is in the `/etc/rc2.d` directory
- For HP-UX, the script is in the `/sbin/init.d` directory and linked to the `/sbin/rc2.d` directory
- For AIX, you must manually update the startup scripts, such as `/etc/rc`

## Automatic disk space checking

While the report server is running, it checks for available, unused disk space using the values set with the Actuate variables `AC_DISK_SPACE_WARNING` and `AC_DISK_SPACE_FATAL`. These variables found in the report server `diskconf.txt` file in the `$AC_SERVER_HOME/ etc` directory. The report server uses the values:

- `AC_DISK_SPACE_WARNING` value. When the free disk space falls below this value, report server sends an e-mail to the administrator informing the administrator of low disk space.
- `AC_DISK_SPACE_FATAL` value. When the free disk space falls below this level, request server will shutdown other servers and shut itself down. `AC_DISK_SPACE_FATAL` is also checked when the report server starts. If there is not enough space in the disk drive, the report server will not start.
- server log to indicate that fact

The values set as a default in `diskconf.txt` are 50MB and 100MB:

```
AC_DISK_SPACE_FATAL, 50000
AC_DISK_SPACE_WARNING, 100000
```

Actuate recommends a value of 100MB for `AC_DISK_SPACE_WARNING` and 50MB for `AC_DISK_SPACE_FATAL` when the Report Encyclopedia files become greater than 1GB. If the Report Encyclopedia files are less than 100MB, the default values of 50MB and 100MB are sufficient.

Since a Report Encyclopedia tends to grow over time, it is always a good practice update these values when the capacity planning is done. In a production environment, it is always advisable to ensure that there is adequate free space on the report server system.

## Setting up mail notification

To send electronic mail notifying users that a request has been processed, the account that runs the report server must have access to the `sendmail` program, which is located in `/usr/lib/sendmail`.

To ensure that users are notified by electronic mail, their addresses must be correctly registered with the report server. If an address is not properly registered, attempts to send mail to that address fail and the unsent mail is returned to the mailbox of the report server account.

To check for such errors, log in to the account that runs the report server and look for returned mail. Monitoring of this kind provides early information about problems with account registration and mail notification.

---

## Printing reports containing OLE objects

Printing reports containing OLE objects is supported only on the report server on Windows NT and Windows 2000. If you print a report containing an OLE object from the report server on UNIX, the OLE object appears as a rectangle containing an X.

---

## Adding Postscript printer models

Actuate provides support for many printer models on UNIX. To add a postscript printer model, you need the model's Postscript Printer Definition files (.PPD files) supplied by the printer manufacturer. After you add the PostScript printer model, the model appears in the list of printer models when you configure a printer for the Actuate e.Reporting Server.

### How to add a Postscript printer model

- 1 Obtain the Postscript Printer Definition (PPD) file for the printer model from the printer manufacturer.
- 2 Convert the file to UNIX format if necessary.  
If the PPD file is a Microsoft Windows text file, convert it to UNIX file format. Use a UNIX utility such as dos2unix. The UNIX report server cannot use Windows PPD files without converting it to UNIX format.
- 3 Change the extension of the file to .ps. For example, if the PPD file name is hp4si.ppd, rename it to hp4si.ps.
- 4 Copy the renamed PPD file into the Actuate directory `$AC_SERVER_HOME/operation/xprinter/ppds`.

---

## Using PostScript and PCL fonts

The report server can print reports using fonts installed in a PostScript or PCL printer. You can extend the number of available fonts by adding soft fonts to the report server.

When printing reports, the report server looks in the file `font_mapping` to match fonts used in the report with available report server fonts. If the report server finds the font name in `font_mapping`, it looks in the file `psstd.fonts` or `pclstd.fonts` for PostScript or PCL fonts, respectively. If the font is found, it is used for printing. If the font is not found, then the report server uses the default font specified in `font_mapping`.

When you add a soft font that you have licensed for use with the report server and your printers, the report server print process detects it and uses it as necessary. To add a soft font to the report server, you need access to the report server directories to add the soft font files to the appropriate directories and to update report server font-list files.

Actuate supplies the UNIX utility `fontutils` that adds PostScript fonts to a report server. For information about `fontutils` see “UNIX PostScript font utility,” in Chapter 5, “Report Encyclopedia utilities.” The following is information about manually installing PostScript and PCL fonts:

- Add the PCL or PostScript font files to the report server:
  - PostScript font binary file (.PFB) and Adobe font metric file (.AFM)  
If the PostScript AFM file is a Microsoft Windows text file, convert it to UNIX file format. Use a UNIX utility such as `dos2unix`. The UNIX report server cannot use Windows AFM files without converting it to UNIX format.
  - PCL binary font file and .TFM file
- Update the report server PCL or PostScript file list in the `$AC_SERVER_HOME/operation/xprinter` directory:
  - PostScript printer font-list file `psstd.fonts`
  - PCL printer font-list file `pclstd.fonts`
- Update the report server report font-list file `font_mapping` in the `$AC_SERVER_HOME/operation` directory.

## How to add a PostScript or PCL soft font to the report server

- 1 Add the soft-font files to the report server directories.

For PostScript fonts, copy the PostScript font binary file `.pfb` to the `$AC_SERVER_HOME/operation/xprinter/psssoftfonts` directory.

For PCL fonts, copy the binary font file to the `$AC_SERVER_HOME/operation/xprinter/pclsoftfonts` directory.

- 2 Rename the font files.

For PostScript fonts, rename the `.pfb` file to the face name of the font. For example, for the font `Johnny-Bold`, rename the file `Johnny-Bold.pfb` to `Johnny-Bold`.

For PCL fonts, rename the binary font file in a similar manner.

- 3 Add the font metric file for the font to the directory and rename the file to face name of the font.

For PostScript fonts, copy the font's Adobe font metric file .afm to the \$AC\_SERVER\_HOME/operation/xprinter/fontmetrics/afm directory and rename the file.

For PCL fonts, copy the .tfm file to the metric file directory fontmetrics/tfm and rename the font.

- 4 Add an entry for the font in the PostScript or PCL font-list file.

For information about the syntax of the entry, see "Specifying and mapping fonts," next in this chapter.

For PostScript fonts, add an entry to the psstd.fonts file located in the AC\_SERVER\_HOME/operation/xprinter directory. For example, to add the font Johnny-Bold Actuate added this line to the file:

```
Johnny-Bold, -adobe-johnny-medium-i-normal--%d-%d-%d-%d-p-0-iso8859-1
```

For PCL fonts, add an entry in the pclstd.fonts file located in the AC\_SERVER\_HOME/operation/xprinter directory.

- 5 Add an entry in the Actuate file font\_mapping in the AC\_SERVER\_HOME/operation directory.

For information about the syntax of the entry, see "Specifying and mapping fonts," next in this chapter. Actuate added this entry for the font Johnny-Bold:

```
Johnny-Bold, -adobe-johnny-%s-%s-normal--*-%d-%d-%d-%d-p-0-iso8859-1
```

## Specifying and mapping fonts

Actuate uses the X Logical Font (XLF) Description specification to specify and map fonts. The XLF description specifies font properties such as the foundry, family, weight, slant, size, resolution, and character set encoding. For information about the specification for the X Logical Font Description, see *Introduction to the X Window System* by Oliver Jones (ISBN 0-13-499997-5).

## Font entry examples

The Courier New entry in the file font\_mapping consists of two, comma-separated fields:

```
Courier New, -adobe-courier-%s-%s-normal--*-%d-%d-%d-%d-m-0-iso8859-1
```

The first field, Courier New, is the font name used in the report design. The second field, the rest of the line, is the XLF description for Adobe Courier. The report server uses the second field to map the report font to a printer font. In this example, there is an asterisk (\*) in the second field. Actuate uses an asterisk as a wildcard character when mapping fonts. The %s and %d are used as string and numeric placeholders.

In the file `psstd.fonts`, this is the matching entry for the Courier font:

```
Courier, -adobe-courier-medium-r-normal--%d-%d-%d-%d-m-0-iso8859-1
```

This string has specific entries for some of the placeholders and the wildcard in the `font_mapping` entry. The first field, Courier, is the PostScript file name. In this example, Courier.pfb, and Courier.afm.

If you are adding soft fonts to the report server, contact your font manufacturer or UNIX system administrator for the XLF description information.

To map the report design font Arial Narrow to the printer font Adobe Courier, the entry in the file `font_mapping` would be:

```
Arial Narrow, -adobe-courier-%s-%s-normal--*-%d-%d-%d-%d-m-0-iso8859-1
```

For a PostScript printer, the matching entry in the `psstd.fonts` file is:

```
Courier, -adobe-courier-medium-r-normal--%d-%d-%d-%d-m-0-iso8859-1
```

Actuate uses the Courier font, the first field in the entry, when printing the report.

---

## Indicating the report server's home directory

Many of the procedures described in this chapter refer to the environment variable `AC_SERVER_HOME`. This variable must be defined to point to the top-level directory where the server software was installed.

If you use the C Shell, for example, your `.cshrc` file should contain a line similar to this:

```
$ setenv AC_SERVER_HOME /usr/local/actuate
```

## Report server temporary files

The report server creates all the server related temporary files in the `$AC_SERVER_HOME/tmp` directory. The report server start scripts also sets `TMPDIR` to `$AC_SERVER_HOME/tmp`. Any report server operation that creates a temporary file is responsible for deleting that file.

When the report server restarts, it deletes all the files, except for the `*.lock` files, in the `$AC_SERVER_HOME/tmp`. This removes temporary files that remain in the `$AC_SERVER_HOME/tmp` directory from the previous report server shut down.

## Displaying the report server version

Make sure the following path

`$AC_SERVER_HOME/lib`

is appended to the library path environment variable:

- On SunOS the environment variable is `LD_LIBRARY_PATH`
- On AIX the environment variable is `LIBPATH`
- On HP-UX the environment variable is `SHLIB_PATH`

To see the report server version number, run one of the report server executables using the `-v` option:

- 1 Open a command window.
- 2 Go to the Actuate report server's bin directory, for example, `\Actuate\Server\bin`.
- 3 Run one of the report server processes, `admserv4`, `pobserv4`, or `reqserv4` with the `-v` option; for example:

```
# pobserv4 -v
```

The version number is displayed in the command window.

---

## Adjusting Factory process timing

When processing report requests, the report server starts and stops Factory processes to generate reports. After a Factory process is started, the report server waits two seconds before trying to use the Factory process. After a Factory process is started, the report server waits one second between attempts to communicate with the process. If a report server machine is being used heavily by applications, these settings might not allow enough time for a Factory process to start or to reply after it has started. You can adjust the Factory process timing.

### Setting Factory process variables

You can adjust these delay times by using environment variables. `AC_DELAY_AFTER_PROCESS_CREATED` specifies the time delay after a Factory process is started. `AC_DELAY_DURING_PROCESS_STARTUP` specifies the time delay between attempts to communicate with a Factory process. By default, these environment variables are not set.

If you use the C Shell, you can create the environment variable to set delay after the Factory process is created to four seconds:

```
$ setenv AC_DELAY_AFTER_PROCESS_CREATED 4
```

This example uses the environment variable to set the delay between communication attempts after the Factory has started to two seconds:

```
$ setenv AC_DELAY_DURING_PROCESS_STARTUP 2
```

## Setting Factory report generation status interval

When generating an Actuate report, the Actuate Factory server sends checkpoint information to the Report Encyclopedia. The checkpoints occur after the completion of the 1st, 5th, and 10th pages. After the first 10 pages, the interval between updates is 15 seconds. The interval after the first 10 pages can be overridden by an environment variable.

`AC_FACTORY_PAGE_COUNT_UPDATE_INTERVAL` is the environment variable to override the default setting of 15 seconds. The environment variable value is an integer specifying the number of seconds.

---

## Adjusting system process restarts

Administrators can set UNIX environment variables that affect report server startup and report server automatic restarts.

### Adjusting report server startup

Set the `AC_SKIP_CHECK_POINT_AT_START_UP` UNIX environment variable to delay the creation of data specific files by the report server until the report server updates data in the Report Encyclopedia. The default is to create the data files during report server start up. If the environment variable is present, consistency checking is delayed until the first updates to the Report Encyclopedia.

If the environment variable is set, the Persistent Object process starts faster. However, users might experience slower response during first few updates to the Report Encyclopedia.

### Adjusting report server automatic restarts

Actuate automatically restarts the system processes periodically based on the number of requests to ensure optimal use of memory. The system processes are the request, administration, and the Persistent Object processes. The automatic restarts can be turned off and the number of requests between restarts can be modified by setting an environment variable:

```
AC_RECYCLE_OPERATION_SERVERS
```

This defaults to TRUE. Set it to FALSE to disable automatic restarts.

Use this environment variable to set the requests used to trigger a restart:

`AC_OPERATION_SERVERS_RECYCLE_COUNT`

The default is 1000 requests. Set to any other positive integer to change the number of requests that triggers an automatic restart.

If you use the C Shell, for example, you can create the environment variable to increase the `AC_OPERATION_SERVERS_RECYCLE_COUNT` value to 5000:

```
$ setenv AC_OPERATION_SERVERS_RECYCLE_COUNT 5000
```

---

## Adjusting object aging start times

By default, aging starts at 2:15 a.m. every day. The report server administrator can define the times to perform aging each day.

On UNIX systems, set the environment variable `AC_OBJECT_ARCHIVE_THREAD_SCHEDULE`.

The format of the value of the Registry key or environment variable is a semi-colon separated list of the times in ascending order using the following 24-hour format. For example, the following value will start the aging cycle at 3:15 a.m. and 4:15 p.m.:

`03:15;16:15`

To disable auto-archive, set `AC_OBJECT_ARCHIVE_THREAD_SCHEDULE` and do not specify a time.

For information on using object aging, see Chapter 4, “Managing report server resources,” *Administering the Report Encyclopedia*. For information on the auto-archive driver, see “About the archive driver” in Chapter 5, “Report Encyclopedia utilities.”

---

## Setting View process variables

As administrator, you can set environment variables to configure cache sizes, timeout values, and other View process parameters. In most cases, you do not have to change the values for these parameters from the default values. The default values are used if the environment variable is not set, or is not set to a valid value.

There are also two environment variables used for the View process socket port configuration. See “View process socket port configuration” later in this chapter.

The following two variables specify the size of data packets being sent to the View process or from the View process to an agent application:

- `AC_VS_MIN_DATA_PACKET_SIZE` changes the data packet size sent from the report server View process to the agent programs. The default value is 2048 bits. The value should be set to a positive number in multiples of 1024 bytes.
- `AC_VS_READ_BUFFER_SIZE` is the packet size of the data the report server View process sends to a requesting process such as the View process. The default value is 2048 bytes. The data is the XML, DHTML, or other data generated by the report server.

The following two variables set the number of retry attempts between a View process and an agent application and the timeout value for a View process search:

- `AC_VS_CONNECTION_RETRY` is useful in cases when a View process shuts down and the entries in the agent-side cache are no longer valid. The default is to attempt two retries. The retry attempts are made only after receiving the new connection information from the request server. The old entries are deleted from the cache. To change the connection retry number, set `AC_VS_CONNECTION_RETRY` to a non-negative integer.
- `AC_VIEWSERVER_SEARCH_TIMEOUT` is the number of seconds used as the timeout for a View process search operation. If users are experiencing search time outs, increase this value to allow for longer times for searches. The default is 180 seconds or 3 minutes.

If users have performance problems when viewing Actuate reports in DHTML format using a web browser, creating and setting the value of the following environment variable might help performance:

- `AC_VS_CSS_TEMPLATE_CLASS_LIMIT` specifies the maximum number of Actuate report classes allowed in a report that uses cascading style sheets. The default value is 400.

Actuate uses cascading style sheets to display reports in DHTML format. When viewing an Actuate report that contains a large number of Actuate classes in DHTML format in a web browser, problems might occur. If an Actuate report has more report classes than the value of `AC_VS_CSS_TEMPLATE_CLASS_LIMIT`, the View process switches to a form of DHTML that does not use cascading style sheets. The View process switches from the Actuate format DHTML that uses cascading style sheets to the format DHTMLLong that does not use cascading style sheets.

If you use the C Shell, for example, you can create the environment variable to decrease the `AC_VS_READ_BUFFER_SIZE` value to 1024:

```
$ setenv AC_VS_READ_BUFFER_SIZE 1024
```

After starting a View process, the report server sends a communication message to the View process and waits 30 seconds for the reply. If the report server does not receive a response, the report server kills the View process. If a report server machine is being used heavily by applications or is running on a slow machine, the default wait time might not be sufficient for the View process to reply. Administrators adjust the wait time with the environment variable `AC_VIEWSERVER_STARTUP_TIMEOUT`.

- `AC_VIEWSERVER_STARTUP_TIMEOUT` specifies the time in seconds to wait for the reply after the report server sends the first communication message to the View process. By default, this environment variable is not set. If the environment variable is not set, the report server kills the View process after 30 seconds.

The following two variables control the behavior of the Sender ID cache:

- `AC_VIEW_SESSION_CACHE_SIZE` defines the maximum number of view sessions (user and .ROIs) in the view session cache. The default is 4K.
- `AC_VIEW_SESSION_CACHE_TIMEOUT` is the number of seconds before invalidating the contents of a view session cache. In an environment where the Report Encyclopedia data is read-only, this variable should be set to a large number. The default is 3600 seconds or 1 hour.

A View process uses a report's privilege information stored in the cache. Any changes made to the report's Report Encyclopedia privileges is not updated in the cache until the report server invalidates the information in the cache.

The following variables are other View process parameters:

- `AC_MAX_FILE_CACHE_ENTRIES` is the maximum number of entries in the View process cache. The default is 3000.
- `AC_VIEW_FILE_CACHE_SIZE` is the maximum total size of files in the cache directory for all the cached files. The default is 50 for 50 MB.
- `AC_VIEW_FILE_CACHE_TIMEOUT` controls the timeout of the file cache.
- `AC_VIEW_ROX_CACHE_SIZE` is the maximum total size of .ROX files cached in the View process .ROX cache. The default is 64 MB.
- `AC_VS_LOCATION_TTL` is the value used as the timeout for View process connection information (system name + port number) in the cache. The default is 10 minutes.
- `AC_VS_LOCATION_SIZE` is the maximum number of view server locations cached by ReportCast. The cache is used to lower the frequency of interactions with the request server. The default is 2048 entries.
- `AC_VS_MAX_THREAD` is the number of concurrent service threads allowed in the View process before the reactor slows down picking up new messages. Administrators should use multiple View processes if one View

process does not utilize a CPU during peak times. This limit affects the internal behavior of the View process. It does not affect how many requests can be sent to the View process simultaneously. The default is 16.

---

## Open server configuration

When generating or printing an open server report, the report server looks in the report server's \driver directory and the directories specified in the operating system's shared library environment variable to find shared libraries. On UNIX the open server driver uses the paths specified in these environment variables:

- LD\_LIBRARY\_PATH (SunOS)
- LIBPATH (AIX)
- SHLIB\_PATH (HP-UX)

and also looks in the \$AC\_SERVER\_HOME\drivers directory.

You can either add the path name that point to the required libraries using the appropriate environment variable or copy required third-party shared libraries or DLLs to the report server's \driver directory.

If a path is specified for an open server driver, the path is relative to the report server's \driver directory.

The open server driver and third party software must be available and executable from the report server processes.

---

## Adjusting scheduled request removal

By default, the report server removes expired scheduled requests at least 24 hours after the schedule expiration. This allows users to edit expired scheduled requests if they want a similar request to be submitted with a new schedule. The report server removes obsolete scheduled requests when the following conditions are met:

- When all completed notices have been deleted
- If the request will never execute again
- When the last time the schedule could possibly execute is at least 24 hours in the past

## Scheduled request removal environment variable

On UNIX, the following environment variable can be set to specify a longer or shorter time in seconds to override the 24-hour default:

`AC_PURGE_EXPIRED_SCHEDULED_REQUEST_DELAY`

The environment variable must contain an integer value expressing the delay value in seconds. If the value is zero, scheduled requests are not purged. For example, if the delay is set to 3600 seconds (one hour) the scheduled request will expire only one hour after the last time it could execute, and not before.

The report server deletes obsolete requests and information associated with these requests, such as schedules and temporary .ROV files at specific times of the day. You can set the times the deletions occur and specify newly created temporary files to be deleted using the following two environment variables:

`AC_DEAD_REQUEST_PURGE_TIMES`  
`AC_PURGE_WINDOW`

`AC_DEAD_REQUEST_PURGE_TIMES` determines at what times of the day obsolete requests and associated files are deleted. The format of the value of the environment variable is a list of the times separated by semicolons, in ascending order, using a 24-hour format. For example, the following value will start the thread at 3:15 am and 4:15 pm:

`03:15;16:15`

If the times are specified incorrectly, or the variable is not present, the default of 00:15 is used.

`AC_PURGE_WINDOW` can be used to stop deletion of valid temporary .ROV files not yet associated with a request. Any .ROV file created less than the value of `AC_PURGE_WINDOW` seconds before the start of the purge will not be deleted, regardless of its status. The default time is 900 seconds or 15 minutes.

---

## Configuring locale-specific settings

In order to specify formatting for locale specific data such as date, time, and currency the UNIX operating system environment variables must be set properly:

- The `LANG` environment variable must be set to the proper language code.
- The Actuate environment variables must be set to the proper formats in the report server startup scripts.

In a C shell, the following command sets the locale to standard French:

```
#setenv LANG fr
```

You set this environment variable in the `admserv.sh`, `pobserv.sh`, and `reqserv.sh` scripts. For example, add these two lines to set the locale to standard French:

```
export LANG
LANG=fr
```

The language codes for the LANG environment variable differ on Sun, AIX, and HP system. For the language codes for the LANG environment variable, see the system documentation.

When available, locale specific settings are taken from UNIX operating system. However, some local-specific formatting information is not available on UNIX platforms. For those that are not available in UNIX platform, Actuate provides environment variables. To match the locale-specific formatting on UNIX with the Windows formatting, you need to specify the format strings using the environment variables. On Microsoft Windows, Actuate uses the Windows Control Panel settings for locale specific formatting.

## Language environment variables

The following table lists the UNIX environment variables and default values the report server uses.

Format item	Environment variable	Default value
Long Date format	AC_LONG_DATE	dddd, MMMM dd,yyyy
Short Date format	AC_SHORT_DATE	M/d/yy
Date Mode	AC_DATE_MODE	0(m/d/y)
Time separator	AC_TIME_SEPCH	:
Date separator	AC_DATE_SEPCH	/
Time format	AC_TIME_FORMAT	H:mm:ss tt
Currency format	AC_CURRENCY_FORMAT	\$\$,##0.00;(\$\$,##0.00)

### How to set the language environment variables

- 1 Stop the report server processes.
- 2 Set the LANG environment variable.
- 3 On UNIX, the following lines appear in the three report server startup scripts `admserv.sh`, `pobserv.sh`, and `reqserv.sh`. Uncomment the lines and set the format strings in all three startup scripts:

```
## Environment variable to set Format strings
##
## Set LANG environment and Uncomment the following lines
```

```

## and set the format
## strings same as Windows Regional Setting when running Report
## Server other than English locale.
## If the environmnet variable is not set, the default(US) format is used.
##
## export AC_LONG_DATE                               uncomment
## export AC_SHORT_DATE                             uncomment
## export AC_DATE_MODE                               uncomment
## export AC_TIME_SEPCH                             uncomment
## export AC_DATE_SEPCH                             uncomment
## export AC_TIME_FORMAT                             uncomment
## export AC_CURRENCY_FORMAT                         uncomment
##
##AC_LONG_DATE="dddd, MMMM dd,yyyy"                  uncomment
##AC_SHORT_DATE="M/d/yy"                             uncomment
##AC_DATE_MODE=0                                     uncomment
## 0 --> m/d/y
## 1 --> d/m/y
## 2 --> y/m/d
##AC_TIME_SEPCH=":"                                  uncomment
##AC_DATE_SEPCH=""                                   uncomment
##AC_TIME_FORMAT="H:mm:ss tt"                        uncomment
##AC_CURRENCY_FORMAT="$#,##0.00;($#,##0.00)"        uncomment

```

- 4 Restart the report server.

---

## Configuring year conversion for Actuate

When generating a report, the report server converts two-digit year values into four-digit years when the two-digit values are used for scheduling requests in the Report Encyclopedia, or in Actuate Basic functions or Acutate QBE. For example, if a user enters date 01/01/14 in a QBE expression, Actuate converts the date to 2014-01-01.

You can control how Actuate interprets two-digit years by setting the value of the environment variable `AC_CENTURY_BREAK`. Check with your Actuate report developer to see whether you need to set the `AC_CENTURY_BREAK`.

For information about how Actuate converts two-digit years into four-digit years, see Chapter 5, “Using query data streams,” in *Developing Advanced e.Reports*.

---

## View process locale configuration

The report server View process, the process that converts an Actuate report document .ROI to DHTML for viewing using a browser, or to XML or PDF format, can accommodate browsers that use a different locale than the report server machine locale.

To support different locales for converting reports in DHTML, XML, or PDF format on UNIX, Actuate installs a localemap file in the directory \$AC\_SERVER\_HOME/etc. The localemap file contains date and time formatting information for the locales supported by the Actuate report server that the View process cannot get from the machine's operating system. The supported locales are:

- English, Australia
- English, Canada
- English, United Kingdom
- English, United States
- Spanish, Spain
- French, France
- German, Germany
- Japanese, Japan

The localemap file is a text file that contains some Japanese characters. Administrators must make sure the text editor does not corrupt the Japanese characters. Administrators can change locale formats in the file. Administrators must not edit the localemap file while a View process is running. If a wrong entry is made or a locale is specified that is not installed on the machine, an error is logged.

If an administrator changes the report server localemap file, the same change must also be made to the Actuate ReportCast localemap of any web server using Actuate ReportCast that can connect to the report server.

The web browser or other agent application specifies the locale for a viewing session. If the web browser or other application specifies a locale that is not supported or does not specify a locale, the View process sets the viewing session's locale to the default locale. The default locale for the report server is configurable in the localemap file.

There are six fields in an entry line, all of them mandatory. Tabs or spaces separate the fields. All the fields except the first and fifth field, the Actuate locale name, and the input date format fields are enclosed in double quotes (""). The last line of the file is:

\*END\_OF\_FILE.

Do not remove or change the end of file line.

Lines starting with a pound sign, #, are treated as comment lines.

The following is the structure of the localemap file:

```
# Comments
default "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
Locale1 "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
Locale2 "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
...
LocaleN "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
*END_OF_FILE
```

The line with the Actuate locale name field as DEFAULT is the default locale for the View process. The equivalent system locale is the ANSI standard, C. If the DEFAULT localemap entry exists, the View process uses the default fields for the locale and the formats.

The following describe the fields in a localemap locale line:

- Locale is the Actuate locale name that the client must use when specifying a locale, for example, enu. The DEFAULT is the default locale.
- SysLocale is the locale name as defined by the operating system, for example, en\_US on UNIX, and English\_United States.1252 on Windows NT.
- ShortDate is the Actuate Basic Short date format, for example, M/d/yy.
- LongDate is the Actuate Basic Long date format, for example, dddd, MMMM dd, yyyy.
- InputDateMode is the Input date mode. The mode can be 0, 1, or 2. The number specifies the order in which the day, month, and year are specified. The mode does not specify the date separator:
  - 0 represents the order month, day, and year
  - 1 represents the order day, month, and year
  - 2 represents the order year, month, and day
- Time is the time format, for example, h:mm:ss tt.

The first two fields in the localemap line, the LocaleName and SysLocaleName, are specific to Actuate and the operating system in which the report server is running. The administrator cannot change these fields. The administrator can change the other four format fields.

In the format fields, administrators can use spaces and delimiters such as commas ( , ), colons ( : ), dashes ( - ) and slashes ( / ) to format date and time. Enclose text that isn't formatting information in single quotes ( ' ). Double

quotes ( " ) are not allowed within localemap format fields. For example, to specify a format that displays text such as 29th of March, 1999 use this format:

```
"dd'th of 'MMMM, yyyy"
```

In the format, the "th of " is enclosed in single quotes. To use single quotes as part of text, use two single quotes in a row. For example, to create a format that displays dates in the format 29 March, '99 use this format:

```
"dd MMMM, "yy"
```

In the format, there are two single quotes before the yy.

---

## Configuring report server port numbers

You can specify a separate range of port numbers the report server attempts to use for the request server RPC ports, client communication ports, and view server communication ports. Client communication includes communication with Actuate ReportCast. You might need to configure the port numbers for communication between the report server and other applications that use specific port number, such as a firewall.

### RPC and client communication port configuration

If a free port cannot be found in the specified range, the report server tries to find an unused port number outside the range. By default, the report server automatically finds and uses an unused port numbers for RPC and client communication. The request server process also uses the port number 111 for RPC communication. This port number is fixed and cannot be changed.

For the request server RPC port, the start of the range of port numbers is configured with the environment variable `AC_SERVER_RPC_SOCKET_BASE`. The range of port numbers is configured with the environment variable `AC_SERVER_RPC_SOCKET_COUNT`. Both values must be positive integers. The maximum value for both environment variables is 32767.

The report server starts at the `BASE` port and attempts to use up to `BASE + COUNT - 1` port to find a single port. The port is used as the destination port for two-way TCP for ONC/RPC connection from clients or ReportCast.

For ports used for communication between the report server and clients or Actuate ReportCast, the start of the range of port numbers is configured with the environment variable `AC_SERVER_REPLY_SOCKET_BASE`. The range of port numbers is configured with the environment variable `AC_SERVER_REPLY_SOCKET_COUNT`. The report server starts at the `BASE`

port and attempts to use up to  $\text{BASE} + \text{COUNT} - 1$  port to find ports for client communication. Both values must be positive integers. The maximum value for both environment variables is 32767.

The ports are used as origination ports for two-way TCP data reads from the object server, and for one-way UDP request server request notifications. The report server looks for ports starting at BASE to  $\text{BASE} + \text{COUNT} - 1$  port for logins to a report server from a client or ReportCast.

For both pairs of environment variables `AC_SERVER_RPC*` and `AC_SERVER_REPLY*`, if the start of the range of port numbers is not set, the report server ignores the range setting. If the start of the range is set and the range is not set, the report server uses a default range of 1.

## How to configure UNIX report server process port range

The following describes how to configure the RPC and client communication ports:

- 1 If necessary, stop the report server processes.
- 2 Add lines to set the environment variables to report server request server startup script `$AC_SERVER_HOME/bin/reqserv.sh`.

Add the new line before the first line that uses the export command. For example, add these lines to change the `AC_SERVER_RPC_SOCKET` base port number to 4096 and the range to 100, and the `AC_SERVER_REPLY_SOCKET` base port number to 1100 and the range to 200:

```
AC_SERVER_RPC_SOCKET_BASE=4096
export AC_SERVER_RPC_SOCKET_BASE
AC_SERVER_RPC_SOCKET_COUNT=100
export AC_SERVER_RPC_SOCKET_COUNT

AC_SERVER_REPLY_SOCKET_BASE=1100
export AC_SERVER_REPLY_SOCKET_BASE
AC_SERVER_REPLY_SOCKET_COUNT=200
export AC_SERVER_REPLY_SOCKET_COUNT
```

- 3 Start the report server.

## View process socket port configuration

Two environment variables control the View process socket port configuration. Administrators can set these environment variables to control the socket ports used when the View process communicates with the Actuate ReportCast or other clients through a network firewall.

- `AC_VIEW_SERVER_SOCKET_BASE` represents the starting number for the View process socket port to receive the ReportCast or client viewing

requests. `AC_VIEW_SERVER_SOCKET_BASE` must be an integer above 1024. If the environment variable is not set, the default value is 15500.

- `AC_VIEW_SERVER_SOCKET_COUNT` represents the range from the base up to which the view server socket port number can go. The sum of `AC_VIEW_SERVER_SOCKET_BASE` and `AC_VIEW_SERVER_SOCKET_COUNT` must be less than 65535. If the environment variable is not set then the default value is 200.

---

## Starting and stopping a report server

The report server is usually installed so that it starts automatically when the system boots. The report server processes start automatically when a client request arrives. If the server or one of its processes exits abnormally, a monitoring daemon restarts it, and crash recovery procedures restore data files to their condition before the crash. Thus, the normal operation of the report server does not require intervention.

Abnormal situations might arise if, for instance, the report server host is unstable or is connected to a troublesome network. In such situations it is sometimes convenient or necessary to start the report server manually. This task involves starting the Request process, the Administration process, and the Persistent Object process.

### How to start the report server

- 1 Determine whether other instances of these processes are running. Scan the process list with a command of the form:

```
$ ps -ef | grep <process name> | grep -v grep
```

For `<process name>` substitute one of the following:

- `admserv4` (administration process)
- `pobserv4` (persistent object process)
- `reqserv4` (request process)

If a process is already running, you do not need to restart it.

- 2 Change to the server's bin directory and start all three processes at once by issuing the following commands:

```
$ cd $AC_SERVER_HOME/bin
$ start_srvr.sh &
```

Alternatively, you can change to the server's bin directory and invoke shell scripts to start the processes:

```
$ cd $AC_SERVER_HOME/bin
$ admserv.sh &
```

```
$ reqserv.sh &  
$ pobserv.sh &
```

## How to stop the report server

As the Actuate report server user, run the Actuate script `shutdown_srvr.sh` from the report server's `bin` directory. This script notifies current report server users that the report server will be stopped and logs them out before stopping the report server processes. For example:

```
$ shutdown_srvr.sh -t 180 -g 30
```

stops the process in three minutes (180 seconds) and notifies the Administrator at the command line if the Processes have not been halted after 30 seconds. For information about stopping the report server, see “The UNIX `shutdown_srvr` script,” later in this chapter.

As an alternate method, you can stop the processes manually:

- 1 Determine the process ID of each process to be stopped. Scan the process list with a command of the form:

```
$ ps -ef | grep <process name> | grep -v grep
```

For `<process name>` substitute one of the following:

- `admserv4` (administration process)
- `pobserv4` (persistent object process)
- `reqserv4` (request process)

- 2 Kill each process you discovered in the previous step with a command of the form:

```
$ kill <process ID>
```

## The UNIX `shutdown_srvr` script

Actuate provides a script to stop report server processes. The Actuate report server administrator can use this command to stop an Actuate report server. The shutdown script sends current users a message once every 60 seconds warning of the shutdown. If the shutdown time is less than 60 seconds, it sends a single warning message. Prior to shutdown, a final shutdown notice is sent in all cases before logging off users.

During shut down, the script prompts the administrator for a final confirmation to stop the report server processes. Then the script tries to stop the report server processes. If the server processes have not stopped, the script sends a message to the administrator to either continue to wait for the processes to stop or to stop the script.

This is the command line syntax:

```
shutdown_srvr.sh [-t <seconds>] [-g <grace_period>] [-y]
```

These are the shutdown\_srvr parameters

**-t** <seconds>

Number of seconds before server shutdown starts. Default is 60 seconds

**-g** <grace\_period>

Number of seconds to wait for server shutdown. Default is 60 seconds. If the server processes have not stopped, the script sends a message to the administrator to either continue to wait for the processes to stop or to stop the script.

**-y**

Yes responses to any command line prompts. Automatically answer yes to any command line prompts sent to the report server administrator's console. This lets the script stop processes without requiring administrator responses.

---

## Clearing items and queues

Some Report Encyclopedia items can be cleared with a single command from the UNIX server account:

- Schedules and requests, if it is inconvenient to delete requests one at a time from the Administrator Desktop, or if you want to restart the report server with fresh queues
- Report objects, if you want to replace existing objects with a new set
- Administration files, if you want to destroy the existing information about users, privileges, roles, groups, servers, printers, configurations, and so on

Schedules and requests can be cleared independently of the other items. If you clear the report objects, requests that refer to those objects are invalid. If you clear the administration files, information such as owners and privileges for objects is lost, and requests that depend on such information will fail. The objects will still be visible to the administrator user, however.

### How to clear report objects

- 1 Stop the report server.

For information about stopping the report server, see "How to stop the report server," earlier in this chapter.

- 2 Clear report objects with the following command:

```
$ rm -rf $AC_SERVER_HOME/object/*
```

- 3 Restart the report server.

For information about restarting the report server, see “How to start the report server,” earlier in this chapter.

### **How to clear schedules and requests**

- 1 Stop the report server.

For information about stopping the report server, see “How to stop the report server,” earlier in this chapter.

- 2 Clear the queue with the following command:

```
$ rm $AC_SERVER_HOME/request/*
```

- 3 Restart the report server.

For information about restarting the report server, see “How to start the report server,” earlier in this chapter.

### **How to clear the administration databases**

- 1 Stop the report server.

For information about stopping the report server, see “How to stop the report server,” earlier in this chapter.

- 2 Clear the databases with the following command:

```
$ rm $AC_SERVER_HOME/admin/*
```

- 3 Restart the server.

For information about restarting the report server, see “How to start the report server,” earlier in this chapter.

---

## **Backing up and restoring the Report Encyclopedia**

Actuate recommends that you back up the Encyclopedia at least weekly, increasing the frequency as report server activity increases. Back up the Encyclopedia by saving all the files in the appropriate directory tree, for which you can use standard UNIX tools such as tar, or any other backup scheme your installation uses.

You must stop the report server before backing up, or restoring from backup, Report Encyclopedia files.

If you use the Actuate command-line utility ACEXPORT to extract information from an Encyclopedia or ACIMPORT to modify a Report Encyclopedia, you should back up your Report Encyclopedia. For more information about the ACEXPORT and ACIMPORT utilities, see Chapter 5, “Report Encyclopedia utilities.”

The following instructions illustrate how to create and compress a tar file in the report server directory; you might prefer to route the tar file to a tape drive.

### **How to back up the Report Encyclopedia**

- 1** Stop the report server.

For information about stopping the report server, see “How to stop the report server,” earlier in this chapter.

- 2** Use a backup utility. For example, you can use the UNIX tar command:

```
$ cd $AC_SERVER_HOME  
$ tar cvf <archive-name>.tar ./object ./admin ./request
```

On UNIX machines, the environment variable AC\_SERVER\_HOME points to the Report Encyclopedia directory.

- 3** To compress the file you just created, issue the following command:

```
$ compress <archive-name>.tar
```

### **How to restore the Report Encyclopedia**

- 1** Obtain the archive file from its storage location. The example assumes that your current working directory is AC\_SERVER\_HOME, and that the archive file is in the working directory.

- 2** Stop the report server.

For information about stopping the report server, see “How to stop the report server.”

- 3** Delete all the files and directories in the Admin, Object, and Request directories.

- 4** If you compressed the tar file after backing up, uncompress it now.

If you used the UNIX tar command to back up a Report Encyclopedia, restore the backed-up files with the following command:

```
$ tar xvf <archive-name>.tar
```

- 5** Restart the report server.

For information about restarting the report server, see “How to start the report server,” earlier in this chapter.

---

## Setting up mail notification

The report server sends all electronic-mail notifications from the report server's user account. You can test the server's ability to notify a particular user by sending mail to that user from the server's account.

To ensure that users are notified by electronic mail, their addresses must be correctly registered with the report server using the Administrator Desktop. If an address is not properly registered, the report server's attempts to send mail to that address fail and the unsent mail is returned. For more information about creating and modifying user accounts with the Administrator Desktop, see *Administering the Report Encyclopedia*.

### How to test mail notification

- 1 Log in to the account that runs the report server.
- 2 Assuming that your test message is in the file `message.txt`, use the following command to send it, replacing `mail-address` with the user's account name as registered with the report server:

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

If your message arrives, the account is set up correctly and mail notifications will succeed. If the message does not arrive, perform some or all of the following steps:

- Check the spelling of the mail address.
- Log in to an account other than the one used for report server mail and try sending mail to the user in question.
- Compare the user's account name that is registered with your mail facility with the account name registered with the report server. These two account names must match exactly.

If you determine that none of these problems are affecting your situation, and you still cannot send mail to the user from the report server account, contact Actuate support for assistance.

# Report server administration on Windows NT and Windows 2000

This chapter contains the following topics:

- Setting up the report server account
- Printing reports containing OLE objects
- Finding the report server's home directory
- Displaying the report server version
- Adjusting Factory process timing
- Adjusting object aging start times
- Adjusting scheduled request removal
- Configuring year conversion for Actuate
- Configuring report server port numbers
- Starting and stopping a report server
- Configuring report server processes on Windows NT and Windows 2000
- Clearing items and queues
- Backing up and restoring the Report Encyclopedia
- Setting up mail notifications

---

## Setting up the report server account

During installation, the e.Reporting Server setup program uses the account of the user running the setup program and the current machine name to register the report server processes as services. If necessary, the report server administrator should work with the Windows NT or Windows 2000 system administrator to change the account used by the report server services after the software is installed. The Windows NT or Windows 2000 system administrator should set up an account for the report server services. The account should have the appropriate privileges and access to the required software such as database connectivity software, and hardware such as remote database servers and printers. When you set up an account for the report server, you should consider one of these items:

- The report server cannot be run in the system's Administrator account
- The report server account must be able to interact with the Desktop if you use OLE Automation or other utilities that interact with the desktop

Then you can run the report server in an account with one of the configurations that allow interact with the Desktop:

- The report server runs in an account that is a member of the Administrators group
- The report server runs in an account that belongs to a group and a member of the group is logged in when the report server is running
- The report server runs in a user's account that is logged in when the report server is running

## Maintaining security

Your choice of an account to run the report server directly affects the security of your system and your report server. To maintain the optimum balance of security and convenience, you should consider this choice carefully when deciding how to apply your site's account conventions.

In most cases, the best option is to create a special account for the report server. This simplifies report server administration by ensuring that the server account is associated only with report server-specific issues and events. If you exert control over the report server account at a level equivalent to the control your site exercises on system administrator accounts, you can maintain an equivalent level of security for the report server. If you choose this option, you must ensure that system facilities, such as those that manage electronic mail and ODBC connections, are aware of the special report server account; otherwise, features that require those facilities, such as e-mail notification, will be unavailable.

To run the report server, an account must have the appropriate access privileges to the server's directory hierarchy, which contains all the report server objects. As a result, you must control access to the account to maintain a secure server.

## Accessing resources

The account that runs the report server must have access to resources such as data sources and hardware devices in order to provide this access to report server clients. For instance, if a network printer is available and accessible to the server account, you can make it accessible to clients; if the printer is not accessible to the server account, it is not accessible to clients. Use Printers in the Control Panel to make printers accessible on your system.

There is a problem in Microsoft Windows NT 4.0 with respect to services such as the Actuate Report Server Request server process accessing network printers. This problem occurs when the report server processes are installed specifying a User Account with automatic startup.

The work-around is to create a local printer on the NT system that uses the network printer. Using the Printer Administration Wizard, create a local printer and add a port. Specify the location of the network printer as the port. For example, Create a local printer called Rainbow\_local that specifies the network printer \\PrntSrvr2\Rainbow as the port. From the report server, use the printer Rainbow\_local to print to the network printer Rainbow.

For more information about the Microsoft Windows NT 4.0 problem, see the Microsoft Knowledge Base article Q137631 available at [www.support.microsoft.com](http://www.support.microsoft.com).

For the latest information about supported printers and printing, see the latest version of Actuate Supported Products Matrix which can be found on the Actuate web site at the following URL <http://www.actuate.com>.

## Specifying a configuration file

An Actuate configuration file provides database connections and data sources that are used in Actuate reports. To specify a configuration file with e.Reporting Server, set the string registry entry AC\_SERVER\_CONFIG\_FILE in the key HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1. As the string value, specify a path and file name. If AC\_SERVER\_CONFIG\_FILE is not defined, Actuate uses the database connection properties that are stored with the .ROX. For information on the configuration use and syntax, see *Designing e.Reports*.

## Specifying an image search path

For Actuate reports that include images when viewing or printing the report, use the Windows Registry entry `AC_IMAGE_SEARCH_PATH` to specify the directories used to search for report images. The value of `AC_IMAGE_SEARCH_PATH` is a semi-colon separated list of directory paths. For example, to set the image search path to the directories `C:\WEB\REPORT\IMAGES` and `C:\WEB\IMAGES` use

```
C:\WEB\REPORT\IMAGES;C:\WEB\IMAGES
```

Use the Registry editor to create the registry entry.

- 1 To run the Registry editor, choose File→Run.
- 2 In the dialog box type:  
`regedit.exe`  
The Registry editor presents a hierarchical list of Registry keys.
- 3 Highlight the key `HKEY_LOCAL_MACHINE\Software\Actuate\Actuate Report Server\4.1`.
- 4 To create a new string entry for the report server key, choose Edit→New→String value. Change the name of the key to `AC_IMAGE_SEARCH_PATH`.
- 5 To change the value for the entry, highlight the entry and choose Edit→Modify. Enter the list of paths separated by semi-colons. For example, to set the image search path to the directories `C:\WEB\REPORT\IMAGES` and `C:\WEB\IMAGES` enter

```
C:\WEB\REPORT\IMAGES;C:\WEB\IMAGES
```

## Automatic disk space checking

While the report server is running, it checks for available, unused disk space using the values set with the Actuate variables `AC_DISK_SPACE_WARNING` and `AC_DISK_SPACE_FATAL`. These variables found in the report server `DISKCONF.TXT` file in the report server's `\etc` directory. For the default installation, the directory is `C:\Actuate4\server\etc`. The report server uses the values:

- `AC_DISK_SPACE_WARNING` value. When the free disk space falls below this value, report server sends an e-mail to the administrator informing the administrator of low disk space.
- `AC_DISK_SPACE_FATAL` value. When the free disk space falls below this level, request server will shutdown other servers and shut itself down. `AC_DISK_SPACE_FATAL` is also checked when the report server starts. If there is not enough space in the disk drive, the report server will not start.

The values set as a default in DISKCONF.TXT are 50MB and 100MB:

```
AC_DISK_SPACE_FATAL, 50000
AC_DISK_SPACE_WARNING, 100000
```

Actuate recommends a value of 100MB for AC\_DISK\_SPACE\_WARNING and 50MB for AC\_DISK\_SPACE\_FATAL when the Report Encyclopedia files become greater than 1GB. If the Report Encyclopedia files are less than 100MB, the default values of 50MB and 100MB are sufficient.

Since a Report Encyclopedia tends to grow over time, it is always a good practice update these values when the capacity planning is done. In a production environment, it is always advisable to ensure that there is adequate free space on the report server system.

## Configuring Factory process default locale

For Factory processes, the system default locale is ENU. To change the Factory process default locale to C, create the registry entry AC\_OPERATION\_SERVER\_DEFAULT\_C\_LOCALE and set the value to TRUE. The value of the registry entry is either TRUE or FALSE. Using C enhances Factory performance. Use the Registry editor to create the registry entry.

- 1 To run the Registry editor, choose File→Run.
- 2 In the dialog box type:  
regedit.exe  
The Registry editor presents a hierarchical list of Registry keys.
- 3 Highlight the key HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1.
- 4 To create a new string entry for the report server key, choose Edit→New→String value. Change the name of the key to AC\_OPERATION\_SERVER\_DEFAULT\_C\_LOCALE.
- 5 To change the value for the entry, highlight the entry and choose Edit→Modify. Enter TRUE.

---

## Printing reports containing OLE objects

Printing reports containing OLE objects is supported only on the report server on Windows NT and Windows 2000. If you print a report containing an OLE object from the report server on UNIX, the OLE object appears as a rectangle containing an X.

---

## Finding the report server's home directory

Many of the procedures described in this chapter refer to the location `AC_SERVER_HOME`, the top-level directory where the server software is installed.

To find the report server's home directory:

- 1 To run the Registry editor, choose File→Run.
- 2 In the dialog box type:

`regedit.exe`

The Registry editor presents a hierarchical list of Registry keys.

- 3 Under `HKEY_LOCAL_MACHINE\Software\Actuate\Actuate Report Server\4.1`, locate the `AC_SERVER_HOME` entry.

The value of the `AC_SERVER_HOME` entry is the report server's home directory.

## Report server temporary files

The report server creates all the server related temporary files in the `AC_SERVER_HOME\Tmp` directory. Any report server operation that creates a temporary file is responsible for deleting that file.

When the report server restarts, it deletes all the files, except for the `*.LOCK` files, in the `AC_SERVER_HOME\Tmp`. This removes temporary files that remain in the `AC_SERVER_HOME\Tmp` directory from the previous report server shut down.

---

## Displaying the report server version

To see the report server version number, use Microsoft Explorer's file Properties dialog box, for a report server process:

- 1 Start the Microsoft Explorer.
- 2 Go to the Actuate report server's bin directory, for example, `C:\Actuate\Server\bin`.
- 3 Select one of the report server processes, `ADMINSRVR4.EXE`, `POBSRVR4.EXE`, or `REQRSRVR4.EXE` and right-click to display the file's context menu.

- 4 Select properties to display the file's Properties dialog box and select the Version tab to display the file's version information.

---

## Adjusting Factory process timing

When processing report requests, the report server starts and stops Factory processes to generate reports. After a Factory process is started, the report server waits two seconds before trying to use the Factory process. After a Factory process is started, the report server waits 1 second between attempts to communicate with the process. If a report server machine is being used heavily by applications, these settings might not allow enough time for a Factory process to start or to reply after it has started. You can adjust the Factory process timing.

### Setting Factory timing Registry entries

You can adjust the Factory process timing by creating Registry entries for the Actuate e.Reporting Server. The entry `AC_DELAY_AFTER_PROCESS_CREATED` specifies the number of seconds after a Factory process is started. The registry entry `AC_DELAY_DURING_PROCESS_STARTUP` specifies the number of seconds between attempts to communicate with a Factory process. By default these registry keys are not set:

- 1 To run the Registry editor, choose File→Run.
- 2 In the dialog box type:  
`regedit.exe`  
The Registry editor presents a hierarchical list of Registry keys.
- 3 Highlight the key `HKEY_LOCAL_MACHINE\Software\Actuate\Actuate Report Server\4.1`.
- 4 To create a new string entry for the report server key, choose Edit→New→String value. Change the name of the key to either `AC_DELAY_AFTER_PROCESS_CREATED` or `AC_DELAY_DURING_PROCESS_STARTUP`.
- 5 To change the value for the entry, highlight the entry and choose Edit→Modify. The value is the number of seconds.

### Setting Factory report generation status interval

When generating an Actuate report, the Actuate Factory server sends checkpoint information to the Report Encyclopedia. The checkpoints occur

after the completion of the 1st, 5th, and 10th pages. After the first 10 pages, the interval between updates is 15 seconds. The interval after the first 10 pages can be overridden by a registry setting.

AC\_FACTORY\_PAGE\_COUNT\_UPDATE\_INTERVAL is the environment variable to override the default setting of 15 seconds. The environment variable value is an integer specifying the number of seconds.

The registry entry AC\_FACTORY\_PAGE\_COUNT\_UPDATE\_INTERVAL specifies the number of seconds between checkpoint messages sent to the Report Encyclopedia. By default this registry entry is not set:

- 1 To run the Registry editor, choose File→Run.
- 2 In the dialog box type:  
regedit.exe  
The Registry editor presents a hierarchical list of Registry keys.
- 3 Highlight the key HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1.
- 4 To create a new string entry for the report server key, choose Edit→New→String value.
- 5 Change the name of the key to  
AC\_FACTORY\_PAGE\_COUNT\_UPDATE\_INTERVAL.
- 6 To change the value for the entry, highlight the entry and choose Edit→Modify. The value is the number of seconds.

---

## Adjusting system process starts and restarts

Administrators can set registry entries that affect report server startup and report server automatic restarts.

### Adjusting report server startup

Set the registry entry AC\_SKIP\_CHECK\_POINT\_AT\_START\_UP to delay the creation of specific data files by the report server until the report server updates data in the Report Encyclopedia. The default is to create the data files during report server startup. If the registry setting is present, consistency checking is delayed until the first updates to the Report Encyclopedia.

If the registry entry is set, the Persistent Object process starts faster. However, users might experience slower response during first few updates to the Report Encyclopedia.

As administrator, use REGEDIT.EXE to create the registry entry. Create the string entries in this registry key:

HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1

## Adjusting automatic restarts

Actuate automatically restarts the system processes periodically based on the number of requests to ensure optimal use of memory. The system processes are the request, administration, and the Persistent Object processes. The automatic restarts can be turned off and the number of requests between restarts can be modified by setting a registry string entry with the name:

AC\_RECYCLE\_OPERATION\_SERVERS

The default value is TRUE. Set the registry entry string value to FALSE to disable automatic restarts.

Use a registry string entry with the following name to set the requests used to trigger a restart:

AC\_OPERATION\_SERVERS\_RECYCLE\_COUNT

The default is 1000 requests. Set the registry entry string value to any other positive integer to change the number of requests that triggers an automatic restart.

As administrator, use REGEDIT.EXE to create or update the registry entries. Update or create the string entries in this registry key:

HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1

---

## Adjusting object aging start times

By default, aging starts at 2:15 a.m. every day. The report server administrator can define the times to perform aging each day.

On Windows NT and Windows 2000, the administrator sets a string entry on the registry key HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1 with a name AC\_OBJECT\_ARCHIVE\_THREAD\_SCHEDULE.

The format of the value of the Registry key or environment variable is a semi-colon separated list of the times in ascending order using the following 24-hour format. For example, the following value will start the aging cycle at 3:15 a.m. and 4:15 p.m.:

03:15;16:15

To disable auto-archive, set `AC_OBJECT_ARCHIVE_THREAD_SCHEDULE` and do not specify a time.

For information on using object aging, see Chapter 4, “Managing report server resources,” *Administering the Report Encyclopedia*. For information on the auto-archive driver, see “About the archive driver” in Chapter 5, “Report Encyclopedia utilities.”

---

## Setting View process parameters

As administrator, you can create Windows NT and Windows 2000 registry entries to configure cache sizes, timeout values, and other View process parameters. In most cases, you do not have to change the values for these parameters from the default values. There are also two environment variables used for the View process socket port configuration. See “View process socket port configuration” later in this chapter.

The default values are used if the registry entry is not set, or is not set to a valid value. On Windows NT and Windows 2000 as administrator, you update or create the string entries in this registry key:

`HKEY_LOCAL_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1`

After starting a View process, the report server sends a communication message to the View process and waits 30 seconds for the reply. If the report server does not receive a response, the report server kills the View process. If a report server machine is being used heavily by applications or is running on a slow machine, the default wait time might not be sufficient for the View process to reply. Administrators adjust the wait time with the string registry entry `AC_VIEWSERVER_STARTUP_TIMEOUT`.

- `AC_VIEWSERVER_STARTUP_TIMEOUT` specifies the time in seconds to wait for the reply after the report server sends the first communication message to the View process. By default, this registry entry is not set. If the registry entry is not set, the report server kills the View process after 30 seconds.

The following two string registry entries specify the size of data packets being sent to the View process or from the View process to an agent application:

- `AC_VS_MIN_DATA_PACKET_SIZE` changes the data packet size sent from the report server View process to the agent programs. The default value is 2048 bits. The value should be set to a positive number in multiples of 1024 bytes.
- `AC_VS_READ_BUFFER_SIZE` is the packet size of the data the report server View process sends to a requesting process such as the View process.

The default value is 2048 bytes. The data is the XML, DHTML, or other data generated by the report server.

If users have performance problems when viewing Actuate reports in DHTML format using a web browser, creating and setting the value of the following string Registry entry might help performance:

- `AC_VS_CSS_TEMPLATE_CLASS_LIMIT` specifies the maximum number of Actuate report classes allowed in a report that uses cascading style sheets. The default value is 400.

Actuate uses cascading style sheets to display reports in DHTML format. When viewing an Actuate report that contains a large number of Actuate classes in DHTML format in a web browser, problems might occur. If an Actuate report has more report classes than the value of `AC_VS_CSS_TEMPLATE_CLASS_LIMIT`, the View process switches to a form of DHTML that does not use cascading style sheets. The View process switches from the Actuate format DHTML that uses cascading style sheets to the format DHTMLLong that does not use cascading style sheets.

The following two string registry entries set the number of retry attempts between a View process and an agent application and the timeout value for a View process search:

- `AC_VS_CONNECTION_RETRY` is useful in cases when a View process shuts down and the entries in the agent-side cache are no longer valid. The default is to attempt two retries. The retry attempts are made only after receiving the new connection information from the request server. The old entries are deleted from the cache. To change the connection retry number, set `AC_VS_CONNECTION_RETRY` to a non-negative integer.
- `AC_VIEWSERVER_SEARCH_TIMEOUT` is the number of seconds used as the timeout for a View process search operation. If users are experiencing search time outs, increase this value to allow for longer times for searches. The default is 180 seconds, or 3 minutes.

The following two string registry entries control the behavior of the Sender ID cache:

- `AC_VIEW_SESSION_CACHE_SIZE` defines the maximum number of view sessions (user and .ROIs) in the view session cache. The default is 4096.
- `AC_VIEW_SESSION_CACHE_TIMEOUT` is the number of seconds before invalidating the contents of a view session cache. In an environment where the Report Encyclopedia data is read-only, this variable should be set to a large number. The default is 3600 seconds, 1 hour.

A View process uses a report's privilege information stored in the cache. Any changes made to the report's Report Encyclopedia privileges is not updated in the cache until the report server invalidates the information in the cache.

The following string registry entries are other View process parameters:

- `AC_MAX_FILE_CACHE_ENTRIES` is the maximum number of entries in the View process cache. The default is 3000.
- `AC_VIEW_FILE_CACHE_SIZE` is the maximum total size of files in the cache directory for all the cached files. The default is 50 for 50 MB.
- `AC_VIEW_FILE_CACHE_TIMEOUT` controls the timeout of the file cache.
- `AC_VIEW_ROX_CACHE_SIZE` is the maximum total size of .ROX files cached in the View process .ROX cache. The default is 64 MB.
- `AC_VIEWSERVER_MAX_TRANSLATOR` is the maximum number of concurrent XML translator instances in Windows NT and Windows 2000. The default is 9.
- `AC_VS_LOCATION_TTL` is the value used as the timeout for the View process connection information (system name + port number) in the cache. The default is 10 minutes.
- `AC_VS_LOCATION_SIZE` is the number of View process locations cached by ReportCast. The cache is used to lower the frequency of interactions with the request server. The default is 2048 entries.
- `AC_VS_MAX_THREAD` is the number of concurrent service threads allowed in the View process before the reactor slows down picking up new messages. On a Windows NT or Windows 2000 system with less than four CPUs, this number should be fairly small, between eight and sixteen, to avoid thrashing. Administrators should use multiple View processes if one View process does not utilize a CPU during peak times. This limit affects the internal behavior of the View process. It does not affect how many requests can be sent to the View process simultaneously. The default is 8.

---

## Open server configuration

When generating or printing an open server report, the report server looks in the report server's `\DRIVERS` directory and the directories specified in the operating system's shared library environment variable to find shared libraries. On Windows NT and Windows 2000, the report server uses the `PATH` variable. For the default installation, the report server also looks in the following directory:

```
C:\ACTUATE 4\SERVER\DRIVERS
```

You can either add the path names to the `PATH` variable that point to the required libraries or copy required third-party shared libraries or DLLs to the report server's `\DRIVERS` directory.

If a path is specified for an open server driver, the path is relative to the report server's \DRIVERS directory.

The open server driver and third party software must be available and executable from the report server processes.

## Open server configuration for Scribe

To use the Actuate open server with Scribe reports, the Scribe client product is required. The Actuate open server driver for Scribe reports ACSQRDRV.EXE uses the Scribe client software as well as some specific Scribe DLL files. DLL files are installed in C:\Scribe\Sqr\Odb\Binw. To enable access to DLLs either add the directory to the PATH or copy the following Scribe DLLs to the <server\_home>\DRIVERS:

STIMAGES.DLL

SQRWT.DLL

LIBSTI32.DLL

BCLW32.DLL

The report server uses the following Actuate open server files that are in the drivers directory:

ACSQRDRV.EXE

SQREXT.DLL

## Printing Scribe open server reports

Using the Scribe interface, the open server driver cannot redirect output to different printers. The default report server printer is used. For printing .SPF files, the SQRWP.EXE program is assumed to be at the following location:

C:\Scribe\Sqr\ODB\Binw\sqrwp.exe

User's can set a parameter to specify a different path. The string parameter is SQRIBE\_SQRWP\_PATH. In most cases, this parameter can be set as a file type parameter for Report Encyclopedia .SPF file type definition.

## Open server configuration for Crystal Reports

To use Actuate open server with Crystal Reports, the Crystal Reports 7 client product is required. The report server uses the Crystal Reports client software and the following Actuate open server driver in the report server's DRIVERS directory:

ACCRPDRV.EXE

The Crystal Report Engine can be used with the Advanced e.Reporting Server. The Crystal Report Engine is part of the Crystal Reports 7 installation. To use the Crystal Report Engine, you need to install the appropriate Crystal files: required runtime files, database files, export files, and library files. See the Crystal Reports documentation for information on the Crystal Report Engine and its use.

## Printing Crystal Reports open server reports

When printing a Crystal open server report, printing is sent to the default printer and only the number of copies parameter is used.

---

## Adjusting scheduled request removal

By default, the report server removes expired scheduled requests at least 24 hours after the schedule expiration. This allows users to edit expired scheduled requests if they want a similar request to be submitted with a new schedule. The report server removes scheduled requests when the following conditions are met:

- When all completed notices have been deleted
- If the request will never execute again
- When the last time the schedule could possibly execute is at least 24 hours in the past

## Scheduled request removal registry entry

On Windows NT and Windows 2000, set the following string registry entry to specify a longer or shorter time in seconds to override the 24-hour default:

`AC_PURGE_EXPIRED_SCHEDULED_REQUEST_DELAY`

The registry entry string value is an integer value expressing the delay value in seconds. If the value is zero, scheduled requests are not purged. For example, if the delay is set to 3600 seconds (one hour) the scheduled request will expire only one hour after the last time it could execute, and not before. By default this registry key is not set.

The report server deletes obsolete requests and information associated with these requests, such as schedules and temporary .ROV files at specific times of the day. You can set the times the deletions occur and specify newly created temporary files to be deleted using the following two string registry entries:

`AC_DEAD_REQUEST_PURGE_TIMES`  
`AC_PURGE_WINDOW`

AC\_DEAD\_REQUEST\_PURGE\_TIMES determines at what times of the day obsolete requests and associated files are deleted. The format of the value of the environment variable is a list of the times separated by semicolons, in ascending order, using a 24-hour format. For example, the following value will start the thread at 3:15 am and 4:15 pm:

03:15;16:15

If the times are specified incorrectly, or the variable is not present, the default of 00:15 is used.

AC\_PURGE\_WINDOW can be used to stop deletion of valid temporary .ROV files not yet associated with a request. Any .ROV file created less than the value of AC\_PURGE\_WINDOW seconds before the start of the purge will not be deleted, regardless of its status. The default time is 900 seconds or 15 minutes.

On Windows NT and Windows 2000, as administrator update or create the string entries in the following registry key:

HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1

The following describes how to add a string registry entry to the registry key HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1.

- 1 To run the Registry editor, choose Run from the Start menu.
- 2 In the dialog box type:  
regedit.exe  
The Registry editor presents a hierarchical list of Registry keys.
- 3 Highlight the key:  
HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1
- 4 To create a new string entry for the report server key, choose Edit→New→String value. Change the name of the string entry to AC\_PURGE\_EXPIRED\_SCHEDULED\_REQUEST\_DELAY.
- 5 To change the value for the entry, highlight the entry and choose Edit→Modify. The value is the number of seconds.

---

## Configuring year conversion for Actuate

When generating a report, the report server converts two-digit year values into four-digit years when the two-digit values are used for scheduling requests in the Report Encyclopedia, or in Actuate Basic functions or Actuate

QBE. For example, if a user enters date 01/01/14 in a QBE expression, Actuate converts the date to 2014-01-01.

You can control how Actuate interprets two-digit years by setting the value of the registry string value AC\_CENTURY\_BREAK. AC\_CENTURY\_BREAK is located in the registry key HKEY\_LOCAL\_MACHINE\SOFTWARE\Actuate. Check with your Actuate report developer to see whether you need to set AC\_CENTURY\_BREAK.

For information about how Actuate converts two-digit years into four-digit years, see Chapter 5, “Using query data streams,” in *Developing Advanced e.Reports*.

---

## View process locale configuration

The report server View process, the process that converts an Actuate report document .ROI to DHTML for viewing using a browser, or to XML or PDF format, can accommodate browsers that use a different locale than the report server machine locale.

To support different locales for converting reports in DHTML, XML, or PDF format on Windows NT or Windows 2000, Actuate installs a localemap file in the report server etc directory. For example, the location of the etc directory using the default installation directory is C:\Actuate\Server\etc. The localemap file contains date and time formatting information for the locales supported by the Actuate report server that the View process cannot get from the machine’s operating system. The supported locales are:

- English, Australia
- English, Canada
- English, United Kingdom
- English, United States
- Spanish, Spain
- French, France
- German, Germany
- Japanese, Japan

The localemap file is a text file that contains some Japanese characters. Administrators must make sure the text editor does not corrupt the Japanese characters. Administrators can change locale formats in the file. Administrators must not edit the localemap file while a View process is running. If a wrong entry is made or a locale is specified that is not installed on the machine, an error is logged.

The web browser or other agent application specifies the locale for a viewing session. If the web browser or other application specifies a locale that is not supported or does not specify a locale, the View process sets the viewing session's locale to the default locale. The default locale for the report server is configurable in the localemap file.

If an administrator changes the report server localemap file, the same change must also be made to the Actuate ReportCast localemap of any web server using Actuate ReportCast that can connect to the report server.

There are six fields in an entry line, all of them mandatory. Tabs or spaces separate the fields. All the fields except the first and fifth field, the Actuate locale name, and the input date format fields are enclosed in double quotes (""). The last line of the file is:

```
*END_OF_FILE.
```

Do not remove or change the end of file line.

Lines starting with a pound sign, #, are treated as comment lines.

The following is the structure of the localemap file:

```
# Comments
default "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
Locale1 "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
Locale2 "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
...
LocaleN "SysLocale" "ShortDate" "LongDate" InputDateMode "Time"
*END_OF_FILE
```

The line with the Actuate locale name field as DEFAULT is the default locale for the View process. The equivalent system locale is the ANSI standard, C. If the DEFAULT localemap entry exists, the View process uses the default fields for the locale and the formats.

The following describe the fields in a localemap locale line:

- Locale is the Actuate locale name that the client must use when specifying a locale, for example, enu. The DEFAULT is the default locale.
- SysLocale is the locale name as defined by the operating system, for example, en\_US on UNIX and English\_United States.1252 on Windows NT.
- ShortDate is the Actuate Basic Short date format, for example, M/d/yy.
- LongDate is the Actuate Basic Long date format, for example, dddd, MMMM dd, yyyy.
- InputDateMode is the Input date mode. The mode can be 0, 1, or 2. The number specifies the order in which the day, month, and year are specified. The mode does not specify the date separator.
  - 0 represents the order month, day, and year

- 1 represents the order day, month, and year
- 2 represents the order year, month, and day
- Time is the time format, for example, h:mm:ss tt.

The first two fields in the localemap line, the LocaleName and SysLocaleName, are specific to Actuate and the operating system in which the report server is running. The administrator cannot change these fields. The administrator can change the other four format fields.

In the format fields, administrators can use spaces and delimiters such as commas ( , ), colons ( : ), dashes ( - ) and slashes ( / ) to format date and time. Enclose text that isn't formatting information in single quotes ( ' ). Double quotes ( " ) are not allowed within localemap format fields. For example, to specify a format that displays text such as 29th of March, 1999 use this format:

```
"dd'th of 'MMMM, yyyy"
```

In the format, the "th of " is enclosed in single quotes. To use single quotes as part of text, use two single quotes in a row. For example, to create a format that displays dates in the format 29 March, '99 use this format:

```
"dd MMMM, ''yy"
```

In the format, there are two single quotes before the yy.

## Configuring report server port numbers

You can specify a separate range of port numbers the report server attempts to use for the Request Server RPC port, client communication ports, and View process ports. Client communication includes communication with the Actuate ReportCast. You might need to configure the port numbers for communication between the report server and other applications that use specific port numbers such as a firewall.

### RPC and client communication port configuration

If a free port cannot be found in the specified range, the report server tries to find an unused port number outside the range. By default, the report server automatically finds and uses unused port numbers for RPC and client communication. The request server process also uses the port number 111 for RPC communication. This port number is fixed and cannot be changed.

For the request server RPC port, the start of the range of port numbers is configured by setting a registry string value with the name AC\_SERVER\_RPC\_SOCKET\_BASE. The range of port numbers is set using a string value with the name AC\_SERVER\_RPC\_SOCKET\_COUNT. Both string

value names are added to the key  
HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1.  
Both values must be positive integers. The maximum value for both values is 32767.

The report server starts at the BASE port and attempts to use up to BASE + COUNT – 1 port to find a single port. The port is used as the destination port for two-way TCP for ONC/RPC connection from clients or the ReportCast.

For ports used for communication between the report server and clients or the ReportCast, the start of the range of port numbers is configured by setting a registry string value with the name AC\_SERVER\_REPLY\_SOCKET\_BASE. The range of port numbers is set using a string value with the name AC\_SERVER\_REPLY\_SOCKET\_COUNT. Both string value names are added to the key HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1. Both values must be positive integers. The maximum value for both values is 32767.

The ports are used as origination ports for two-way TCP data reads from the object server and for one-way UDP Request Server request notifications. The report server looks for ports starting at BASE to BASE + COUNT – 1 port for logins to a report server from a client or the ReportCast.

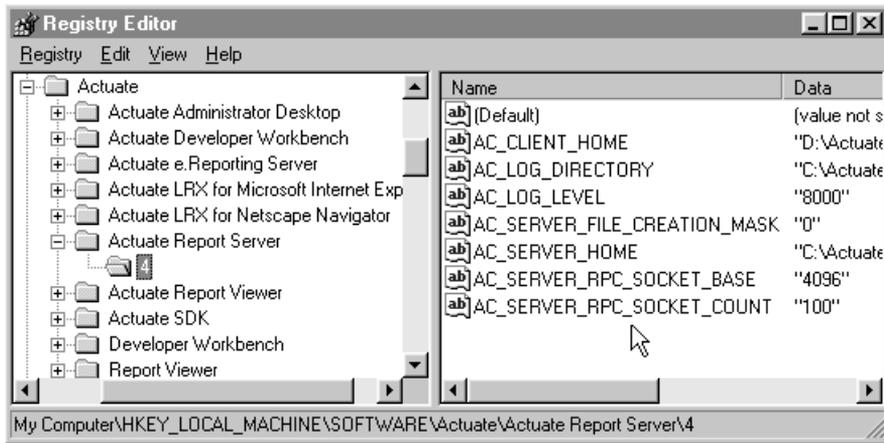
For both pairs of values AC\_SERVER\_RPC\* and AC\_SERVER\_REPLY\*, if the start of the range of port numbers is not set, the report server ignores the range setting. If the start of the range is set and the range is not set, the report server uses a default range of 1.

## **How to configure Windows NT report server port range**

The following describes how to configure the RPC and client communication ports:

- 1 If necessary, stop the report server processes.
- 2 Start the registry editor REGEDIT.EXE.
- 3 Go to the registry key  
HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1.
- 4 Create the string value name AC\_SERVER\_RPC\_SOCKET\_BASE and the value of first port number.
- 5 Create the string value name AC\_SERVER\_RPC\_SOCKET\_COUNT and value of the range of port numbers.

This example shows the base port number set to 4096 and the range of port numbers set to 100.



- 6 Create value names AC\_SERVER\_REPLY\_SOCKET\_BASE and AC\_SERVER\_REPLY\_SOCKET\_COUNT and their values.
- 7 Close the Registry Editor.
- 8 Restart the report server.

## View process socket port configuration

Two string registry entries control the View process socket port configuration. Administrators can set these registry entries to control the socket ports used when the View process communicates with Actuate ReportCast or other clients through a network firewall.

- AC\_VIEW\_SERVER\_SOCKET\_BASE represents the starting number for the View process socket port to receive the ReportCast or client viewing requests. AC\_VIEW\_SERVER\_SOCKET\_BASE must be an integer above 1024. If the Registry entry is not set, the default value is 15500.
- AC\_VIEW\_SERVER\_SOCKET\_COUNT represents the range from the base up to which the view server socket port number can go. The sum of AC\_VIEW\_SERVER\_SOCKET\_BASE and AC\_VIEW\_SERVER\_SOCKET\_COUNT must be less than 65535. If the Registry entry is not set then the default value is 200.

Set the registry entries in the following key:

HKEY\_LOCAL\_MACHINE\Software\Actuate\Actuate Report Server\4.1

---

## Starting and stopping a report server

The report server is started by the Windows service-control manager each time the system boots, or at your explicit direction. Normal operation of the report server does not require intervention. Abnormal situations might arise if, for instance, the report server host is unstable or is connected to a troublesome network. In such situations it is sometimes convenient or necessary to start the report server manually. This task involves starting the Request process, the Administration process, and the Persistent Object process.

### How to start the report server

- 1 Open the Services dialog box from the Control Panel. Locate the following processes in the Services list:
  - Actuate Request Server
  - Actuate Administration Server
  - Actuate Persistent Object Server
  - NobleNet Portmapper for TCP
- 2 The status column contains “Started” for active processes and is blank for inactive ones. Choose Start for each process that is not currently running. If a process is already running, you do not need to restart it.

### How to stop the report server

- 1 Open the Services dialog box from the Control Panel. Locate the following processes in the Services list:
  - Actuate Request Server
  - Actuate Administration Server
  - Actuate Persistent Object Server
- 2 Choose Stop for each process that is currently running.
- 3 If you are stopping the report server to install a new version, you also need to stop the NobleNet Portmapper for TCP service.

---

## Configuring report server processes on Windows NT and Windows 2000

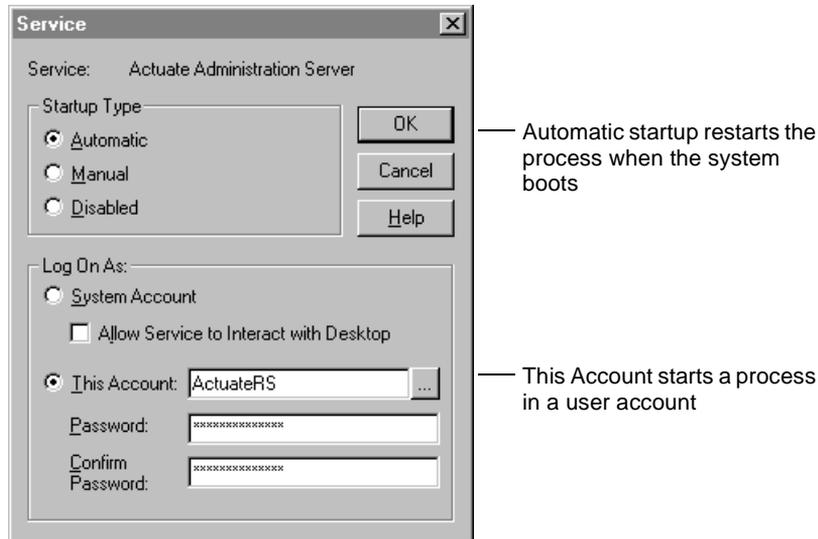
To access printers and ODBC data sources, the Actuate Report processes must be configured to run as a user account, not as the system account. You must configure the report server processes to run in a user account.

## How to assign an account to a process

Open the Services dialog box from the Control Panel. Locate the following processes in the Services list:

- Actuate Request Server
- Actuate Administration Server
- Actuate Persistent Object Server

- 1 For each process, select it from the Services list and choose Startup to open the Service dialog box.
- 2 In the Log On As section, select the This Account option, select the report server account from the list and enter the password for the account. In the Startup Type section, select the Automatic option. Selecting Automatic starts the process when the system boots.



- 3 Stop and restart the processes. If a process configured correctly, you do not need to restart it.

For information about starting and stopping a report server, see “Starting and stopping a report server,” earlier in this chapter.

---

## Clearing items and queues

Some items and queues can be cleared quickly from the MS-DOS prompt:

- Report objects, if you want to replace existing objects with a new set.

- Request queue, if it is inconvenient to delete requests one at a time from the Administrator Desktop, or if you want to restart the report server with fresh queues.
- Administration databases, if you want to destroy the existing databases for users, roles, groups, servers, printers, configurations, and so on.

To clear these items and queues, you need to know the location of the report server's home directory, which is stored in the Registry as the value of the key `AC_SERVER_HOME`.

Schedules and requests can be cleared independently of the other items. If you clear the report objects, requests that refer to those objects are invalid. If you clear the administration files, information such as owners and privileges for objects is lost, and requests that depend on such information will fail. The objects will still be visible to the administrator user, however.

### **How to clear report objects**

- 1 Stop the report server.

For more information about stopping the report server, see "How to stop the report server," earlier in this chapter.

- 2 Change to the `AC_SERVER_HOME` directory.

For information about finding this directory, see "Finding the report server's home directory," earlier in this chapter.

- 3 Clear report objects by issuing the following commands from the DOS prompt:

```
C:\> cd object  
C:\> deltree *.*
```

- 4 Restart the report server.

For more information about starting the report server, see "How to start the report server," earlier in this chapter.

### **How to clear the request queue**

- 1 Stop the report server.

For information about stopping the report server, see "How to stop the report server," earlier in this chapter.

- 2 Change to the `AC_SERVER_HOME` directory.

For information about finding this directory, see "Finding the report server's home directory," earlier in this chapter.

- 3 Clear the queue by issuing the following commands from the DOS prompt:

```
C:\> cd request  
C:\> deltree *.*
```

- 4 Restart the report server.

For more information about restarting the report server, see “How to start the report server,” earlier in this chapter.

### **How to clear the administration databases**

- 1 Stop the report server.

For information about stopping the report server, see “How to stop the report server,” earlier in this chapter.

- 2 Change to the AC\_SERVER\_HOME directory.

For information about finding this directory, see “Finding the report server’s home directory,” earlier in this chapter.

- 3 Clear the databases by issuing the following commands from the DOS prompt:

```
C:\> cd admin  
C:\> del *.*
```

- 4 Restart the report server.

For information about restarting the report server, see “How to start the report server,” earlier in this chapter.

---

## **Backing up and restoring the Report Encyclopedia**

Actuate recommends that you back up the Encyclopedia at least weekly, increasing the frequency as server activity increases. You back up the Encyclopedia by saving all the files in the appropriate directory tree. You can use tools such as Winzip or any other backup utility your installation uses.

The data files are stored in the Admin, Object, and Request subdirectories under the AC\_SERVER\_HOME directory. These directories contain files with long file names and file names with two periods ( . ). For example, the files \_127C\_4700A8C0\_B435CEF3\_8A.POB and 1.PRP.DAT are in the Object\Store directory. Any archive tool you use must be able to handle these file names.

You must stop the report server before backing up, or restoring from backup, Report Encyclopedia files. While the report server is running, it sets a write lock on Encyclopedia files so other Windows programs cannot access the files.

If you use the Actuate command-line utility ACEXPOR to extract information from an Encyclopedia or ACIMPORT to modify a Report Encyclopedia, you should back up your Report Encyclopedia. For information about the ACEXPOR and ACIMPORT utilities, see Chapter 5, "Report Encyclopedia utilities."

The following instructions illustrate how to use the Winzip utility to create a zip file in the report server directory and save the relative directory structure and long file names.

## How to back up the Report Encyclopedia

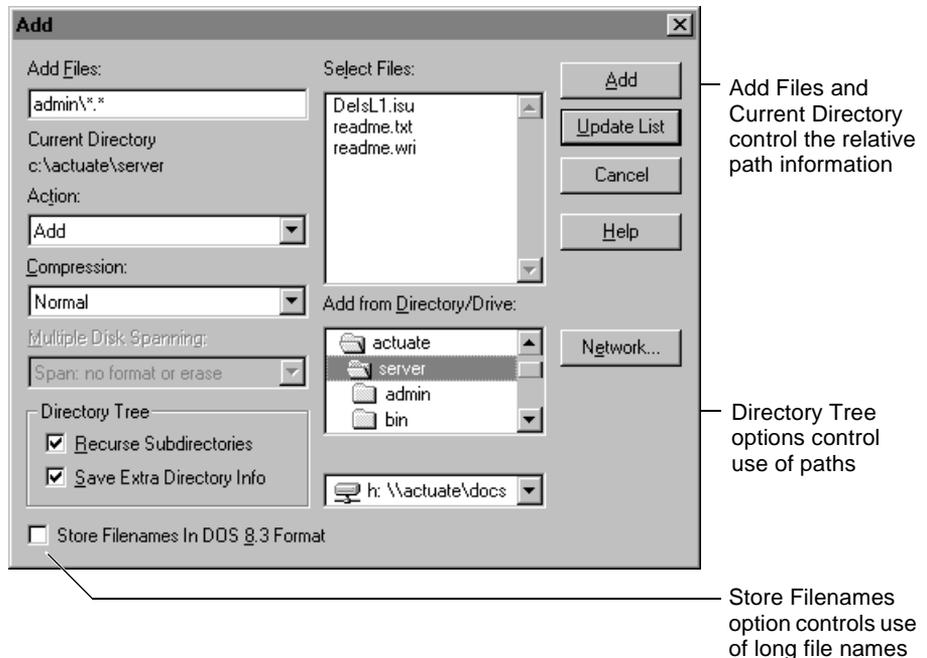
On Windows NT and Windows 2000, the Registry entry AC\_SERVER\_HOME on the report server machine points to the Report Encyclopedia directory:

- 1 Stop the report server.

For information about stopping the report server, see "How to stop the report server," earlier in this chapter.

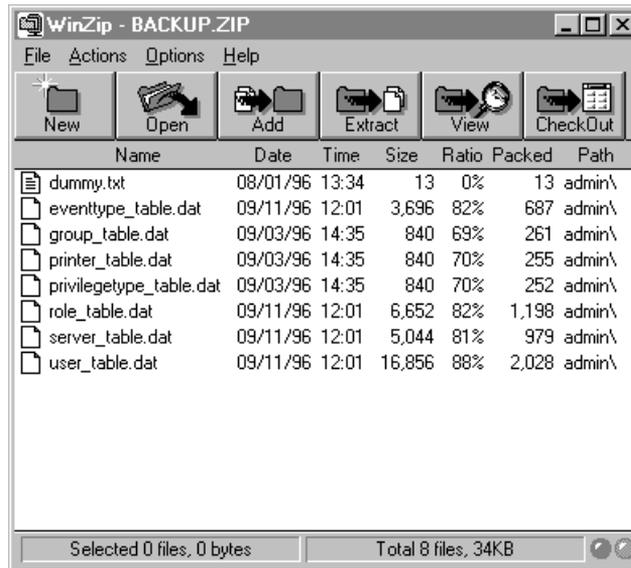
- 2 Start the Winzip utility winzipnt.exe, create a new archive file in the report server's home directory. On Windows NT and Windows 2000, the Registry entry AC\_SERVER\_HOME on the report server machine points to the Report Encyclopedia directory.

- 3 Select Add to display the Add dialog box.



- 4 In the Directory Tree section, enable both Recurse Subdirectories and Save Extra Directory Info. Disable Store Filenames in DOS 8.3 Format. This saves files with relative directory structure and long file names.
- 5 From the Add From Directory/Drive list, select the AC\_SERVER\_HOME directory.
- 6 Save the Admin, Object, and Request directories.

In the Add Files box, enter the subdirectory and files to add to the archive and choose Add. For example, to save the files in the Admin directory type "admin\\*.\*". Repeat for the Object and Request directories.



- 7 Close the archive file and move it to the storage location.
- 8 Restart the report server.

For information about restarting the report server, see "How to start the report server," earlier in this chapter.

## How to restore the Report Encyclopedia

- 1 Obtain the archive file from its storage location.
- 2 Stop the report server.

For information about stopping the report server, see "How to stop the report server," earlier in this chapter.

- 3 Delete all the files and directories in the Report Encyclopedia's Admin, Object, and Request directories. On Windows NT and Windows 2000, the Registry entry AC\_SERVER\_HOME on the report server machine points to the Report Encyclopedia directory.
- 4 Restore the backed-up files by using Winzip or your archive utility. When you restore the Encyclopedia, restore the files to the correct directories.
- 5 Restart the server.

For information about restarting the report server, see "How to start the report server," earlier in this chapter.

---

## Setting up mail notifications

The report server sends all electronic-mail notifications from a single account, which must be listed in the system Registry. When the report server's mail account is registered, it sends normal mail messages using standard tools. Therefore you can test the server's ability to notify a particular user by sending mail to that user from the server's account.

To use Windows messaging, the report server machine must be set up properly. Windows messaging must be installed on the report server machine and the report server must have a mail account. For Microsoft Exchange, the report server must have a profile. For Microsoft Mail, the report server must have a mailbox ID. For information about installing and configuring Windows messaging, see the Microsoft documentation.

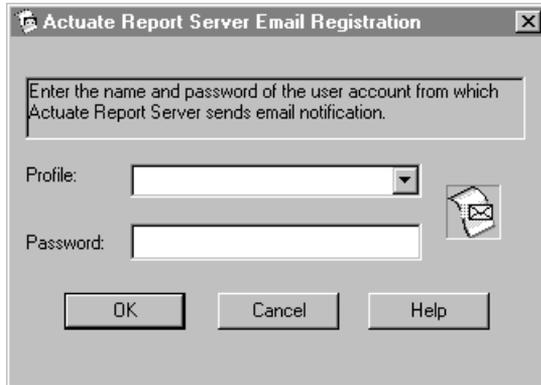
On Windows NT 4.0, if the report server account is configured to use Microsoft Exchange, you must set up a profile before registering the report server's mail account. Use the Mail icon in the Control Panel folder to create a profile. For more information about profiles, see the Microsoft Exchange documentation.

To ensure that users are notified by electronic mail, their addresses must be correctly registered with the report server using the Administrator Desktop. If an address is not properly registered, the report server's attempts to send mail to that address fail and the unsent mail is returned. For more information about creating and modifying user accounts with the Administrator Desktop, see *Administering the Report Encyclopedia*.

### How to register the report server's mail account

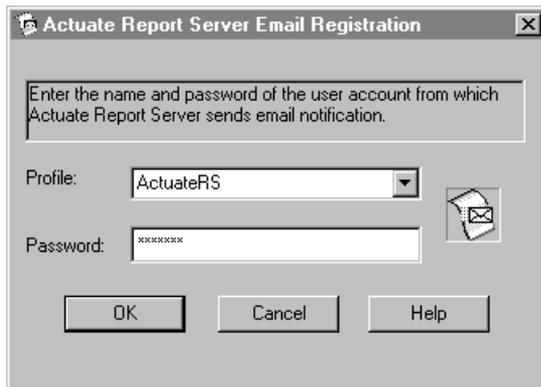
- 1 Run the mailinst program, which is stored in the report server's bin directory.

A dialog box appears prompting for a profile name and a password, as shown in the following illustration.



- 2 If you use Microsoft Exchange, select a profile name, if you use a Microsoft Mail server, select a mailbox ID.
- 3 Type the password and choose OK.

Mailinst modifies the Registry to reflect the information you have provided.



### How to test mail notification

- 1 Log in to the account that runs the report server.
- 2 Assuming that your test message is in the file MESSAGE.TXT, use the following command to send it as an attachment, replacing mail-address with the user's account name as registered with the mail system:

```
C:\actuate\server\bin:> rsmail.exe -Mmail-address -Amessage.txt -LTRUE
```

For information about the rsmail parameters, see "RSMAIL.EXE parameters," later in this chapter.

The following are examples using various types of e-mail names:

```
C:\actuate\server\bin> rsmail.exe -Macutate -Amessage.txt -LTRUE
C:\actuate\server\bin> rsmail.exe -M"Dana Smith" -Amessage.txt -LTRUE
C:\actuate\server\bin> rsmail.exe -M"smtp:dsmith@mchip.com"
-Amessage.txt -LTRUE
```

The second example uses quotes because the name contains a space. The last example is an Internet e-mail address.

If your message arrives, the account is set up correctly and mail notifications will succeed. If the message does not arrive, perform some or all of the following steps:

- Check the spelling of the mail address.
- Log in to an account other than the one used for report server mail and try sending mail to the user in question.
- Compare the user's account name that is registered with your mail facility with the account name registered with the report server. These two account names must match exactly.

The name must be resolvable by your mail facility. Examples of unresolvable names are names that are invalid e-mail names, names that do not have aliases, or names that can be matched to more than one user. An example of the last situation would be an entry of jds which can be mapped to either jdsmith or jdsmythe.

If you have eliminated these possibilities, and you still cannot send mail to the user from the report server account, contact Actuate support for assistance.

## RSMAIL.EXE parameters

The RSMAIL.EXE is a command-line utility that sends mail to a user. This utility uses the Microsoft Mail account or Exchange profile configured for the report server as the originator of the mail.

You execute RSMAIL.EXE in a Command Prompt window and you must be in the Actuate\_server\Bin directory. You specify parameters when you execute the command. The -M parameter is required, the others are optional. The parameters can be in any order but the option letters must be uppercase letters.

These are the RSMAIL.EXE parameters:

**-M**<mailing address>

Required parameter. Specifies the recipient's mail address.

**-S<subject>**

Specifies the text displayed as the subject. If the subject text contains spaces, enclose the text in spaces.

```
C:\actuate\server\bin> rsmail.exe -M"Dana Smith" -S"This only a test"
```

**-P<prelude file>**

Specifies a file sent as included text in the mail.

**-A<attachment file>**

Specifies a file sent as an attachment.

**-D<attachment file display name>**

Specifies the attachment's file name that appears in the mail.

**-L<TRUE or FALSE>**

Specifies the linked file setting. If you use the **-A** parameter to send an attachment, you must use the **-LTRUE** parameter. The default is **FALSE**.

**-H**

Lists the RSMAIL.EXE parameters.

The following example sends mail to Dana Smith with the file MESSAGE.TXT as an attachment:

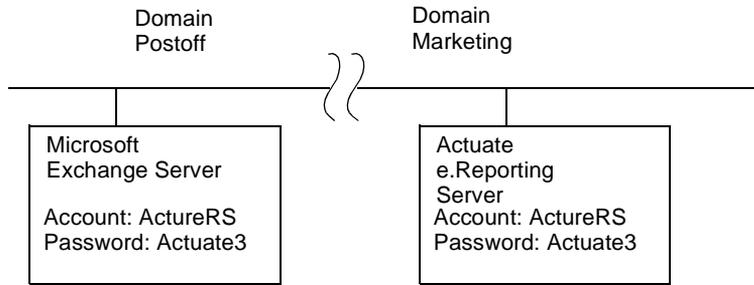
```
C:\actuate\server\bin> rsmail.exe -M"Dana Smith" -Amessage.txt  
-D message.txt -LTRUE.
```

## Working with the Exchange Server and domains

To use Microsoft Exchange and the report server e-mail notification, you have to configure both the report server and Exchange Server if they are on different domains. You must set up an Actuate report server account with these properties:

- The account running the Actuate report server must also be an account on the Microsoft Exchange Server.
- The passwords for the account on the Actuate report server and the account on the Microsoft Exchange Server must be the same.

For example, if the report server is on a machine in the domain Marketing and is running in the account ActuateRS and the Exchange Server on a machine in the domain Postoff, the machine running the Exchange Server must have an account named ActuateRS and the password must be the same on both machines.





# Database connections

This chapter contains the following topics:

- About database connections
- Defining environment variables
- Connecting to DB2 databases
- Connecting to Informix databases
- Connecting to MS/SQL databases
- Connecting to databases through ODBC
- Connecting to Oracle databases
- Connecting to Progress 8 databases
- Connecting to Progress 9.1 databases
- Connecting to Sybase databases
- UNIX factsrvr database compatibility

---

## About database connections

Some reports can be generated without database information; others require information from several databases. Reports that use information from databases must connect to the databases to obtain the information.

When connecting to a database, users must supply a user name and a password. Databases from most vendors require additional information that is specific to the vendor; for example, many databases need definitions for environment variables. To supply the user name and password, developers edit a connection component with the Component Editor in the Developer Workbench. For information about using the Component Editor, see Chapter 5, “Using query data streams,” in *Developing Advanced e.Reports*.

In most cases, the report server and the database run on different computers for load-balancing purposes. However, this division is not required; you can run the report server on the database host if you want to use that computer’s resources to improve performance of some report server tasks.

### Stored procedures

When you use the following Actuate connections:

- ODBC connection on Windows NT and Windows 2000
- Sybase ctlib
- Oracle 8 client and server
- Progress 9.1

Actuate supports stored procedures with the stored procedure data source component. In the Developer Workbench, you use a stored procedure data source and the Stored Procedure Data Source Builder dialog box to specify the stored procedure used in your report. For information about using stored procedures and the Stored Procedure Builder, see Chapter 5, “Using query data streams,” in *Developing Advanced e.Reports*.

Using the stored procedure data source, Actuate supports only single data sets returned to Actuate. If a complex result set is returned, Actuate uses only the first data set. For information about processing multiple result sets, see Chapter 11, “Using stored procedures in SQL databases,” in *Programming e.Reports*.

---

## Defining environment variables

Many of the database connections described here require you to define environment variables. Two important considerations apply when you create these definitions.

If your report server runs on UNIX, insert the definitions in the file `reqserv.sh`, which is in the directory `$AC_SERVER_HOME/bin`. Thus you might add the following lines to your `reqserv.sh` file if you used a Sybase database:

```
export SYBASE
SYBASE=/usr/local/sybase
```

If your report server runs on Windows NT or Windows 2000, you must define system environment variables rather than user environment variables. Defining a user environment variable is not equivalent; report server processes are system processes, and cannot access your user environment.

---

## Connecting to DB2 databases

The Actuate e.Reporting Server works with DB2 (IBM Database 2) on UNIX, Windows NT and Windows 2000. To connect with DB2 databases, you must:

- Define appropriate environment variables.
- Specify the database environment.
- Supply an account name and password.
- Specify the protocol your site uses.
- On UNIX, install the proper Factory server software.

For more information about installing and using DB2 with the DB2 Client Application Enabler software, CAE, see your DB2 documentation.

DB2 supports two-level upward compatibility and one-level downward compatibility. For downward compatibility a Call Level Interface (CLI) version 6.1 application can run in a 5.2 client environment, and for upward compatibility a CLI version 6.1 can run in two future releases of DB2. In a mixed-level environment, DB2 can only use the functions available to the lowest level of the database environment. For example, if a version 6.1 client connects to a version 5.2 server, the client can only use version 5.2 functions. Even though such a mixed environment is supported by DB2, Actuate recommends having the latest version of CLI.

## DB2 environment variables

For UNIX, Windows NT and Windows 2000 you must define the environment variables INSTHOME, DB2INSTANCE, and DB2COMM in order to connect to a DB2 instance.

INSTHOME specifies to the DB2 instance home, DB2INSTANCE specifies the instance name, and DB2COMM specifies the communication protocol. For Actuate software, the DB2COMM should specify TCP/IP.

The environment variables INSTHOME and DB2INSTANCE are defined when you install your DB2 software. You have to set the environment variable DB2COMM. On Windows NT and Windows 2000, use the System applet from the Control panel to check and set the environment variables. On UNIX, use the setenv command to check and set environment variables.

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.

For more information about setting environment variables, see your system documentation.

## HP-UX shared objects for DB2 clients

The shared object file for DB2 supplied with the report server for HP-UX, libdb2.sl, is for DB2 version 5.2. The shared object is in the \$AC\_SERVER\_HOME/lib directory. If the server machine is using a different version of the DB2 client, the user must copy the libdb2.sl shared object from the DB2 client installation directory to the \$AC\_SERVER\_HOME/lib directory.

## Sun shared objects for DB2 clients

The shared object file for DB2 supplied with the Report Server for Sun, libdb2.so, is for DB2 version 5.2. The shared object is in the \$AC\_SERVER\_HOME/lib directory. If the server machine is using a different version of the DB2 client, the user must copy the libdb2.so shared object from the DB2 client installation directory to the \$AC\_SERVER\_HOME/lib directory.

## Checking a connection to a DB2 instance

To check that a connection exists between the report server machine and the DB2 instance, use the command-line utility db2. This utility comes with the DB2 software. The db2 utility is available for both UNIX and Windows NT.

To use the db2 utility, open a command-line window and enter db2 at the command prompt to start the utility. 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 will be 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.

## Installing Factory server software for DB2

Actuate uses DB2 clients V5.2 and newer to connect to DB2 servers V5.2 and newer. On AIX, DB2 is incompatible with ODBC. If your report server runs on AIX, you must replace the Factory server software with the DB2 Factory server. The DB2 Factory server is factsrvr.db2 and is in the directory \$AC\_SERVER\_HOME/operation.

First, back up the existing fctsrvr4 file, then you can either move or copy fctsrvr4. To replace the Factory server software, copy factsrvr.db2 to fctsrvr4 so the report server uses the DB2 Factory server process.

## DB2 version 5 fixpac

The following fixpacs upgrade the DB2 software to DB2 V5.2. For information about the IBM fixpac, contact your IBM sales representative.

- AIX platforms require fixpac FP6\_U459852
- HP-UX (10.20) platforms require fixpac FP6\_U459854
- NT platforms require fixpac FP6\_WR09074
- SunOS platforms require fixpac FP6\_U459853

## DB2 version 6.1 connection issues

The DB2 connection class AcDB2Connection has the property MaximumStringLength to accommodate the increased maximum length of the VARCHAR in DB2 version 6.1. This property is displayed in the AcDB2Connection class Properties dialog box. The default maximum column length Actuate uses with DB2 version 6.1 databases is 8000 characters. Developers can change this property value when designing an Actuate report. Problems might occur when generating an Actuate report if the report uses a DB2 database column with a large column length.

The MaximumStringLength parameter can also be specified in an Actuate configuration file. If specified in a configuration file, MaximumStringLength,

overrides the value specified in the report design. To support more than 8000 characters, change the `MaximumStringLength` parameter in the connection component or in the configuration file. For configuration file information on UNIX systems, see “Specifying a configuration file” in Chapter 2, “Report server administration on UNIX.” For Windows NT and Windows 2000 systems, see “Specifying a configuration file” in Chapter 3, “Report server administration on Windows NT and Windows 2000.”

Actuate does not support CAE versions below 5.2 in Actuate 4.0. If you use CAE version 5.2, apply fixpac version 7 or higher. Without the fixpac, SQL0805N errors are returned whenever schema CLI functions are executed.

The list of fixpacs can be found at <http://www-4.ibm.com/software/data/db2/db2tech/indexsvc.html>

---

## Connecting to Informix databases

The Actuate e.Reporting Server works with Informix databases of version 7.3 and later using Informix Connect 2.40 or Informix Connect 2.01. If you are using an earlier version of Informix client connection software, Informix recommends you upgrade to the latest version. See the Informix web site for more information. To connect with Informix databases, you must:

- Define appropriate environment variables.
- Specify the database environment.
- Supply an account name and password.
- Specify the protocol your site uses.
- Specify the proper client connection.

On UNIX systems, the installation script asks for the Informix information during report server installation and uses this information to create report server startup scripts in the `$AC_SERVER_HOME/bin` directory and to update the system’s startup file. On Solaris systems, this is the `rc.local` file.

### Informix environment variables

If your report server runs on Windows NT or Windows 2000, you must define the system environment variables `INFORMIXDIR` to point to the directory where the Informix product is installed and `INFORMIXSERVER` to specify the name of the database. Defining a user environment variable is not equivalent; report server processes are system processes, and cannot access your user environment.

If your report server runs on UNIX, you must define two environment variables, `INFORMIXDIR` and `INFORMIXSERVER`. As on Windows NT and

Windows 2000, INFORMIXDIR points to the directory where the Informix product is installed and INFORMIXSERVER is the name of the database, obtained from the Informix services file as described in following section.

## **Informix database environment**

The Informix database environment specification includes the database name and the Informix server name in the following format:

```
dbname@servername
```

Both the database name and the server name must match entries in the file named services. On Windows NT and Windows 2000, this file is in <NT>\system32\drivers\etc, where <NT> represents the directory in which Windows NT or Windows 2000 was installed. To modify this file, use the Informix program setnet32. On UNIX, this file is in /etc; to modify the file, use your favorite text editor.

## **Informix account information**

Indicate the account your report server uses to connect to the database by supplying its user name and password.

## **Using UNIX Factory server software with Informix**

If your report server runs on Solaris, AIX, or HP-UX, you might need to change the Informix client version specified in the Actuate report's connection component DLLPath property. For information about specifying the appropriate connection, see "Accessing Informix libraries."

The Actuate Factory server software uses different libraries depending on the Informix client software. The Factory server connection uses either Informix Connect 2.40 with ESQL/C 9.3. or Informix Connect 2.01 with ESQL/C 9.1x.

## **Determining Informix version on UNIX**

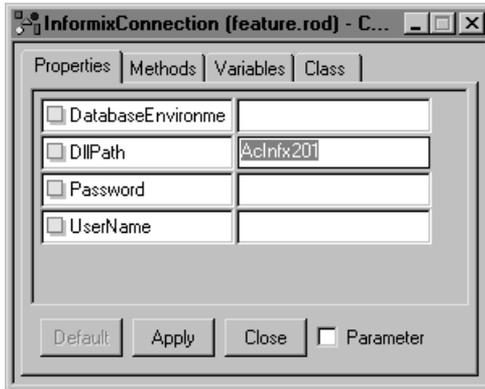
To display Informix version installed on the system, use the Informix ESQL utility with the -V option to display the ESQL version:

```
esql -V
```

## **Accessing Informix libraries**

If your report server runs on Windows servers, you must ensure that the proper dynamic link libraries for Informix are in the path. See the Informix documentation on the libraries used with client applications. The dynamic link library used depends on the Informix client software installed.

For Windows and UNIX, you also need to specify the proper ACINFX\* in the DLLPath property when you create a database connection component. The default library ACINFX240. This DLL is used with Informix client software that uses Informix Connect 2.40 with ESQL/C 9.3. If you installed Informix client software that uses Informix Connect 2.01 with ESQL/C 9.1x, use ACINFX201. ACINFX\* libraries files are part of the Actuate software. The following illustration shows the DLLPath property as ACINFX201.



For an Actuate report executable (ROX) built before 4.1 and using the default Informix connection DLLPath ACINFX720, Informix Connect 2.01 is used.

## Overriding Informix DLLPath

To override the Informix DLLPath property when running a report on a report server using a different Informix client, specify the DLLPath property in the Actuate configuration file. For configuration file information on UNIX systems, see “Specifying a configuration file” in Chapter 2, “Report server administration on UNIX.” For Windows NT and Windows 2000 systems, see “Specifying a configuration file” in Chapter 3, “Report server administration on Windows NT and Windows 2000.”

## Determining Informix version on NT

To find the client version information on Windows use the Informix utility SETNET32. The information in the About SetNet32 page shows Informix version numbers. For example, if the version number is 9.14 - 2.01 the ESQL/C version is 9.14 and you use Actuate DLL ACINFX201.

## Setting the maximum column length

The default maximum column length Actuate uses with Informix databases is 4000 characters. Problems might occur when generating an Actuate report if the report uses an Informix database column with a large column length. You

can use a registry setting on Windows NT, Windows 2000 or an environment variable on UNIX to set a smaller maximum column length used by Actuate.

To change the maximum character length:

- On Windows NT and Windows 2000, change the value of the string value name MaxVarLen. The value name is in the registry key:  
HKEY\_CURRENT\_USER\Software\Actuate\DBMS\Informix
- On UNIX, set the environment variable  
AC\_DBMS\_INFORMIX\_MAXVARLEN

---

## Connecting to MS/SQL databases

The Actuate e.Reporting Server works with MS/SQL databases on Windows NT and Windows 2000.

Actuate release 4.0 supports Microsoft SQL Server 7.0.

To connect with MS/SQL databases, you must:

- Identify the database server.
- Supply an account name and password.
- Indicate the dynamic link library (DLL) the report server uses to establish the connection.

### MS/SQL database server

To indicate the database server, supply the name that is defined with the SQL Client Configuration Utility. This is usually the name of the host computer, but other names are allowed.

### MS/SQL account information

Indicate the account your report server uses to connect to the database by supplying its user name and password.

### Accessing MS/SQL libraries

The Component Editor supplies a default library name of NTWDBLIB, which you should accept. You must also ensure that the PATH environment variable includes the location of the library.

## Testing the MS/SQL connection

If you suspect problems with the connection to your database, use the program `isqlw` to test it. To run this program, click the icon labeled `ISQL_W`, or enter a command of the following form:

```
$ isqlw <server> -u <username> -p <password>
```

For `<server>`, `<username>`, and `<password>`, substitute the information you supplied to the Component Editor.

This test uses the same information and connection mechanism employed by the report server. Thus a positive connection test indicates that the problem lies elsewhere.

## Driver versions

Microsoft SQL Server 7.0 includes an updated SQL ODBC Server version 3 driver that is compliant with the Open Database Connectivity 3.5 specification. The SQL Server ODBC driver fully supports SQL Server 7.0 and 6.x servers.

## Db-library limitations

Version 7.0 of the Db-library API has the same functionality as the 6.5. That is, the Db-library 7.0 has not been upgraded to support all new features. This is a possible limitation for Actuate products that use the Db-library API interface to the database. However, the SQL Server ODBC driver has been upgraded to version 3.7x or higher and can be used to exploit some of the new features of SQL Server 7.0.

Even though `char` and `varchar` datatypes have an increased length of 8000 characters, Actuate can only display 255 characters.

Unicode implementation of text is `ntext`. The Db-library is unable to retrieve data of such type. It generates an error message:

```
ntext message can't be sent to pre 7.0 clients  
because Db-library is considered a pre-7.0 client.
```

## 7.0 Compatibility with 6.5

If a user does not want to use any of the SQL Server 7.0 features in new databases, it is necessary to change the compatibility level setting by executing the `sp_dbcmptlevel` system stored procedure. The compatibility level setting can be set to 6.5 to ensure backward compatibility (by executing the `sp_dbcmptlevel` system stored procedure with a compatibility level setting of 6.5). This should not affect Actuate, as it is transparent to whether the client is 6.5 or 7.0.

---

# Connecting to databases through ODBC

The Actuate e.Reporting Server works with the Merant DataDirect Connect ODBC 3.6 drivers for both UNIX, Windows NT and Windows 2000, and with the Microsoft ODBC 32-bit drivers on NT. The Actuate release 4.0 Merant ODBC 3.6 interface supports ODBC version 3.52. For information on capabilities and limitations of the Merant DataConnect ODBC drivers see Merant's documentation and READ.ME file supplied with the DataDirect products.

To connect to a database through ODBC you must create an ODBC data source that includes the information needed to connect to the database.

When you connect to databases through ODBC, you must supply all the requirements for the target database type, plus a few pieces of information needed by the ODBC layer. For example, you must define the same environment variables, indicate the same library names, and so forth, when connecting to an Oracle database through ODBC as when connecting to the database directly. In addition, all of the applicable conditions in the following list must be met:

- You must have installed ODBC on the host running the report server.
- You must have installed the appropriate client-interaction software on the database host: SQL\*Net for Oracle, Open Client for Sybase.

For Sybase ASE on Windows NT and Windows 2000, a Sybase client is not required. On Solaris, AIX, and HP-UX DBLIB is not required for MS SQL server 7.0.

- The report server host must have the appropriate information in the ODBC initialization file. On UNIX, this file is .odbc.ini; it should be in the home directory of the account running the report server. To modify .odbc.ini, use your favorite text editor. On Windows NT, and Windows 2000 the file is named odbc.ini; it should be in the Windows directory. To modify odbc.ini, use the ODBC Administrator.
- To access ODBC data sources, Actuate e.Reporting Server on NT must be configured to run as a user account, not as the system account.
- The proper version of the report server Factory executable is required for an ODBC connection. For example on UNIX platforms, using the ODBC Informix driver requires the proper version of the Informix report server Factory executable. For information on Factory server executables, see "UNIX factsrvr database compatibility" later in this chapter.

When these conditions are met, you can connect to your database as usual. The Component Editor expects a user name and password for the database, an indication of the database server, and a connection string. To indicate the

database server, use a data-source name from the initialization file. In these examples from a .odbc.ini file showing a pair of Sales data sources, the data-source names are enclosed in brackets:

```
[Sales1]
Driver=/usr2/local/qe-odbc/odbc/dlls/qesyb08.so
ServerName=sales1srv
Description=Sybase 10
LogonID=
```

```
[Sales2]
Driver=/usr2/local/odbc/drivers/vsorac.so.1
Server=t:sales2srv:orcl
Description=
UID=
```

The connection string includes any optional parameters you want to add for the current session. You can use the connection string to supplement or override the options declared in the ODBC initialization file.

## Setting the maximum column length

The default maximum column length Actuate uses with ODBC databases is 8000 characters. Problems might occur when generating an Actuate report if the report uses an ODBC database column with a large column length. You can use a registry setting on Windows NT, Windows 2000 or an environment variable on UNIX to set a smaller maximum column length used by Actuate.

To change the maximum character length:

- On Windows NT and Windows 2000, change the value of the string value name MaxVarLen. The value name is in the registry key:  
HKEY\_CURRENT\_USER\Software\Actuate\DBMS\Odbc
- On UNIX, set the environment variable AC\_DBMS\_ODBC\_MAXVARLEN.

## Using the PeopleSoft ODBC driver

You create a connection to a PeopleSoft database using a PeopleSoft ODBC data source on Windows NT. In the Developer Workbench, you use a stored procedure data source component and the Stored Procedure Data Source Builder with the PeopleSoft ODBC connection.

PeopleSoft supports either a three-tier or two-tier architecture to connect to databases. The three-tier architecture uses an application server, the two-tier architecture does not. For more information about the PeopleSoft architectures see the PeopleSoft documentation.

On the client machine, the PeopleSoft three-tier architecture requires only a data source using the PeopleSoft ODBC data source.

On the client machine, the Peoplesoft two-tier architecture, requires two data sources: one data source using the PeopleSoft driver, and another data source using the native database ODBC driver. For example, if your PeopleSoft software uses a Microsoft SQL Server database, you need to create two data sources, a PeopleSoft ODBC data source and a Microsoft SQL Server data source.

Using either the two-tier or three-tier architecture, when you create an Actuate report that uses a PeopleSoft database, you only need to specify the PeopleSoft data source in the report.

To use the PeopleSoft ODBC database driver with Actuate software you need to:

- Install the Microsoft ODBC 3.0 Administrator.
- Install and configure the PeopleSoft software using the PeopleSoft Configuration Manager PSCFG.EXE. For more information about using the Configuration Manager, see your PeopleSoft documentation.
  - Setup the machine as a PeopleSoft client machine using the PeopleSoft Configuration Manager.
  - Install the PeopleSoft ODBC driver using the PeopleSoft Configuration Manager dialog box Client Setup tab.
- Set up an ODBC data source for the PeopleSoft ODBC driver using the ODBC Administrator ODBCAD32.EXE.
- If you are using a two-tier configuration, set up an ODBC data source for the database containing the data using the ODBC Administrator ODBCAD32.EXE.

For more information about support for stored procedures, see “Stored procedures,” earlier in this chapter.

## Using Red Brick ODBC drivers

The Actuate e.Reporting Server uses ODBC to connect to Red Brick database version 5.1.5. To connect with Red Brick databases:

- On Windows NT:
  - Install the Red Brick ODBC dynamic load libraries (DLL).
  - Configure the ODBC DSN.
- On UNIX:
  - Define appropriate environment variables.
  - Specify the database environment.
  - Install the proper Factory server software.

## Windows NT Red Brick configuration

If your report server runs on Windows NT, you must ensure that the dynamic link libraries for Red Brick version 5.1.5 ODBC drivers are installed. Use the Windows ODBC32 Administrator to verify the version number and to specify an ODBC DSN. In the ODBC configuration Set Advanced Options dialog, verify the proper server name and port number.

The Red Brick ODBC DLLs are supplied by Red Brick and are installed using the Red Brick software.

## UNIX Red Brick configuration

When you configure the report server on a UNIX system to connect to a Red Brick database you must, define Red Brick environment variables, specify the database environment, and install the proper Factory server software.

### Red Brick environment variables

If your report server runs on UNIX, you must define two UNIX environment variables, `RB_PATH` defines the path to the Red Brick software and `RB_CONFIG` defines the path to the Red Brick configuration information. These environment variables must be set in the account that runs the report server.

### Red Brick database environment

On UNIX systems verify that the information for the Red Brick DSN in the `.odbc.ini` file is correct. The following example is a sample Red Brick DSN section from an `.odbc.ini` file:

```
[RBDSN]
SERVER=bigmachine:5050
RB_CONFIG=/net/testmachine/sun6/testusr/redbrick
DATABASE=redbrdata
UID=system
PWD=manager
```

On Sun machines, from `AC_SERVER_HOME/lib` create a symbolic link between `ODBC32` and `$RB_PATH/lib/librbodbc.so`.

On HP-UX machines, from `AC_SERVER_HOME/lib` create a symbolic link between `ODBC32` and `$RB_PATH/lib/librbodbc.sl`.

### Installing Factory server software for Red Brick

First, back up the existing `fctsrvr4` file, then either move or copy the Red Brick `fctsrvr4`. For AIX, HP-UX, and Solaris platforms, rename `factsrvr.redbrick` to `fctsrvr4`.

## Checking the connection to Red Brick databases

On UNIX, to check the connection to a Red Brick database use the utility `rb_client` from a command line window:

```
$> rb_client
```

The utility prompts for log in information: a DSN, User Name, and password. After entering logging into the database you can enter a simple SQL query to check the connection between the client and the database.

On NT, use the Red Brick utilities `RISQL.EXE` or `RBWPING.EXE` to test connectivity to the database server.

---

## Connecting to Oracle databases

The Actuate e.Reporting Server works with Oracle databases of version 7.0 and later. The Actuate interface to Oracle 8 clients supports stored procedures. From the Stored Procedure Data Source Builder in the Developer Workbench, developers can use Oracle 8 server stored procedures to retrieve data to generate reports. For more information about stored procedures, see “Stored procedures,” earlier in this chapter.

To connect with Oracle databases, you must:

- Install the proper software to connect to the Oracle client.
- Supply a connection string.
- Define the appropriate environment variables.
- Ensure that a listener process is running on the database host.

## Oracle java stored procedures

Oracle does not support returning a result set from java stored procedures in Oracle8i release 8.1.5.

## Installing Actuate libraries for Oracle clients

Actuate supports Oracle 7 and Oracle 8 clients that connect to Oracle 7 or Oracle 8 servers. The DLL path property of the connection component uses the Oracle client DLL name `OCIW32.DLL`. To connect to Oracle servers, Actuate supplies libraries for the two versions of Oracle client. The following section describes how to use the Actuate client libraries with Oracle 7 and Oracle 8 client software.

## Using Actuate clients and Oracle clients

For the Actuate client products, Actuate supplies two shared libraries, ACORCL.DLL for Oracle 8 and ACORCL7.DLL for Oracle 7. The default Oracle version supported is Oracle 8 client. Actuate client products use ACORCL.DLL to connect to an Oracle server. If the client machine uses the Oracle 7 client, rename ACORCL.DLL to ACORCL8.DLL, and rename ACORCL7.DLL to ACORCL.DLL.

The Oracle DLLs are found in the \operation subdirectory of the Actuate installation.

## Using the Actuate e.Reporting Server and Oracle clients

For the Actuate e.Reporting Server, Actuate supplies two shared libraries, one for Oracle 7 client and the other for Oracle 8 client. For each of the platforms, the default Oracle version supported is Oracle 8 client. If the report server machine uses the Oracle 7 client software to connect with Oracle databases, you need to rename the Oracle shared library.

### Windows NT DLLs for Oracle clients

There are two DLLs supplied with the Actuate e.Reporting Server for Windows NT, ACORCL.DLL for Oracle 8 and ACORCL7.DLL for Oracle 7. The DLLs are in AC\_SERVER\_HOME/operation folder. The Actuate report server Factory process uses ACORCL.DLL to connect to an Oracle server. If the report server machine uses the Oracle 7 client, rename ACORCL.DLL to ACORACL8.DLL, and rename ACORCL7.DLL to ACORCL.DLL before starting the report server processes.

The Report Serve Oracle DLLs are different from the one used in the client products. You cannot mix the Windows NT client and server DLLs.

### Sun shared objects for Oracle clients

There are two shared object files supplied with Actuate e.Reporting Server for Sun, libacorcl.so for Oracle 8 and libacorcl7.so for Oracle 7. The shared objects are in the \$AC\_SERVER\_HOME/lib directory. The Actuate report server Factory uses libacorcl.so. If the server machine uses the Oracle 7 client, rename libacorcl.so to libacorcl8.so, and rename libacorcl7.so to libacorcl.so.

### HP-UX shared objects for Oracle clients

There are two shared object files supplied with Actuate e.Reporting Server for HP-UX, libacorcl.sl for Oracle 8 and libacorcl7.sl for Oracle 7. The shared objects are in the \$AC\_SERVER\_HOME/lib directory. The Actuate report server Factory process uses libacorcl.sl. If the server machine uses the Oracle 7 client, rename libacorcl.sl to libacorcl8.sl, and rename libacorcl7.sl to libacorcl.sl.

## AIX shared objects for Oracle clients

There are two shared object files supplied with Actuate e.Reporting Server for AIX, `libacorcl_share.a` for Oracle 8 and `libacorcl7_share.a` for Oracle 7. The shared objects are in the `$AC_SERVER_HOME/lib` directory. The Actuate report server Factory process uses `libacorcl_share.sl`. If the server machine uses the Oracle 7 client, rename `libacorcl_share.a` to `libacorcl8_share.a`, and rename `libacorcl7_share.a` to `libacorcl_share.a`.

## Oracle connection strings

The report server connects to Oracle databases using SQL\*Net version 1 or version 2. The information it needs to make the connection, the connection string, takes different forms depending on the version of SQL\*Net you use.

### SQL\*Net version 1

For version 1, the connection string consists of three fields separated by colons:

```
network-prefix:host:SID
```

The network-prefix field identifies the type of network connection. For Oracle databases on UNIX platforms, the network-prefix is `t` (for TCP/IP). For databases on NT, network-prefix can be `t` or `p` (for named Pipe).

The host field identifies the network node running the database you want to reach.

The system identifier, or SID, is a unique identifier used by Oracle processes to distinguish an individual database. This identifier is needed because a single host can run multiple databases concurrently.

### SQL\*Net version 2

For version 2, the connection string is a symbolic name that serves as an index into the file `$ORACLE_HOME/network/admin/tnsnames.ora`. The symbolic name provides a means of grouping properties such as connection type and database host under a convenient identifier. Any name defined in `tnsnames.ora` is a legal connection string.

## Oracle environment variables

For Oracle databases on UNIX platforms, the account running report server 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 report server processes. For more information about the report server scripts, see “Starting and stopping a report server” in Chapter 2, “Report server administration on UNIX.”

For Oracle databases on NT, you must also ensure that the definition of the environment variable PATH indicates the location of the dynamic link library (DLL file) that selects the proper database. For version 7.0, this library is named ora7nt.dll, and for version 7.1 its name is orant71.dll. If your database is version 7.2 or later, you should use ociw32.dll because this library finds the latest available SQL\*Net library and uses it to connect to the database.

## Oracle listener processes

The report server interacts with an Oracle database through the Oracle listener process. The listener process is managed by a controller process, which is called tcpctl for SQL\*Net version 1 and lsnrctl for version 2. Both versions of the controller process have arguments named start, stop, and status, which respectively start, stop, and report the current status of the listener process. If your report server is having trouble communicating with the Oracle database, the link to the listener process might have failed. You can frequently solve such problems by stopping the listener process, if it is still running, and starting it again.

## Testing the Oracle connection

If you suspect problems with the connection to your database, use the Oracle program sqlplus to test it. Enter a command of the following form:

```
$ sqlplus <username>/<password>@<connection-string>
```

For <username>, <password>, and <connection-string>, substitute the information you supplied to the Component Editor. Use the appropriate connection string based on your version of SQL\*Net.

This test uses the same information and connection mechanism employed by the report server. Thus a positive connection test indicates that the problem lies elsewhere.

## Oracle 8 datatype support

The following table lists Oracle 8 datatypes supported by Actuate.

Oracle 8 Datatype	Notes
VARCHAR	This datatype is available in older version of Oracle. In the newer versions, all the VARCHAR columns are automatically changed to VARCHAR2
ROWID	ROWID must be bound to Actuate Basic string datatype. With Oracle 8 databases, Actuate support restricted ROWID only. Actuate does not support extended ROWID.
NCHAR	This is a new datatype available in Oracle 8
NVARCHAR2	This is a new datatype available in Oracle 8

## Setting the maximum column length

The default maximum column length Actuate uses with Oracle databases is 4000 characters. Problems might occur when generating an Actuate report if the report uses an Oracle database column with a large column length. You can use a registry setting on NT or an environment variable on UNIX to set a smaller maximum column length used by Actuate.

To change the maximum character length:

- On Windows NT, change value of the string value name MaxVarLen. The value name is in the registry key:  
HKEY\_CURRENT\_USER\Software\Actuate\DBMS\Oracle
- On UNIX, set the environment variable  
AC\_DBMS\_ORACLE\_MAXVARLEN.

---

## Connecting to Progress 8 databases

The following two sections describe the system requirements for using Actuate software and the Progress 8 database interface.

The Progress documentation also contains information about connecting to Progress databases using Actuate software.

## Actuate software

Actuate software release 3.0 and higher can use the Progress database interface.

### Setting the maximum column length

The default maximum column length Actuate uses with Progress databases is 4000 characters. Problems might occur when generating an Actuate report if the report uses an Informix database column with a large column length. You can use a registry setting on NT or an environment variable on UNIX to set a smaller maximum column length used by Actuate.

To change the maximum character length:

- On Windows NT, change value of the string value name MaxVarLen. The value name is in the registry key:  
`HKEY_CURRENT_USER\Software\Actuate\DBMS\Progress`
- On UNIX, set the environment variable  
`AC_DBMS_PROGRESS_MAXVARLEN`.

## Progress software

You need this software to connect to Progress databases.

- Progress Client software version 8.2. The Actuate interface with the Progress interface connects to Progress server software version 8.2 and higher.
- For Windows 95/98 or NT, ESQ32.DLL. The default directory for ESQ32.DLL is C:\DLC\BIN. The Actuate interface to Progress databases requires ESQ32.DLL on the client computer where you generate Actuate reports that access a Progress database.

### Preparing the Progress Client environment

Follow these steps to prepare the Progress Client environment before attempting to connect to a Progress database from the Actuate software.

#### Preparing the environment on Windows 95/98 or NT machines

- 1 Set the following environment variables:  
`PROCFG=C:\DLC\PROGRESS.CFG`  
`PROMSGS=C:\DLC\PROMSGS`  
`PROSTARTUP=C:\DLC\STARTUP.PF`  
`DLC=C:\DLC`

Add the directories C:\DLC;C:\DLC\BIN to the system's PATH statement.

On Windows NT, make these changes with the System applet in the Control Panel. On Windows 95/98, make the changes in the AUTOEXEC.BAT file.

- 2 Ensure that the file ESQ32.DLL is installed on the client machine.
- 3 Define at least two TCP/IP ports on both the client and server machines, using the exact same syntax and values on both machines.

On the client machine:

- Add a TCP entry to the SERVICES file for the Progress Open Interface.
- Add a TCP entry to the SERVICES file for the Progress Client.

On the server machine:

- Add a TCP entry to the SERVICES file for the Progress OID.
- Add a TCP entry to the SERVICES file for the Progress Client.

The following are sample entries from a SERVICES file:

```
progoid 4096/tcp
sports 4098/tcp
```

The SERVICES file is an ASCII file located in /etc/ on Unix systems or \<WindowsDir>\System32\Drivers\Etc\ on Windows 95/98 or NT systems.

Contact your Progress database administrator for more information about defining TCP/IP ports and setting environment variables.

- 4 Test the client connection to the Progress database before attempting to connect from Actuate software.

## Preparing the environment on UNIX machines

On UNIX machines, you can use Progress Client software with the Actuate e.Reporting Server to access Progress databases from the report server.

After you have installed the Progress Client software and e.Reporting Server on your UNIX system, you need to perform these steps.

- 1 Set environment variables DLC, PROTERMCAP, PROMSGS, and PROCFG. For example:

```
setenv DLC /net/myserver/export/dbms/progress
setenv PROTERMCAP $DLC/protermcap
setenv PROMSGS $DLC/promsgs
setenv PROCFG$DLC/progress.cfg
```

Add the directories \$DLC and \$DLC/bin to the system's PATH environment variable.

- 2 Create entries in the SERVICES file on the client and server machines. See step 3 in the previous section.

Contact your Progress database administrator for more information about defining TCP/IP ports in your SERVICES file and setting environment variables.

## Progress self-service client connection

If you do not have Client Networking installed on your NT Server, you must specify a connection as a self-service client. In the Developer Workbench DatabaseList property, do not use the -S, -N, and -H parameters. For example, to connect to the sports database on your machine as a self-server client, enter this value as the DatabaseList property:

```
-db sports
```

Do not include the path to the sports database.

The StartUpParameters property remains the same.

## Starting a Progress 8 database and the Progress Open Interface Broker

When you start a Progress database and OIB, you specify startup parameters. On Windows NT and Windows 2000 you must also set environment variables. For information about the NT environment variables, see "Setting Progress environment variables on a database server," later in this chapter. When you start the Progress OIB and you are using Actuate with the Progress database interface, you must specify Small Client behavior with the -SV parameter. If you create reports that generate long Progress statements, you can use the -inp parameter to increase the OIB buffer space.

The command-line parameters are shown below for starting a Progress database server and the Progress OIB.

On UNIX:

```
proserve -db <database-name> -S <db-port> -N <protocol> -H <db-host>  
prooibrk -SV -S <broker-port> -N <protocol> -H <broker-host>
```

On Windows NT:

```
_mprosrv -db <database-name> -S <db-port> -N <protocol> -H <db-host>  
oibrkr32.exe -SV -S <broker-port> -N <protocol> -H <broker-host>
```

These are the parameters used with the Progress database server.

Database parameter	Description
-db <database-name>	Database name
-S <db-port>	Registered TCP socket name. Must be different than <broker-port>
-N TCP	Connection protocol, TCP specifies TCP/IP
-H <db-host>	Database host computer name

These are the parameters used with the Progress OIB.

OIB parameter	Description
-SV	Specifies Small Client behavior. This must be the first parameter for the OIB
-S <broker-port>	Registered TCP socket name. Must be different from <db-port>
-N TCP	Connection protocol, TCP specifies TCP/IP
-inp <size>	Maximum number of characters in a Progress statement. The default is 4096 and the maximum is 32000
-H <broker-host>	OIB host computer name

### Setting Progress environment variables on a database server

You must set the PROOIDRV and PROOIBRK environment variables on a UNIX or Windows NT server machine. For example, using the Progress default directory on Windows NT:

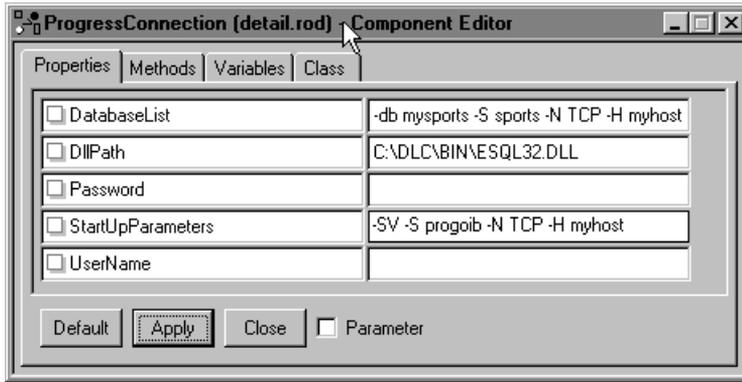
```
PROOIBRK=C:\DLC\BIN\OIBRKR32.EXE
PROOIDRV=C:\DLC\BIN\OIDRVR32.EXE
```

The broker will start without the environment variables, but you will not be able to connect from a client.

## Connecting to a Progress 8 database from a Developer Workbench report

When using the Developer Workbench, you must specify connection parameters for both the database itself and a Progress Open Interface Broker (OIB). You specify the information when you set up a report's connection component. Enter the database connection parameters in the component's DatabaseList property and the OIB connection parameters in the component's StartUpParameters property.

The following is an illustration of an Actuate report Component Editor dialog box for a Progress connection. It shows the Progress database connection parameters in the DatabaseList property and the OIB connection parameters in the StartUpParameters property.



For the Progress database connection properties, you must specify a database list and the startup parameters to connect to a Progress database.

Property	Parameter	Description
DatabaseList	-db	Database name
	<database-name>	
	-S <db-port>	Registered TCP socket name. Must be different from <broker-port>
	-N TCP	Connection protocol, TCP specifies TCP/IP
Dllpath	-H <db-host>	Database host computer name
	ESQL32.DLL	Actuate/Progress interface software
Password		User's Progress database password
StartUpParameters	-SV	Specifies Small Client behavior. This must be the first parameter
	-S <broker-port>	Registered TCP socket name. Must be different from <db-port>
	-N TCP	Connection protocol, TCP specifies TCP/IP
	-H <broker-host>	OIB host computer name
Username		Login name for the Progress database

If Progress Security is active, you can use the following optional DatabaseList parameters to specify a user name and password:

- -U <username> User name registered to Progress Security
- -P <password> User's password

You can log in to multiple databases using different user name and password combinations. For example, you can enter this string for the DatabaseList parameter to log in to the sports and mktg databases:

```
-db sports -H ahost -S asvc -U aname -P apass -db mktg -H bhost -S bsvc -U bname -P bpass
```

Entering a user name or password in the connection dialog box is equivalent to using the -U and -P parameters in the DatabaseList property. The Developer Workbench uses the name and password for each database in the DatabaseList. For example, if you enter:

- DatabaseList: -db mktg -H ahost -S asvc -db sport -H bhost -S bsvc
- Username: fred
- Password: apass

The Developer Workbench creates this database list string:

```
-db mktg -H ahost -S asvc -U fred -P apass -db sport -H bhost -S bsvc -U fred -P apass
```

For a self-service client, do not use the -S, -N, and -H parameters in the DatabaseList property. For example, when OIB is started in same directory as the mktg database on your machine, use this string in the DatabaseList property:

```
-db mktg
```

---

## Connecting to Progress 9.1 databases

Actuate supports Progress 9.1 databases using the Actuate connection component AcProgressSQL92Connection. The Actuate interface to Progress 9.1 supports stored procedures. From the Stored Procedure Data Source Builder in the Developer Workbench, developers can use Progress 9.1 stored procedures to retrieve data to generate reports. For more information about stored procedures, see "Stored procedures," earlier in this chapter.

Progress 9.1 incorporates many new features. Among the new features are:

- Use of Embedded SQL and SQL-92
- Support for new data types
- Support for stored procedures and triggers

The Actuate interface with Progress 9.1 uses Embedded SQL-92 (ESQL). Progress 9.1 supports stored procedures and triggers. The stored procedures and triggers are implemented in Java and stored inside the database. Stored procedures are executed by the database and the results may be returned to the client in the form of output parameters or in the same manner that query results would be fetched.

Stored triggers are defined on a table or column and automatically executed when the conditions for its execution are met.

Progress 9.1 does not support the HP-UX 10.x platform.

## Actuate software

Actuate software release 4.0 and higher supports the Progress 9.1 database interface and Progress 9.1 stored procedures.

## Progress software

You must install the following Progress software to connect to Progress databases:

- SQL Client Access to connect to a Progress 9.1 database
- Java Development Kit (JDK) for stored procedures and triggers

## Windows environment settings

The following environment variables must be set to connect from Actuate to a Progress 9.1 database:

DLC	Progress 9.1 installation directory
PATH	%DLC%/bin and %DLC%/jre/bin must be added to this variable
JDKHOME	This must be set to the JDK installation directory

Ensure that the following registry key is set to the JDK installation directory:

HKEY\_LOCAL\_MACHINE\Software\PSC\PROGRESS\9.0X\JAVA\JDKHOME

## Actuate connection properties

The Progress 9.1 connection component AcProgressSQL92Connection uses the following properties:

Property	Format	Description
Database	string	Progress database name
Dllpath	string	ACPRG9.DLL
Host	string	database host computer name
FetchRowBufferCount	string	Number of rows to be fetched. Value must be less than 65000.
Password	string	Progress database user's password
ReadOnlyQuery	Boolean	Sets the transaction isolation level. If set to True, the default value, the level is set to READ UNCOMMITTED. Otherwise, the level is set to REPEATABLE READ.
Service or Port	string	Registered TCP socket name or port number that connects to the database
UserName	string	Progress database user name

The values for Host, Service or Port, Database, and FetchRowBufferCount are retained between sessions.

FetchRowBufferCount determines the array size used when fetching data from the database. The FetchRowBufferCount is the same for all query objects, which use the connection.

## Progress 9.1 SQL92 connections

You must use multiple instances of AcProgressSQL92Connection class to log on into multiple databases.

Progress9.x client software permits two types of connections. The connection may either be to a local database where the database files are accessed directly or to a remote database where the database is accessed through a database server and the server directly accesses the database files on behalf of its clients.

If you connect to a local database, only the database name is required. If you connect to a remote database, in addition to the database name, the host name or network address and the service name or port number for the database must be given.

If the database is on the local machine and has a database server running, the database must be accessed using a remote connection.

## Stored procedures and triggers

Actuate supports calling Progress 9.1 stored procedures. Single or multiple result sets, and input, output, or input/output parameters are supported. Actuate can describe the result set as well as parameter information.

You must install JDK to execute Progress Java stored procedures and triggers from the database. For information about using JDK with Progress, see the Progress 9 documentation.

## Using the AcProgressConnection connection

The Progress SQL-89 interface is available and compatible with Progress 9.0 and 9.1. Actuate reports using the Progress connection class, `AcProgressConnection`, use the SQL-89 interface to connect to Progress 9.0 and 9.1 databases. If `AcProgressConnection` is used, and the database version is 9.x database, the following views and tables are excluded:

- The tables not owned by PUB
- Views created in SQL92 engine

To connect Actuate Reports to Progress 9.0 databases you must use the `AcProgressConnection`.

Connecting to a Progress9.x database using `AcProgressConnection` is the same as connecting to Progress8 database. The configuration and setup information for connecting to Progress8 database is still valid for Progress 9. During the Progress9 installation, users must make sure that while installing Progress9 database they also install the SQL89 engine. The Progress client software that connects to a SQL89 engine must also be installed on the report server machine.

For more information about connecting to a Progress 9.x database using a Progress 8 connection, see the Progress documentation.

## UNIX platforms

For all UNIX platforms, the following environment variables must be set to connect from Actuate to a Progress 9.1 database:

DLC	Progress 9 installation directory
PATH	%DLC%/bin and %DLC%/jre/bin must be added to this variable
JDKHOME	This must be set to the JDK directory

On UNIX, Actuate does not use the default Progress password set in the DH\_PASSWD environment variable. Actuate always uses the password entered in the AcProgressSQL92Connection connection component.

On UNIX, Actuate uses the shared library name AcPrg9.so or AcPrg9.sl for the Progress9 interface. The Actuate installation process creates a symbolic link from AcPrg9.so or .sl to the Progress9 library. These are the library names for the UNIX platforms:

SunOS	libacprg9.so
HP-UX	libacprg9.sl
AIX	libacprg9_share.a

For example, on SunOS a symbolic link is created between AcPrg9.so to libacprg9.so. If the link is broken, the system administrator must create a link between AcPrg9.so or AcPrg9.sl. and the appropriate Progress9 library.

### SunOS

The following Progress directories must be added to the LD\_LIBRARY\_PATH environment variable:

```
%DLC/lib;%JDKHOME/lib
```

The Progress 9 shared library libacprg9.so is found in \$AC\_SERVER\_HOME/lib.

### HP-UX

The following Progress directories must be added to the SHLIB\_PATH environment variable:

```
%DLC/lib;%JDKHOME/lib
```

The Progress 9 shared library libacprg9.sl is found in \$AC\_SERVER\_HOME/lib.

## AIX

The following Progress directories must be added to the SHLIB\_PATH environment variable:

```
%DLC/lib;%JDKHOME/lib
```

The Progress 9 shared library libacprg9\_share.a is found in \$AC\_SERVER\_HOME\lib.

---

## Connecting to Sybase databases

The Actuate e.Reporting Server uses the Open Client mechanism to connect to Sybase ASE (Adaptive Server) database versions 11.5 and 11.9.2. To connect with Sybase databases, you must:

- Define the SYBASE environment variable.
- Identify the database server.
- Supply an account name and password.
- Indicate the dynamic link libraries (DLLs) the report server uses to establish the connection (Windows NT databases only).

When you use a Sybase ctlib connection, Actuate supports stored procedures with the stored procedure data source. For more information about support for stored procedures, see “Stored procedures,” earlier in this chapter.

## Sybase environment variables

If your report server runs on Windows NT, you must define the system environment variable SYBASE. Defining a user environment variable is not equivalent; report server processes are system processes and cannot access your user environment.

## Sybase database server

To indicate the database server, supply a name that is defined in the file \$SYBASE/interfaces on UNIX, or \$SYBASE/INI/SQL.INI on NT. To modify this file, use the Sybase utility sybinit on UNIX, or the client configuration program sqledit on NT.

If your report server is running on a UNIX host, the name of the database server is interpreted by UNIX software, which is case-sensitive. Therefore you must use the correct case of characters when entering the name of a UNIX report server host regardless of the computer on which you enter the name. Failure to do so prevents successful connections.

## Sybase account information

Indicate the account your report server uses to connect to the database by supplying its user name and password.

## Accessing Sybase dynamic link libraries

If your report server is running on an NT host, you must name the dynamic libraries it uses to connect to the database. For System 11, the Component Editor expects values for the fields CsDllPath and CtDllPath. You use the default names libcs and libct. You must also ensure that the PATH environment variable includes the location of each library.

## Testing the Sybase connection

If you suspect problems with the connection to your database, use the program isql to test it. Enter a command of the following form:

```
$ isql -s <server> -U <username> -P <password>
```

For <server>, <username>, and <password>, substitute the information you supplied to the Component Editor.

This test uses the same information and connection mechanism employed by the report server. Thus a positive connection test indicates that the problem lies elsewhere.

---

## UNIX factsrvr database compatibility

The UNIX platforms must use different Factory servers based on the database connections. The following chart lists the database connectivity options available with the Actuate Factory servers. For the latest information, see the Actuate Supported Products Matrix on the Actuate web site at [www.actuate.com](http://www.actuate.com).

The Factory servers are in the report server's operation server. The Factory server used by the report server is named fctsrvr4. The database-specific Factory servers are named factsrvr.<database name>. For example, the Factory server for Redbrick is factsrvr.redbrick. Actuate ships and installs the default Factory server. To use a database-specific Factory server, save and move or rename the original fctsrvr4 file, and rename the database-specific Factory server to fctsrvr4. These changes must be done while the report server is not running.

OS and Factory server	DB2	Informix Connect 2.40	Informix Connect 2.01	ODBC Merant	Oracle	Progress	Sybase	Red- brick
AIX								
default factsrvr		x	x	x	x	x	x	
db2	x	x			x	x	x	
redbrick		x			x	x	x	x
HP-UX 10.20 and HP-UX 11								
default factsrvr	x	x	x	x	x	x	x	
redbrick	x	x			x	x	x	x
Solaris								
default factsrvr	x	x	x	x	x	x	x	
redbrick	x	x			x	x	x	x

# Report Encyclopedia utilities

This chapter contains the following topics:

- Overview of the utilities and examples
- ACDEFRAG utility
- ACEXPORT utility
- ACIMPORT utility
- ACPIMPRT utility
- ACTOC utility
- UNIX PostScript font utility
- About the archive driver
- About the Report Server Security Extension (RSSE)

---

## Overview of the utilities and examples

The Actuate e.Reporting Server includes:

- Report Encyclopedia maintenance utilities
- e.Reporting Server archive and RSSE application sample programs

### About the utilities

The Actuate Report Encyclopedia command-line utilities let you easily import and export all or part of the Report Encyclopedia to let you transfer data among Encyclopedias. The utilities also let you create new Encyclopedias and defragment Encyclopedia files. The utilities are compatible with the Encyclopedias created by the Actuate e.Reporting Server version 3.0 and later. To let you migrate from previous versions of the report server, the export utility is compatible with the Encyclopedia from the Actuate e.Reporting Server 3.x and later. The utilities perform these functions:

- Export—copy data from a Report Encyclopedia to a binary export file. This utility lets you transfer the information to another report server or rescue data from a damaged Encyclopedia.
- Import—copy data from an export file into an Encyclopedia.
- Create—generate an export file from the native file system hierarchy.
- View contents—view the contents of an export file.
- Defragment—defragment Report Encyclopedia files.

Additional features:

- Security—because the utilities require physical access to the report server files, only a person with proper access to the Report Encyclopedia's physical files can run the program successfully.
- Reporting—the programs report their progress and outline any exceptional cases encountered. Messages are displayed in the command window.

The Report Encyclopedia utilities are run from the command line. Only one instance of any utility can run against a Report Encyclopedia at any time. The utilities must be run while the report server processes are not running. The report server administrator must stop the report server processes if the report server is running. The administrator should ensure that all currently logged-in users are logged off before shutting down the report server.

The utility programs operate directly on the Report Encyclopedia's physical files that contain the administration information and objects that comprise the Report Encyclopedia. The utilities can run locally on the machine containing

the report server or on a different machine with the same architecture that has physical access to Encyclopedia files.

Since ACEXPORT and ACIMPORT work directly with Encyclopedia files, you should back up your existing Encyclopedia before you use either program.

## Utility programs

The report server utilities are located in the <server-home>\Server\bin directory:

- ACDEFRAG—defragments Report Encyclopedia data files.
- ACEXPORT—copies all or part of the Encyclopedia on a report server machine to an export file. ACEXPORT can also check the physical and logical consistency of a Report Server Encyclopedia.
- ACIMPORT—imports all or part of an export file into an Encyclopedia. You can specify whether to merge or overwrite duplicate Encyclopedia data.
- ACIMPRT—creates an export file from the native file system hierarchy.
- ACTOC—lists the contents of an export file.

The following sections describe each utility and its options.

## About the examples

The e.Reporting Server includes two example programs that demonstrate capabilities of the Release 4 e.Reporting Server:

- An example archive driver that can be used with the Report Encyclopedia auto-archive feature. The archive driver archives Report Encyclopedia files by taking the files from the Report Encyclopedia and placing them in a directory on the machine's native file system. For information about the archive driver, see "About the archive driver," later in this chapter.
- A Report Server Security Extension (RSSE) application sample program that can be used with the advanced e.Reporting Server. The RSSE example is designed to demonstrate how the advanced e.Reporting Server can use external page security functionality with external security information. For information about the RSSE application, see "About the Report Server Security Extension," later in this chapter.

---

## ACDEFRAG utility

The ACDEFRAG utility compacts data files stored in the Actuate report server directories by defragmenting report server .DAT files. Defragmented .DAT files take up less space on your hard disk. ACDEFRAG compacts \*.DAT files in the report server directories \Admin, \Request, \Object\Store, and \Object\System. The HIGHEST.DAT file is not compacted by this utility. As an option, you can specify the directory to defragment.

Before using ACDEFRAG, you must stop the report server processes. Since The ACDEFRAG utility works directly with Encyclopedia files, you should back up your existing Encyclopedia before using it.

To specify the Report Encyclopedia directory, use the -home option.

Options	Description
-home <directory>	Required option. Specifies the report server home directory. The default is the current directory
-all	Defragments the .DAT files in all three directories. When no directory option is specified -all is used.
-admin	Defragments the .DAT files in the request directory.
-objects	Defragments the .DAT files in the object directory.
-requests	Defragments the .DAT files in the request directory.
-help	Displays a summary of the options in the command window

On Windows NT and Windows 2000 specify the path to the Encyclopedia using the -home option. If the directory name contains spaces you must enclose the path name in quotes:

```
acdefrag -home c:\actuate4\server
acdefrag -home "c:\actuate\server v3"
```

On UNIX, the following example specifies the Actuate default report server directory:

```
acdefrag -home "/home/Actuate/AcServer"
```

---

## ACEXPORt utility

ACEXPORt exports Report Encyclopedia data to a binary export file. Information is written in a platform-independent format. Some information that is architecture- or server-dependent, for example, printer information, is ignored, with warning, when imported onto a report server running under a different architecture or environment. The export file contains a version information for the format of the export file and the version of the report server. The export file contains ROX, ROI, and other objects that might contain proprietary or sensitive information.

To let you upgrade from previous versions of the report server, ACEXPORt can export an Encyclopedia from the Actuate e.Reporting Server 2.0x and later. Use ACIMPORT to import the Encyclopedia in the current format.

Since ACEXPORt works directly with Encyclopedia files, you should back up your existing Encyclopedia before you use it.

To specify the Report Encyclopedia directory with ACEXPORt, you can use the `-home` option. If you do not use the `-home` option, ACEXPORt assumes that the current directory contains the directories Admin, Object, Request, and Etc that contain the Report Encyclopedia.

To find the Report Encyclopedia directory on Windows NT and Windows 2000, the Registry entry AC\_SERVER\_HOME on the report server machine points to the Report Encyclopedia directory. On UNIX machines, the environment variable AC\_SERVER\_HOME points to the Report Encyclopedia directory.

The default export file name is EXPORT.ACF. The administrator can specify a different name using the `-output` option.

Use the `-no_output` option to check the consistency of the Report Encyclopedia. The `-no_output` option checks the structure of the Report Encyclopedia but does not create an output file.

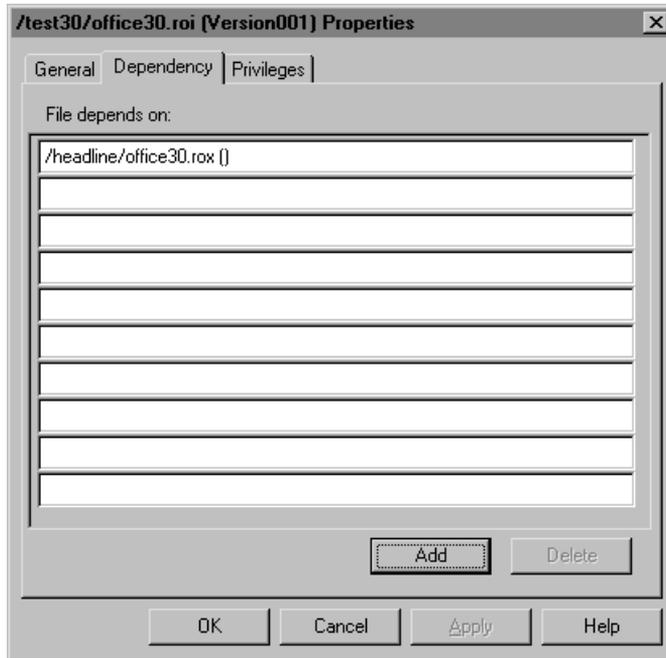
Object Repository—Export allows the administrator to export the contents of the entire Report Server Encyclopedia, or only those objects within one or more specified subdirectories. For each subdirectory, the export may or may not be recursive. Exclusion of subdirectories from the export will also be allowed.

In all cases, information about object ownership is maintained in the export file. Privilege information, users and roles that have various access rights to the objects is also maintained.

The administrator can request that all versions of objects be exported, or that only the latest versions of objects be exported.

Objects also retain the set of users and groups that are to be notified when a related request completes.

Objects that are exported might depend on other objects in the Encyclopedia. For example, an ROI generated from an ROX depends on the ROX. The following illustration of a file's Dependency tab in the Properties dialog box shows that the file is dependent on the file OFFICE30.ROX.



When exporting objects with dependencies, ACEXPORt maintains the dependencies during export. ACIMPORT tries to maintain them during import.

For example, ACEXPORt maintains dependencies when you export the latest reports. If STOCK\_JOHN.ROI;3 is generated from STOCK\_REPORT.ROX;5 and STOCK\_BOB.ROI;4 is generated from STOCK\_REPORT.ROX;6, then both STOCK\_REPORT.ROX;5 and STOCK\_REPORT.ROX;6 are exported even though these may not necessarily be the latest versions of STOCK\_REPORT.ROX.

Even if a file listed in the dependency list resides in a folder that is not being exported, ACEXPORt exports the file and maintains the folder relationship. For example, If STOCK\_BOB.ROX is in the \Reports directory and STOCK\_BOB.ROI is in \Output\Rob and depends on STOCK\_BOB.ROX, ACEXPORt exports both files, even if you export only the files and folders in the \Output directory. If you specify to import only the files in \Output\Bob from the export file, ACIMPORT automatically imports STOCK\_BOB.ROX

and places the .ROX file in the directory \Reports and creates the folder if necessary.

**Groups, Roles, and Users**—The program allows the administrator to export all roles and users or no roles and users. The program also allows the administrator to export all groups or no groups.

**Printers**—The administrator can specify that all or no printers be exported. There may be incompatibilities between Windows NT, Windows 2000 and UNIX printer information, or printers available on the source report server may not be available in a target report server. The utilities display any printer incompatibilities as messages in the command window.

**Process Groups**—The administrator can export all or no process groups.

**ReportCast Channels**—The administrator can export all or no channels.

**Requests**—No active request information will be exported. Completed and scheduled report generation requests related to ROX and ROI objects can be exported.

This example uses the `-no_output` option to check the consistency of the Report Encyclopedia:

```
acexport -all -home c:\actuate\server -no_output
```

This example exports all of the Report Encyclopedia in the directory D:\Marketing\Acserver. The export file is named DAILY.ACF and is placed in C:\Backup:

```
acexport -all -home d:\marketing\acserver -output c:\backup\daily.acf
```

This command exports all the administrative information and just the latest version of items in the Encyclopedia. The export file is named CLEANUP.BAK:

```
acexport -eadmin -latest -ref -output cleanup.bak
```

This example exports users, roles, groups and channels to the default export file EXPORT.ACF:

```
acexport -eusers -egroups -echannels
```

The following table lists the ACEXPORT options.

Export option	Description
GENERAL	
-all	Exports all information from the server. Equivalent to '-ref / -eadmin -erequests'
-home <directory>	Specifies the report server home directory. The default is the current directory

<b>Export option</b>	<b>Description</b>
-check	Checks the Report Encyclopedia database for consistency and errors. Option cannot be used with the -ref, -ef, -bf, or -rbf, options. When -check is used, -no_output is turned on by default
-help	Displays a summary of the options in the command window
<b>OBJECTS</b>	
-ef <folder1> <folder2> ...	Export objects in given folders
-ref <folder1> <folder2> ...	Recursively export objects in given folders, such as export items in subfolders
-bf <folder1> <folder2> ...	Block (exclude) objects from given folders from export
-rbf <folder1> <folder2> ...	Recursively block (exclude) objects from given folder from export
-type <file_type1> <file_type2> ...	Export objects of given file types. Valid file types are ROX and ROI. By default, items of both types are exported.  If any exported file depends on another file, the other file is exported regardless of its file type.
-latest	Export only the most recent version of each exported file. By default, all versions of files are exported
<b>ADMINISTRATION INFORMATION</b>	
-eadmin	Export all administration information. Equivalent to -eusers -egroups -eprinters -epgroups -echannels
-echannels	Export all channels
-eusers	Export all users and roles
-egroups	Export all groups
-eprinters	Export all printers
-epgroups	Export all process groups
<b>REQUEST INFORMATION</b>	
-erequests	Export all scheduled and completed requests associated with exported objects

Export option	Description
OUTPUT	
-output <file name>	Specifies the export file. The default file is EXPORT.ACF
-no_output	Do not generate an output file. Use this option to check the Report Encyclopedia structure without creating an export file

Privilege and notification information is maintained for each exported object.

---

## ACIMPORT utility

ACIMPORT imports Report Encyclopedia administration information such as user and role information, and Encyclopedia objects such as reports from a valid Report Encyclopedia export file. ACPIMPORT is release specific and is used with the same version Encyclopedias. For example, ACPIMPORT shipped with Actuate Release 4 creates Release 4 Report Encyclopedias. ACPIMPORT can import earlier release exported Report Encyclopedia .ACF files. For example, you can use the Actuate Release 4 ACPIMPORT to import a version 3.2.3 .ACF file into a Release 4 Report Encyclopedia. You cannot use newer versions of ACPIMPORT to import data into a older version Encyclopedias. For example, you cannot use the Actuate Release 4 ACPIMPORT to import data into a version 3.2.3 Report Encyclopedia.

If the administrator user of the export Report Encyclopedia has a password, users attempting to import items from the exported Report Encyclopedia must use the -password option and enter the administrator password. If the administrator has no password, or the administrator user was not saved in the export file, the user is not required to specify a password. If the administrator has a password and the -password is not used, a prompt is displayed stating the passwords do not match.

Since ACPIMPORT works directly with Encyclopedia files, you should back up your existing Encyclopedia before you use it.

To specify the Report Encyclopedia directory with ACPIMPORT, you can use the -home option. If you do not use the -home option, ACPIMPORT assumes that the current directory contains the directories Admin, Object, Request and Etc that contain the Report Encyclopedia. If the target directory does not contain a Report Encyclopedia, ACPIMPORT creates the Encyclopedia directories before importing the Encyclopedia.

To find the Report Encyclopedia directory on Windows NT and Windows 2000, the Registry entry AC\_SERVER\_HOME on the report server machine

points to the Report Encyclopedia directory. On UNIX machines, the environment variable `AC_SERVER_HOME` points to the Report Encyclopedia directory.

The administrator can set options to specify what data is imported into a Report Encyclopedia. The administrator has control over information such as user, roles, and Encyclopedia objects such as folders and reports.

**Groups, Roles, and Users**—The administrator can specify that roles, users, and groups be imported to the target server and specify what to do when information conflicts. The options are to either override the existing information or keep the existing information.

Importing information from an export file does not change any Administrator settings or passwords of users in the target Encyclopedia. For example, the Administrator's password and e-mail settings will not change. Even if you specify overriding information using the `-force` option, the Administrator settings and passwords of users are not changed. You can use the `-force` option to change user properties such as the user's e-mail address.

If a user creates a new Encyclopedia using `ACIMPORT`, the Administrator user information in the export file is not used. An Administrator user is created in every new Encyclopedia before `ACIMPORT` adds the data to the new Encyclopedia and `ACIMPORT` does not change the Administrator information when importing data. The Administrator user in the new Encyclopedia has the default settings for a new Encyclopedia, no password and no role or group information.

**ReportCast Channels**—The administrator can import all or no channels.

**Directories, Objects, and Object Versions**—The administrator can import all or part of the data in an export file. The administrator can specify which folders can be imported from the export file. The target Encyclopedia can already contain directories or files whose names coincide with those being imported. The administrator can specify how to resolve these conflicts:

- Explicitly renaming the subdirectories that were exported so that they are imported into a different location in the target server.
- Requesting that imported objects whose names match existing objects replace the existing objects. Note that in this case, existing objects in the target server that depend on the replaced object will have their dependencies broken, and the import program will issue a warning.
- Requesting that imported objects whose names match existing objects become new versions of existing objects.
- Requesting that if any naming conflicts exist between objects in an existing directory and imported objects that the imported objects be placed in a parallel subdirectory. The name of the subdirectory is the name of the

original directory with string `_Copynnn` appended. The `nnn` is a three-digit number.

- Dependencies are maintained where possible. For example, if you exported an entire directory and `STOCK_BOB.ROX` is in the `\Reports` directory and `STOCK_BOB.ROI` is in `\Output\Rob` and depends on `STOCK_BOB.ROX` and you import only the files in `\Output\Bob` from the export file, `ACIMPORT` automatically imports `STOCK_BOB.ROX` and places the `ROX` file in the directory `\Reports` and creates the folder if necessary. For information about exporting and dependencies, see “`ACEXPOR`T utility,” earlier in this chapter.

Privileges are created for imported objects only if the privileges involve roles and users already present in the target server or involve roles and users created during import. The administrator is assigned ownership and privileges if the existing associations cannot be made for the imported object in the target server.

The administrator has the option of turning off the import for all or any subset of the subdirectories exported.

Printers—The administrator can request that exported printers be imported. Printing architectures or environments may not be compatible between the source and target servers. The program warns the administrator when such a problem arises.

Process Groups—The administrator may request that exported process groups be imported.

Requests—The administrator may request that scheduled and completed requests corresponding to imported objects be imported as well.

This example imports all the Encyclopedia from `c:\backup\test.acf` and overwrites any conflicting administration information and versions any duplicate Encyclopedia items. The target Encyclopedia is assumed to be in the current directory:

```
acimport -all -force -version -input c:\backup\test.acf
```

This example imports all report items into the Report Encyclopedia in the directory `d:\marketing\acserver`. Any duplicate files are overwritten. The export file is named `daily.acf` and is in `c:\backup`:

```
acimport -ialf -replace -home d:\marketing\acserver -input c:\backup\daily.acf
```

This command uses export file `newdir.acf`. Imports two directories into an Encyclopedia. The directory `test1` is imported and named `western` and the directory `test2` is imported and renamed `midwest`:

```
acimport -mf test1 western test2 midwest -input newdir.acf
```

The following table lists the ACIMPORT options.

<b>Import option</b>	<b>Description</b>
<b>GENERAL</b>	
-force	When importing administration information, if conflicts exist between existing and imported information, imported information takes precedence over the existing information. By default, existing information takes precedence. Administrator settings are not changed. For example, the Administrator's password and e-mail are not changed.
-all	Import all information. Equivalent to -iallf -iadmin -irequests). Administrator settings are not changed. For example, the Administrator's password and e-mail are not changed.
-home <directory>	Specifies the report server home directory. The default is the current directory.
-input	Specifies the export file to be used to import information.
-password	Specifies the password for the exported Report Encyclopedia administrator user. A prompt is displayed if the administrator user has a password and the -password is not used.
-help	Displays a summary of the options in the command window.
<b>OBJECTS</b>	
-replace	Imported objects replace any existing objects with the same name. If replaced objects have dependents, a warning will be issued, the dependency will be broken, and the object is replaced.
-version	Imported objects become new versions of existing objects with the same name.
-rename	If an imported folder contains any items whose names conflict with existing items, the imported folder will be renamed and the imported objects will be placed in the renamed folder. This is the default among -replace, -version, and -rename.
-iallf	Import all objects from all folders in the export file.

<b>Import option</b>	<b>Description</b>
-if <export_f1> <export_f2> ...	Import objects exported from given folders
-mf <old_dir1> <new_dir1> <old_dir2> <new_dir2> ...	Import objects exported from given folders in source repository, but move (rename) objects to reside in different folders
ADMINISTRATION INFORMATION	
-iadmin	Import all administration information from export
-ichannels	Import channels from the export file
-iusers	Import users and roles from the export file
-igroups	Import groups from the export file
-iprinters	Import printers from the export file
-ipgroups	Import process groups from the export file
REQUEST INFORMATION	
-irequests	Import scheduled and completed requests for all objects imported from the export file

---

## ACPIIMPRT utility

The ACPIIMPRT program is a command-line utility that generates a Report Encyclopedia export file from your native file system hierarchy and input files that contain additional Report Encyclopedia information. The native file system can contain .ROX, .ROI, and other types of Actuate files. The input files specify Report Encyclopedia items to be created such as users, roles, and groups. The input files also contain privilege information for the Report Encyclopedia folders, reports, and other files.

The ACPIIMPRT utility does not import process groups, printers, or request information.

## ACPIMPRT options

The following table lists the ACPIMPRT options.

Option	Description
-toplevel <directory>	Directory in file system that is the root of the hierarchy of items to be packed into an export file
-output <file name>	Specifies the output file. The default file is EXPORT.ACF
-help	Displays a summary of the options in the command window

In the root directory, place a text file named ACINFO.ADM containing administration and notification information about users, roles, and groups to be packed in an export file. Within each subdirectory of the hierarchy you can optionally place a text file named ACINFO.POB indicating properties for the items in the directory such as user, role privileges, and explicit dependencies between files.

The top-level directory can contain folders and can optionally contain an ACINFO.ADM and ACINFO.POB files. The top-level directory cannot contain report documents, report executables, or other files.

In the top-level directory, the ACINFO.POB file can contain only folder definition information: the ACINFO.POB folder cannot contain file definition information.

If you do not use ACINFO.ADM or ACINFO.POB files, the administrator has full privileges assigned to folders and files and no other user has privileges assigned to them.

## ACPIMPRT, ACINFO.ADM and ACINFO.POB file syntax

The ACINFO.ADM and ACINFO.POB files are text files that specify Encyclopedia information using a keyword and value format. There are two types of keywords, primary and secondary. The primary keywords appear as the first word in a line. Secondary keywords and values appear after the primary keyword. You use secondary keywords and values to specify properties for the primary keyword value. For example, this line uses the primary keyword USER and secondary keywords IN\_ROLES and PERSONAL\_FOLDER :

```
USER nathan IN_ROLES support PERSONAL_FOLDER /support/nathan
```

The line creates the user Nathan, specifies that the user Nathan is a member of the role support and specifies that Nathan's personal folder as /support/nathan.

The personal folder information is added only to Nathan's user settings so it appears in the Properties dialog box. When using ACPIMPRT, the personal folder is not automatically created in the Report Encyclopedia and Nathan's privileges on the folder are not automatically changed to visible, read, write delete, execute and grant.

In an ACINFO.POB file, this line sets user privileges for the report executable DEMO.ROX. JSmith has Read and Write privileges and NThompson has Read, Write, Execute, and Delete privileges on the file:

```
FILE demo.rox PRIV_USERS JSmith:RW NThompson:RWXD
```

The keyword format is very flexible. In most cases, the primary keywords can be used in any order in the file and can be used multiple times. Secondary keywords can appear in any order following the primary keyword.

The keyword values are case-sensitive.

Currently, the first keyword in the file is VERSION and can have the value either 3 or 3.0. If a version statement is not present in a file, the file is assumed to conform to the current release of the server.

If a FILE version is specified, it must be the first statement in the file.

For users, roles, groups, and so forth, multiple USER, ROLE, and GROUP statements can be used. It might be convenient if the first part of the file just enumerates all the created items, and subsequent statements for same USER, ROLE, GROUP define interrelationships among them. In specifying interrelationships among users, roles, and groups, we can specify user membership in a role in the USER statement, or also in the ROLE statement. For example:

```
USER nathan IN_ROLES support
ROLE sales
USER nathan IN_ROLES sales
ROLE sales USERS nathan john jake jill
ROLE support
```

For example, to create a user, use the USER keyword with the following syntax:

```
USER user_name
```

To specify a value that contains spaces, enclose the value in quotes:

```
USER "Nathan Smith"
```

To use a keyword as a value, enclose the keyword in quotes. For example, since VERSION is a keyword, the following line is not valid:

```
USER VERSION
```

If you enclose the keyword VERSION in quotes you can use it as a value:

```
USER "VERSION"
```

If you specify a value using a secondary keyword, you must also specify the same value in the appropriate primary keyword. If you do not specify the value using a primary keyword, that value will not be created. Specifying a value in as only a secondary keyword value will not create an object with that value. For example, imagine these lines were the only ones in the ACINFO.ADM:

```
VERSION 3
ROLE sales USERS nathan john jake
USER nathan
USER john
```

When you import the export file, the role sales and the users John and Nathan are created in the target Encyclopedia. If Jake is not a user in the target Encyclopedia, Jake is not created because he is not specified with the keyword USER. If the user Jake exists on the target Encyclopedia, Jake is added to the sales role.

Each primary keyword can accept a set of secondary keywords and values. Any secondary keywords must immediately follow the keyword. The list of secondary keywords cannot contain an intervening primary keyword. This example is valid because the secondary keywords for USER are grouped together and follow the USER keyword:

```
USER nathan
PASSWORD passowrd
EMAIL_ADDRESS nathan@support.com
MAXIMUM_PRIORITY 200
USER john
ROLE sales USERS nathan john jake
```

This example is not a valid set of keywords. The primary keyword ROLE is between the USER secondary keywords PASSWORD and EMAIL\_ADDRESS:

```
USER nathan
PASSWORD passowrd
ROLE sales USERS nathan john jake
EMAIL_ADDRESS nathan@support.com
MAXIMUM_PRIORITY 200
USER john
```

### **ACINFO.ADM primary keywords**

These are the primary keywords that are valid as the first word in a line in an ADM file:

```
VERSION (3 | 3.0)
```

USER name  
 ROLE name  
 CHANNEL name  
 GROUP name

Reserved words. You cannot use these words as values in the ACINFO.ADM files:

PRINTER  
 PROCESS\_GROUP  
 REQEUST

### ACINFO.ADM secondary keywords

Each primary keyword can accept a set of secondary keywords and values. Any secondary keywords and values must immediately follow the primary keyword.

Primary keyword	Secondary keyword	Description
VERSION (3   3.0)		When used as the first keyword in the file, the POB or ADM format version.
USER <name>		Specifies a user and the user's properties.
	PASSWORD <password>	User password.
	EMAIL_ADDRESS <address>	User e-mail address.
	EMAIL_NOTIFICATION	If present, enables e-mail notification when a report is completed.
	SUCCESS_EMAIL_NOTIFICATION	If present, enables e-mail notification when a report is successfully completed.
	FAIL_EMAIL_NOTIFICATION	If present, enables e-mail notification when a report is not completed.
	ATTACH_REPORT_TO_EMAIL	If present with EMAIL_NOTIFICATION or SUCCESS_EMAIL_NOTIFICATION, enables attachment of report with e-mail notification when a report is generated.

Primary keyword	Secondary keyword	Description
	SUCCESS_SERVER_NOTICE <option>	Enables or disables successful report generation Completed folder notification. Options are ON, OFF, or <hours-until-delete>. ON enables server notification. OFF disables notification. <hours-until-delete> enables notification and specifies the number of hours after which the notification is deleted. For a user in a Report Encyclopedia created with ACPIMPRT, the default is disabled Completed folder notification.
	FAIL_SERVER_NOTICE <option>	Enables or disables failed report generation Completed folder notification. Options are ON, OFF, or <hours-until-delete>. ON enables notification. OFF disables notification. <hours-until-delete> enables notification and specifies the number of hours after which the notification is deleted. For a user in a Report Encyclopedia created with ACPIMPRT, the default is disabled Completed folder notification.
	NO_SERVER_NOTICE	If present, disables Completed folder notification when a report is generated.
	MAXIMUM_PRIORITY <number>	Specifies the user's maximum process priority level.
	IN_ROLES <list>	User is in the listed roles. Role names are separated by spaces.
	IN_GROUPS <list>	User is in the listed notification groups. Group names are separated by spaces.
	PERSONAL_FOLDER </path/foldername>	If present, sets the personal folder in the user's Properties dialog box. In the Report encyclopedia, the user's permissions on the specified folder is not updated .

Primary keyword	Secondary keyword	Description
	PRIV_USERS <name>:<privileges> <name>:<privileges> . . .	Privilege template for specified users. Possible privileges are V=visible, S= secure read, R=read, W=write, X=execute, D=delete, and G=grant.
	PRIV_ROLES <name>:<privileges> <name>:<privileges> . . .	Privilege template for specified users. Possible privileges are V=visible, S= secure read, R=read, W=write, X=execute, D=delete, and G=grant.
ROLE <name>		Specifies a role and the roles properties.
	SUPER_ROLES <list>	Child roles. Role names separated by spaces.
	SUB_ROLES <list>	Parent roles. Role names separated by spaces.
	OWNER_USER <name>	Creator of role.
	USERS <list>	Users and roles who are members of this role. Role names separated by spaces.
CHANNEL <name>		Specifies channel name properties.
	SUBSCRIBED <list>	List of subscribed users. User names separated by spaces.
	ROLES <list>	Roles allowed to subscribe to channel. Members of roles can subscribe to channel. Role names separated by spaces.
	COMMENT "<description>"	Description of channel. The description is enclosed in double quotes.
	SMALL_ICON_URL <URL>	Full URL of the small icon used with channels on web pages. If not specified, ReportCast uses a default icon.
	LARGE_ICON_URL <URL>	Full URL of the large icon used with channels on web pages. If not specified, ReportCast uses a default icon.

Primary keyword	Secondary keyword	Description
GROUP <name>	POLLING_INTERVAL <integer>	Specifies, in seconds, how often a web browser automatically refreshes the view of a channel. The default is 180. Zero (0) indicates no automatic refresh.
	EXPIRATION <integer>	Number of days the report is available in the channel. After the expiration date the notice is removed from the channel. The default is 14 days.
	OWNER_USER <name>	Creator of group.
	USERS <list>	Users and roles who are members of the notification group. Role and user names separated by spaces.

### ACINFO.POB primary keywords

These are the primary keywords that are valid as the first word in a line in a POB file.

VERSION  
FOLDER  
FILE

### ACINFO.POB secondary keywords

Each primary keyword can accept a set of secondary keywords and values. Any secondary keywords and values must immediately follow the primary keyword.

Primary keyword	Secondary keyword	Description
FOLDER <name>		Directory name in the native file system
	NAME <name>	Name of the folder in the Encyclopedia
	OWNER_USER <name>	Creator of the folder

<b>Primary keyword</b>	<b>Secondary keyword</b>	<b>Description</b>
	PRIV_USERS <name>:<privileges> <name>:<privileges> . . .	Folder's privilege settings for specified users. Possible privileges are V=visible, S= secure read, R=read, W=write, X=execute, D=delete, and G=grant
	PRIV_ROLES <name>:<privileges>	Folder's privileges setting for specified roles. Possible privileges are V=visible, S= secure read, R=read, W=write, X=execute, D=delete, and G=grant
FILE <name>		Name of file in the native file system
	NAME <file name>	Name of file in the Encyclopedia. Do not specify an extension. Extension is taken from the name specified with FILE keyword
	OWNER_USER <name>	Owner of file
	VERSION <number>	File version number in the Encyclopedia
	PRIV_USERS <name>:<privileges> <name>:<privileges> . . .	Folder's privilege settings for specified users. Possible privileges are V=visible, S= secure read, R=read, W=write, X=execute, D=delete, and G=grant
	NOTIFY_USERS <list>	Users to notify for report generation. User names separated by spaces
	NOTIFY_GROUPS <list>	Groups to notify for report generation. Group names separated by spaces
	DEPENDS_ON <path/file> <ver>	File dependency list. Specify the full file path in the Report Encyclopedia and the file version

---

## ACTOC utility

The ACTOC program lets a user list the contents of an export file. You can list all or part of the contents of the export file.

---

List option	Description
GENERAL	
-all	List contents of entire export file
-input	Specifies the export file to be used
-what	List file format version, source server name and location, parameters for ACEXPORT (or ACPIMPRT), time of export file creation, and other related information
OBJECTS	
-folders	List directories pulled from the source server and those excluded as well
-rfolders	List every directory pulled from the source server, including those that are subfolders exported through recursion
-objects	List all directories and all objects included in the export file
ADMINISTRATION INFORMATION	
-admin	List all administration information
REQUEST INFORMATION	
-requests	List all requests associated with objects listed when running ACTOC

---

---

## UNIX PostScript font utility

The Actuate UNIX utility fontutils installs and uninstalls PostScript fonts to the report server. The utility is in the \$AC\_SERVER\_HOME/bin directory. For information about using fonts with the report server, see "Using PostScript and PCL fonts," in Chapter 2, "Report server administration on UNIX."

Before you use the fontutils utility, make sure the following path:

```
$AC_SERVER_HOME/lib
```

is appended to the library path environment variable:

- On SunOS the environment variable is LD\_LIBRARY\_PATH
- On AIX the environment variable is LIBPATH
- On HP-UX the environment variable is SHLIB\_PATH

## Installing a PostScript font

Copy the font's .AFM and .PFB files to UNIX in the \$AC\_SERVER\_HOME/bin directory. Convert the .AFM file to UNIX format using a utility such as dos2unix. Do not convert the .PFB file to UNIX format. Change the permissions to readable by all users.

With the .AFM and .PFB files in the \$AC\_SERVER\_HOME/bin, use fontutils and the following options from a command line:

```
fontutils -CSoftFont -T1 -1"<font name>" -2<font file>.afm -3 <font file>.pfb
```

where font name is the name of the font used in the report. Enclose the font name in double quotes. Font file is the name of the .AFM or .PFB font. For example, to install a font with the name Ollie Small using the .AFM and .PFB files OllieSmall.afm and OllieSmall.pfb:

```
fontutils -CSoftFont -T1 -1"Ollie Small" -2 OllieSmall.afm -3 OllieSmall.pfb
```

## Uninstalling a PostScript font

To uninstall a font use the fontutils utility and the following options from a command line:

```
fontutils -CSoftFont -T2 -1<font name>
```

where font name is the name of the font used in the report. Enclose the font name in double quotes. For example, to remove the font Ollie Small:

```
fontutils -CSoftFont -T2 -1"Ollie Small"
```

## 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 you have converted a font with Fontographer and installed the font using fontutils, go to the font metrics directory:

```
$AC_SERVER_HOME/operation/xprinter/fontmetrics
```

Use the 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 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.

---

## About the archive driver

Using Actuate software, developers can create an archive driver to archive objects in the Report Encyclopedia using external archiving software and hardware. The archive driver software is an interface between the report server and the archiving software and hardware. For information about the archive driver interface and creating an archive driver, see *Integrating Actuate e.Reporting Server* distributed with the Actuate Software Development Kit.

To demonstrate how an archive driver works, the Actuate e.Reporting Server includes an example archive driver in the report server's /examples directory. This example takes files from the Report Encyclopedia and places them in a directory on the machine's native file system. Within the native file system directory, the archive driver creates a folder hierarchy similar to the one in the report encyclopedia and places the file in the hierarchy.

On Windows NT and Windows 2000, the archive driver is ARCHIVE.DLL. On UNIX systems, the archive driver is a shared library.

## Installing the example archive driver

For information about configuring the Report Encyclopedia to use an archive driver and for information about setting auto-archive rule for archiving objects, see Chapter 4, "Managing report server resources," in *Administering the Report Encyclopedia*.

To install the archive driver, you must set AC\_ROOT\_FOLDER, an environment variable on UNIX, Windows NT and on Windows 2000. The value is the directory where the archive driver places objects removed from the Report Encyclopedia.

On Windows NT and Windows 2000, AC\_ROOT\_FOLDER is a system environment variable.

To have the report server use the variables, stop and restart the report server processes.

## How the example archive driver works

When the report server performs auto archiving on the Report Encyclopedia, the example archive driver copies Report Encyclopedia files that are marked for archival before deletion into the folder in the machine's native file system specified in `AC_ROOT_FOLDER`. The archive driver creates the same directory structure for the object in the native file system that is in the Report Encyclopedia.

For example, if the `AC_ROOT_FOLDER` folder is `C:/a1`, and the object path name in the Report Encyclopedia is `/abc/def/xyz`, and the object version is 1, the archive driver constructs a directory path to file `C:/a1/abc/def/xyz.1`.

The example archive driver does not save any Report Encyclopedia file dependency information when archiving the files. However, you can create an archive driver that saves this information using the Actuate Software Development Kit.

## Logging example archive messages

The example archive driver uses the file `ARCHIVE.LOG` as the log file. If a file named `ARCHIVE.LOG` exists in the report server's `/log` directory, the archive driver appends data to it while archiving objects.

If there is no log file, no logging is done. To turn logging on, create a file named `ARCHIVE.LOG`. To turn off logging, either rename or remove the file.

When logging is on, the file is opened and closed for each message. This allows the rename above, allows you to immediately see any messages, and ensures that the most recent message is in the log file in the event of a crash.

---

## About the Report Server Security Extension

With the advanced e.Reporting Server, Actuate supplies the ability to extend report page security with the Report Server Security Extension, RSSE. Using the RSSE and page security, developers can create an RSSE application that associates security IDs in a report's access control list, ACL, to one or more Report Encyclopedia users or roles. For example, a report developer can create an Actuate report that contains the security ID `Demo` in some of the report's ACLs. Using an RSSE application, `Demo` does not have to be a Report Encyclopedia user or role. The RSSE application can take the security ID `Demo` and map it to a set of Report Encyclopedia users or roles. When a user

associated with Demo views the report, that user can see the report pages where the page's ACL contains Demo.

In addition to the ACL provided by RSSE application, the report server always adds the user name and any roles of which the group is a member. So, even if the RSSE application returns an empty string, the ACL used by the report server always contains the Report Encyclopedia name and roles.

Administrators can install and use an RSSE application if they use the advanced e.Reporting Server. For information about the RSSE interface and creating an RSSE application, see *Integrating Actuate e.Reporting Server* distributed with the Actuate Software Development Kit.

The e.Reporting Server includes an example RSSE application designed to demonstrate the use of an RSSE application with page security. While it is an example, it is designed as a subset of a production RSSE application.

## Installing an RSSE application

To install an RSSE application after you create it for Windows NT and Windows 2000, the name of the DLL must be specified by the following registry entry:

```
HKEY_Local_Machine\SOFTWARE\Actuate\Actuate Report Server\4.1\
AC_RSSE_LIBRARY
```

For UNIX, the shared library is the value of the following environment variable:

```
AC_RSSE_LIBRARY
```

Specify the full path if the RSSE application is not in the search path.

## Using an MT-safe RSSE application

By default, the RSSE application is not multithread-safe and all calls to get the user ACL are serialized. However, the serialization can be turned off by creating a registry entry on NT or an environment variable on UNIX and setting the value to Yes or TRUE. The capitalization of the value must match.

On NT, the string registry entry is AC\_RSSE\_MT\_SAFE\_LIBRARY and is in the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Actuate\Actuate Report Server\4.1
```

On UNIX, the environment variable is:

```
AC_RSSE_MT_SAFE_LIBRARY
```

## About the example RSSE application

The e.Reporting Server includes an example RSSE application designed to demonstrate page security functionality that can be used with the advanced e.Reporting Server. The example RSSE application uses a text file `USERSIDS.TXT` that maps Report Encyclopedia user names to other external security IDs. The RSSE application can be used as a basis for a custom RSSE application.

To control the pages available to a user when viewing an Actuate report while report page security is enabled, the report server compares the user ID and all the Report Encyclopedia roles of which the user is a member with an access control list (ACL) from the report document. The report document ACL is the list of Security IDs for a report document's page. The report server gives the Report Encyclopedia user access to only the pages where the user ID or role is also on the report page ACL.

In some cases, an Actuate report uses security IDs that are not a Report Encyclopedia user ID or role. The RSSE application uses the `USERSIDS.TXT` file to translate a Report Encyclopedia user ID or role to one or more security IDs, which the report server uses to check against the Actuate report page ACL when page security is enabled for a report.

For example, you can create a `USERSIDS.TXT` file with the following line:

```
murphy|region10, region12, region13
```

When `murphy` tries to read an Actuate report with page security enabled, the RSSE application returns a list of security IDs that contain `murphy`, `murphy's` roles, `region10`, `region12`, and `region13`. The report server checks the user ID, roles, and RSSE list against the Actuate report page ACL. The report server lets `murphy` view any report page where a security ID returned from the RSSE application matches a report page security ID.

## Using the example RSSE application

To use the sample RSSE, you must:

- Install the advanced e.Reporting Server
- Stop the report server
- Create the `USERSIDS.TXT` file in the Actuate report server `/etc` directory
- Set the `AC_RSSE_LIBRARY` UNIX environment variable or Windows NT and Windows 2000 registry entry to the Actuate report server example RSSE application. For example on Windows NT, the value of the `AC_RSSE_LIBRARY` is:  
`/Actuate/Server/bin/rsse.dll.`
- Restart the server

You can modify the USERSIDS.TXT file without restarting the report server, the RSSE application picks up changes at the next user log on. If you modify the security IDs for a user who is currently logged on to the server, the report server will not pick up the changes until the user logs out of the server and logs back on.

## **About USERSIDS.TXT file**

The example RSSE application uses a file in the report server /etc directory called USERSIDS.TXT. The file has a very simple format for a user name and a list of security IDs separated by commas:

```
userName|sid1, sid2, ... sidn
```

The userName field is the Report Encyclopedia user name. Each user name in the file must be unique, the user name can be specified only once in the file.

The vertical bar separates the user name from the security ID list.

The report page security ACL can contain zero, one, or more security IDs. If there is more than one security ID, then they must be separated by commas. White space before or after the user name, and before or after the security ID list is ignored. The list can contain users that do not actually appear in the Encyclopedia. The information for these users will simply never be used.

The file can contain blank lines and comment lines. Comment lines must use #, the pound sign character, as the first non-space character.

The example RSSE application loads the user list when the report server first needs to access the information, and automatically reloads the list whenever the USERSEDS.TXT file changes on disk.

## **Logging example RSSE application messages**

The example RSSE application also provides simple logging. The log file is called RSSE.LOG and resides in the report server /log directory. To turn on logging, simply create an RSSE.LOG file. To turn off logging, delete or rename the file.

# Index

## Numerics

32-bit drivers 81

## A

-A option (rsmail) 68

AC\_CENTURY\_BREAK setting 28, 54

AC\_CURRENCY\_FORMAT setting 27

AC\_DATE\_MODE setting 27

AC\_DATE\_SEPCH setting 27

AC\_DBMS\_INFORMIX\_MAXVARLEN  
setting 79

AC\_DBMS\_ODBC\_MAXVARLEN setting 82

AC\_DBMS\_ORACLE\_MAXVARLEN setting  
89

AC\_DBMS\_PROGRESS\_MAXVARLEN  
setting 90

AC\_DEAD\_REQUEST\_PURGE\_TIMES  
setting 26, 53

AC\_DELAY\_AFTER\_PROCESS\_CREATED  
setting 20, 45

AC\_DELAY\_DURING\_PROCESS\_STARTUP  
setting 20, 45

AC\_DISK\_SPACE\_FATAL setting 15, 42

AC\_DISK\_SPACE\_WARNING setting 15, 42

AC\_FACTORY\_PAGE\_COUNT\_UPDATE\_I  
NTERVAL setting 21, 46

AC\_HIGH\_QUALITY\_GRAPH setting 14

AC\_IMAGE\_SEARCH\_PATH setting 14, 42

AC\_LONG\_DATE setting 27

AC\_MAX\_FILE\_CACHE\_ENTRIES setting  
24, 50

AC\_OBJECT\_ARCHIVE\_THREAD\_SCHED  
ULE setting 22, 48

AC\_OPERATION\_SERVER\_DEFAULT\_C\_L  
OCALE setting 43

AC\_OPERATION\_SERVERS\_RECYCLE\_CO  
UNT setting 22, 47

AC\_PURGE\_EXPIRED\_SCHEDULED\_REQ  
UEST\_DELAY setting 26, 52

AC\_PURGE\_WINDOW setting 26, 53

AC\_RECYCLE\_OPERATION\_SERVERS  
setting 21, 47

AC\_ROOT\_FOLDER setting 126

AC\_RSSE\_LIBRARY setting 128

AC\_RSSE\_MT\_SAFE\_LIBRARY setting 128

AC\_SERVER\_CONFIG\_FILE setting 14, 41

AC\_SERVER\_HOME setting 19, 44, 107, 111,  
112

AC\_SERVER\_REPLY\_SOCKET\_BASE setting  
31

AC\_SERVER\_REPLY\_SOCKET\_COUNT  
setting 31

AC\_SERVER\_RPC\_SOCKET\_BASE setting  
31, 56, 57

AC\_SERVER\_RPC\_SOCKET\_COUNT setting  
31, 56, 57

AC\_SERVER\_UNIX\_FD\_MAX setting 13

AC\_SHORT\_DATE setting 27

AC\_SKIP\_CHECK\_POINT\_AT\_START\_UP  
setting 21, 46

AC\_TIME\_FORMAT setting 27

AC\_TIME\_SEPCH setting 27

AC\_VIEW\_FILE\_CACHE\_SIZE setting 24, 50  
AC\_VIEW\_FILE\_CACHE\_TIMEOUT setting  
24, 50

AC\_VIEW\_ROX\_CACHE\_SIZE setting 24, 50

AC\_VIEW\_SERVER\_SOCKET\_BASE setting  
32

AC\_VIEW\_SERVER\_SOCKET\_COUNT  
setting 33

AC\_VIEW\_SESSION\_CACHE\_SIZE setting  
24, 49

AC\_VIEW\_SESSION\_CACHE\_TIMEOUT  
setting 24, 49

AC\_VIEWSERVER\_MAX\_TRANSLATOR  
setting 50

AC\_VIEWSERVER\_SEARCH\_TIMEOUT  
setting 23, 49

AC\_VIEWSERVER\_STARTUP\_TIMEOUT  
setting 24, 48

AC\_VS\_CONNECTION\_RETRY setting 23,  
49

AC\_VS\_CSS\_TEMPLATE\_CLASS\_LIMIT  
setting 23, 49

AC\_VS\_LOCATION\_SIZE setting 24, 50

- AC\_VS\_LOCATION\_TTL setting 24, 50
- AC\_VS\_MAX\_THREAD setting 24, 50
- AC\_VS\_MIN\_DATA\_PACKET\_SIZE setting 23, 48
- AC\_VS\_READ\_BUFFER\_SIZE setting 23, 48
- access control lists (ACLs) 127, 129
  - security IDs and 130
- access privileges *See* privileges
- accessing
  - Encyclopedia items 3
  - online help xiii
- accounts
  - See also* user accounts
  - UNIX server-specific 12–15
  - Windows NT server-specific 40–43
- ACCRPDRV.EXE 51
- ACDEFRAG utility 106
  - caution for using 106
- ACEXPORT utility 107–111
  - caution for using 10
  - file dependencies 108
  - getting information 124
  - running on UNIX servers 36
  - running with Windows NT 63
  - usage examples 109
- ACIMPORT utility 111–115
  - caution for using 10
  - creating new Encyclopedia 112
  - file dependencies 113
  - resolving conflicts 112
  - running on UNIX servers 36
  - running with Windows NT 63
  - usage examples 113
- ACINFO.ADM 116
  - adding information to 116
  - keywords listed 118
- ACINFO.POB 116
  - adding information to 116
  - keywords listed 122
- ACINFX201.DLL 78
- ACL *See* access control lists
- ACORACL8.DLL 86
- ACORCL.DLL 86
- ACORCL7.DLL 86
- ACORCL8.DLL 86
- ACPIMPRT utility 115–123
  - ADM keywords listed 119
  - creating directory structure for 116
  - creating text files 116
  - entering values 117
  - getting information 124
  - POB keywords listed 122
- AcPrg9.sl 99
- AcPrg9.so 99
- AcProgressConnection class 98
- AcProgressSQL92Connection class 97
- ACSQRDRV.EXE 51
- ActiveX controls x
- ACTOC utility 124
- Actuate Basic
  - syntax conventions xviii
- Actuate e.Reporting Server
  - supported database connections 73
- Actuate e.Reporting Server *See* report server
- Actuate home page xi
- Actuate LRX ix
- Actuate product summary viii
- adding Postscript printers 16
- addresses (e-mail) 15, 65
  - testing 38, 66
- adjusting checkpoint information 21, 45
- adjusting object aging start intervals
  - UNIX servers 22
  - Windows NT servers 47
- adjusting process timing 21, 47
- adjusting View process
  - UNIX servers 22
  - Windows servers 48
- admin option (ACDEFRAG) 106
- admin option (ACTOC) 124
- admin directory 7
- Administration process 5
- administrative databases 7
  - clearing 35, 61, 62
  - exporting 110
  - importing 115, 116
  - listing export information 124
- Administrator Desktop 9
- administrators 3
  - exporting data 107, 109
  - importing data 112, 113
  - task overview 8
  - UNIX server-specific tasks 11
- ADMINSRVR4.EXE 44

- admserv4 option 33, 34
- Adobe fonts 17
- Advanced e.Reporting Server ix
- Advanced e.Reporting Server 4
- age-based rules 22, 47
- aged reports 4
- aging interval
  - resetting 22, 47
- AIX servers
  - DB2 clients 75
  - installing/uninstalling fonts 125
  - Oracle clients 87
  - Progress9 clients 100
- all option (ACDEFRAG) 106
- all option (ACEXPORT) 109
- all option (ACIMPORT) 114
- all option (ACTOC) 124
- analysis tools (third-party) x
- application programming interfaces x
- applications 2, 4, 5
  - accessing resources from UNIX 12
  - accessing resources from Windows NT 41
  - handling client requests 6
  - locale-specific configurations 26
  - recovering from system failures 33, 59
- archive driver 105, 126–127
- ARCHIVE.DLL 126
- ARCHIVE.LOG 127
- archiving 127
  - disabling 22, 48
  - example program for 105
  - logging messages for 127
  - setting aging interval for 22, 47
- ATTACH\_REPORT\_TO\_EMAIL keyword (ACPIMPRT) 119
- attaching to servers *See* connections
- attachments 38, 66, 119
- auto archiving
  - disabling 22, 48
- auto-archiving
  - driver for 127
  - logging messages for 127
- automatic disk space checking
  - UNIX servers 15
  - Windows NT servers 42
- automating reporting tasks x

## B

- backing up Report Encyclopedia
  - running command-line utilities and 105
  - UNIX systems 36–37
  - Windows NT systems 62–65
- bf option (ACEXPORT) 110
- binary files
  - creating 115
  - exporting to 107
  - listing contents 124
- buffer size 23

## C

- C shells (UNIX servers)
  - adjusting Factory process timing 20
  - adjusting process timing 22
  - decreasing read buffer 23
  - setting home directory 19
  - starting server processes 33
- cache 7
- cache sizes
  - configuring for UNIX 22
  - configuring for Windows 48
- capacity planning 15, 43
- cascading style sheets 23, 49
- case sensitivity (ACPIMPRT keywords) 117
- channel icon 121
- CHANNEL keyword (ACPIMPRT) 121
- channel members 121
- channels ix, 10
  - exporting to 109
  - importing 112
  - removing notices 122
  - roles available for 121
  - UNIX file descriptors for 13
  - updating 122
- check option (ACEXPORT) 110
- checkpoints 21, 45
- child roles 121
- clearing Encyclopedia items
  - UNIX systems 35
  - Windows NT systems 60–62
- client configuration program 100
- Client Networking (NT servers) 92

- client/server applications 2, 4
  - accessing resources from UNIX 12
- client/server applications (*continued*)
  - accessing resources from Windows NT 41
  - handling requests 5, 6
  - recovering from system failures 33, 59
  - starting report server
    - UNIX systems 33–35
    - Windows NT systems 59
- clients 85
- column length
  - DB2 data sources and 75
  - Informix data sources and 78
  - ODBC data sources and 82
  - Oracle data sources and 89
  - Progress data sources and 90
- command-line options
  - backing up and restoring items on UNIX servers 37
  - clearing report items on UNIX servers 35, 36
  - clearing report items on Windows NT servers 61, 62
  - compacting data 106
  - getting server version 20
  - importing data 116
  - listing export file contents 124
  - setting UNIX environments 74
  - starting Progress servers 92
  - starting report server on UNIX platforms 33
  - stopping report server on UNIX platforms 34
  - testing connections 75, 80, 88, 101
  - testing mail notifications 38, 66, 67
- command-line utilities 103
  - ACDEFRAG 106
  - ACEXPRT 107–111
  - ACIMPORT 111–115
  - ACPIMPRT 115–123
  - ACTOC 124
  - caution for using 10
  - home directories 105
  - overview 104–105
  - rsmail 66, 67
- COMMENT keyword (ACPIMPRT) 121
- comments 130
- communication protocols
  - DB2 databases 74
  - Oracle databases 87
  - Progress connections 91, 97
- communication settings
  - NT servers 56–58
  - UNIX servers 31–33
- compacting data 106
- concurrent threads 24, 50
- configuration files 7, 14, 41
- configurations
  - for international applications 26
  - open server on UNIX 25
  - open server on Windows NT 50–52
  - TCP/IP ports 31–33, 56–58
  - UNIX View processes 29–31
  - Windows View processes 54–56
- connection components 72
- connection parameters (Progress tables) 92, 93
- connection strings
  - Informix tables 77
  - ODBC drivers and 81
  - Oracle tables 87
- connections
  - DB2 databases 73–76
  - failing 33, 59
  - Informix databases 76–79
  - MS/SQL databases 79–80
  - ODBC drivers and 72, 81
  - Oracle databases 85–88
  - overview 71, 72
  - PeopleSoft databases 82
  - Progress8 databases 89–95
    - as self-service client 92
  - Progress9 databases 95–100
  - Red Brick databases 83–85
  - Sybase databases 72, 100–101
  - troubleshooting 75
  - UNIX servers 14
- context-sensitive help xiii
- controller processes (Oracle tables) 88
- conversions 28, 53
- crashes (system) 33, 59
- creating data files 21, 46
- creating report-specific online help xvi
- Crystal Reports 51, 52
- ctlib connections 100
- currency formats 27

## D

-D option (rsmail) 68

.DAT files 106

data

backing up on UNIX servers 36

backing up on Windows NT servers 62

compacting/defragmenting 106

exporting 107, 124

importing 111, 115

restoring from UNIX backups 36

restoring from Windows NT backups 62

system-specific 7

data files 21, 46

data formats

UNIX-specific 27, 30

Windows-specific 56

data packets 23

data sets 72

data sources 13, 41

ODBC drivers and 59, 81, 82

stored procedures as 72

Database property 97

database servers *See* report server; servers

DatabaseList property 92, 93

databases 72

administration contents described 7

clearing administration 35, 61, 62

connecting to *See* connections

exporting administration 110

importing administration 115, 116

listing administration export information  
124

ODBC drivers and 81

datatypes 89

date formats 27, 30, 56

date-based rules 22, 47

-db parameter (Progress connections) 93

DB2 databases 73–76

testing connections 74

troubleshooting connections 75

DB2 Factory server 75

db2 program 74

DB2COMM variable 74

DB2INSTANCE variable 74

default locale 29, 55

default values 22, 48

deleting

obsolete requests 26, 53

temporary files 26, 52

DEPENDS\_ON keyword (ACPIMPRT) 123

Developer Workbench viii, 93

DH\_PASSWD variable 99

DHTML reports 4

generating 29, 54

viewing 5, 23, 49

directories 7

setting home for data compaction 106

setting home for exporting 107

setting home for importing 111

setting home for UNIX 19

setting home for Windows NT 44

setting JDK installation 96

setting up for ACPIMPRT 116

disabling auto archiving

UNIX servers 22

Windows NT servers 48

disk space

UNIX servers 15

Windows NT servers 42

DISKCONF.TXT 42

diskconf.txt 15

DISPLAY variable 12

displaying

DHTML reports 5, 23, 49

online documentation xiii

reports 4, 5

DLC variable 90, 91, 96, 99

DLLPath property 78

Dllpath property 97

DLLs *See* dynamic link libraries

documentation x, xi

online xi, xii

syntax conventions xviii

typographical conventions xvii

domains (Windows NT) 68

drivers *See* ODBC drivers

dynamic link libraries

Informix connections and 77

MS/SQL connections and 79

Oracle connections and 85, 88

Progress connections and 90, 99

Red Brick databases and 84

Sybase connections and 101

## E

- e.Report Designer viii
- e.Reporting Server ix
  - See also* report server
  - administrative tasks described 8
  - configuring printers for 16
  - example programs 105
  - features described 3
  - ODBC drivers and 83
  - overview xi, 1–6
  - setting Factory timing 45
  - setting up 40, 91
  - supported database connections 76, 79, 81, 85
- e.Reporting Suite vii
  - product summary viii
- admin option (ACEXPORT) 110
- channels option ACEXPORT) 110
- ef option (ACEXPORT) 110
- egroups option (ACEXPORT) 110
- electronic mail
  - enabling notification for 119
  - UNIX servers 15, 38
  - Windows NT platforms 65–69
- EMAIL\_ADDRESS keyword (ACPIMPRT) 119
- EMAIL\_NOTIFICATION keyword (ACPIMPRT) 119
- Embedded SQL-92 (ESQL) 96
- Encyclopedia *See* Report Encyclopedia
- End User Desktop viii
- environment variables
  - DB2 databases 74
  - defining 73
  - Informix databases 76
  - MS/SQL databases 79
  - Oracle databases 87
  - Progress8 databases 90, 91, 93
  - Progress9 databases 96, 99, 100
  - Sybase databases 100
  - UNIX-specific 19, 20, 26, 27, 107, 112
- epgroups option (ACEXPORT) 110
- eprinters option (ACEXPORT) 110
- erequests option (ACEXPORT) 110
- errors (mail notification) 15
- ESQL utility (Informix) 77
- ESQL32.DLL 90, 91

- etc directory 7
- eusers option (ACEXPORT) 110
- example programs 105
- executing reports 2
- EXPIRATION keyword (ACPIMPRT) 122
- export files *See* binary files; text files
- export utilities 107, 124
  - caution for using 10
- EXPORT.ACF 107, 116
- exporting Encyclopedia items 104, 107, 124

## F

- F1 key xiii
- Factory process variables 20, 21
- Factory processes
  - adjusting timing for UNIX servers 20–21
  - adjusting timing for Windows NT servers 45
  - default locale 43
  - defined 6
- Factory server installations
  - DB2 databases 75
  - Informix databases 77
  - Red Brick databases 84
  - UNIX-specific platforms 101
- factsrvr.db2 file 75
- FAIL\_EMAIL\_NOTIFICATION keyword (ACPIMPRT) 119
- FAIL\_SERVER\_NOTICE keyword (ACPIMPRT) 120
- failed report generation 120
- failures (system) 33, 59
- fctsrvr4 101
- FetchRowBufferCount property 97
- file dependencies 116
  - ACEXPORT 108
  - ACIMPORT 113
  - getting list of 123
- file descriptors (UNIX) 13
- FILE keyword (ACPIMPRT) 117, 123
- files
  - backing up on UNIX servers 36
  - backing up on Windows NT servers 62
  - caching 24, 50
  - compacting data 106
  - compressing UNIX backup 37
  - exporting 107, 124
  - importing 111, 115

- files (*continued*)
  - organizing 3
  - restoring from UNIX backups 36
  - restoring from Windows NT backups 62
  - setting up font 125
- fixpac upgrades 75, 76
- FOLDER keyword (ACPIMPRT) 122
- folders option (ACTOC) 124
- folders 3, 7
- font files 16, 17
  - setting up 125
  - updating 18
- font properties 18
- fonts
  - installing/uninstalling PostScript 16, 124
  - mapping on UNIX servers 18
- fontutils (UNIX) 17, 124
- force option (ACIMPORT) 114
- formats (locale specific)
  - UNIX servers 29
  - Windows NT servers 54
- functions
  - year conversions for 28, 53

## G

- generating reports
  - for UNIX servers 20
  - for Windows NT servers 45
- graph image quality
  - UNIX servers 14
- grepping report server processes 33, 34
- GROUP keyword (ACPIMPRT) 117, 122
- groups
  - See also* process groups
  - exporting 109
  - importing 112, 116, 117

## H

- H parameter (Progress connections) 93
- H option (rsmail) 68
- hardware devices 13, 41
- help option (ACDEFRAG) 106
- help option (ACEXPORT) 110
- help option (ACIMPORT) 114
- help option (ACPIMPRT) 116
- help xiii–xvii

- creating report-specific xvi
- usage overview xiii
- Help menu xiii
- help topics xiii
- home option (ACDEFRAG) 106
- home option (ACEXPORT) 107, 109
- home option (ACIMPORT) 111, 114
- home directories
  - specifying for data compaction 106
  - specifying for exporting 107
  - specifying for importing 111
  - UNIX servers 19
  - Windows NT servers 44
- home folders 7
- Host property 97
- hosts 2
  - database-specific 72
  - ODBC drivers and 81
  - Oracle connections and 87
  - Sybase connections and 100, 101
- HP systems 13, 14
  - DB2 clients 74
  - installing/uninstalling fonts 125
  - Progress9 clients 99
- HTML reports 10

## I

- iadmin option (ACIMPORT) 115
- iallf option (ACIMPORT) 114
- IBM Database 2 databases *See* DB2 databases
- IBM fixpac upgrades 75, 76
- IBM systems 13, 14
- ichannels option (ACIMPORT) 115
- if option (ACIMPORT) 115
- igroups option (ACIMPORT) 115
- image search path
  - UNIX servers 14
  - Windows NT servers 42
- import utilities 111, 115
  - caution for using 10
- importing Encyclopedia items 104, 111, 115
- IN\_GROUPS keyword (ACPIMPRT) 120
- IN\_ROLES keyword (ACPIMPRT) 120
- Informix Connect versions
  - Factory servers used with 77, 78
- Informix databases 76–79
  - getting version installed 78

- Informix factory server 77
- Informix MaxVarLenRegistry entry 79
- INFORMIXDIR variable 76
- INFORMIXSERVER variable 76
- inp parameter (Progress connections) 93
- input option (ACIMPORT) 114
- input option (ACTOC) 124
- input files (ACPIMPRT) 115
- installation scripts (UNIX) 76
- installing
  - archive driver 126
  - Factory server 75, 77, 84, 101
  - online documentation xiii
  - PostScript fonts 124, 125
  - RSSE applications 128
- INSTHOME variable 74
- invalid requests 35, 61
- ipgroups option (ACIMPORT) 115
- iprinters option (ACIMPORT) 115
- irequests option (ACIMPORT) 115
- isql program 101
- isqlw program 80
- iusers option (ACIMPORT) 115

## J

- JDK installation 96
- JDKHOME variable 96, 99

## K

- kerning 125
- keywords (ACPIMPRT) 116
  - ADM primary/secondary listed 118
  - as literal values 117
  - POB primary/secondary listed 122

## L

- L option (rsmail) 68
- LANG environment variable 27
- latest option (ACEXPORT) 110
- libacorcl.sl 86
- libacorcl.so 86
- libacorcl\_share.a 87
- libacorcl\_share.sl 87
- libacorcl7.sl 86
- libacorcl7.so 86
- libacorcl7\_share.a 87

- libacorcl8.sl 86
- libacprg9.sl 99
- libacprg9.so 99
- libacprg9\_share.a 99
- libcs variable 101
- libct variable 101
- libraries
  - Informix connections and 77
  - MS/SQL connections and 79
  - Oracle connections and 85, 88
  - Progress connections and 90, 99
  - Red Brick databases and 84
  - Sybase connections and 101
- listener processes (Oracle tables) 88
- literal values (ACPIMPRT) 117
- load-balancing 72
- localemap file 29, 54
- LocaleName setting 56
- locales (supported) 29, 54
- locale-specific configurations
  - UNIX servers 26–28
  - UNIX View processes 29–31
  - Windows View processes 54–56
- locking information 7
- logging archiving messages 127
- logging RSSE messages 130
- log-in requests 5
- login scripts (Oracle tables) 87
- LRX (Live Report Extension) ix

## M

- M option (rsmail) 67, 68
- mail notification
  - disabling 120
  - enabling 119
  - export information and 111
  - import information and 116
  - UNIX servers 15, 38
  - Windows NT platforms 65–69
- mailbox IDs (Windows NT) 65
- mailinst program 65
- Manuals directory xiii
- mapping fonts on UNIX servers 18
- mapping information 7
- MAXIMUM\_PRIORITY keyword (ACPIMPRT) 120
- MaximumStringLength property 75

- memory 21, 47
- Merant ODBC drivers 81
- messages *See* notification messages
- mf option (ACIMPORT) 115
- Microsoft Exchange 65, 68
- Microsoft Mail 65
- MS/SQL databases 79–80
  - ODBC drivers and 83
  - testing connections 80
- multithreaded applications 5

## N

- N TCP parameter (Progress connections) 93
- NAME keyword (ACPIMPRT) 122, 123
- naming conventions 3
- network administrators *See* administrators
- networks 2, 3
  - printing from 13, 41
- no\_output option (ACEXPORT) 107, 111
  - usage example 109
- NO\_SERVER\_NOTICE keyword (ACPIMPRT) 120
- NobleNet Portmapper (NT servers) 59
- nofiles (UNIX) 13
- notification groups 122
- notification messages 2, 3
  - disabling 120
  - enabling e-mail 119
  - exporting 111
  - importing 116
  - UNIX servers 15, 38
  - Windows NT platforms 65–69
- NOTIFY\_GROUPS keyword (ACPIMPRT) 123
- NOTIFY\_USERS keyword (ACPIMPRT) 123
- NT servers *See* Windows NT server
- NTWDBLIB.DLL 79

## O

- object aging 22, 47
- object repository 107
- object/store directory 7
- object/system directory 7
- objects option (ACDEFRAG) 106
- objects option (ACTOC) 124
- objects 7

- accessing Encyclopedia 3
- clearing 35, 60, 61
- exporting Encyclopedia 104, 107, 124
- importing Encyclopedia 104, 111, 115
- persistent described 5
- storing 7
- obsolete requests 26, 52
- ociw32.dll 88
- ODBC data sources 59, 82
- ODBC drivers 81
  - caution for AIX servers 75
  - PeopleSoft databases 82
  - Windows NT servers 72
- Odbc maxvarlen Registry entry 82
- odbc.ini files 81
- ogress 93
- OLE objects 16, 43
- online documentation xi, xii
  - syntax conventions xviii
  - typographical conventions xvii
- online help xiii–xvii
  - creating report-specific xvi
  - usage overview xiii
- online reports vii
  - See also* reports
- Open Client mechanism 100
- Open Interface Broker (Progress tables) 92, 93
  - startup parameters 93
- open server ix
  - configuring for UNIX 25
  - configuring for Windows NT 50–52
- open server report generation 4
- operation directory 17
- ora7nt.dll 88
- Oracle databases 85–88
  - maximum column length 89
  - ODBC drivers and 81
  - setting up clients 85–87
  - supported datatypes 89
  - testing connections 88
- Oracle listener process 88
- Oracle MaxVarLen Registry entry 89
- ORACLE\_HOME variable 87
- orant71.dll 88
- output option (ACEXPORT) 107, 111
- output option (ACPIMPRT) 116
- output files (ACEXPORT) 107, 111

output files (ACPIMPRT) 116  
OWNER\_USER keyword (ACPIMPRT) 121,  
122, 123

## P

-P option (rsmail) 68  
packet size 23  
page security 4, 105, 129  
parent roles 121  
-password option (ACIMPORT) 114  
PASSWORD keyword (ACPIMPRT) 119  
Password property 97  
passwords 3, 72  
PATH variable 99  
    MS/SQL connections and 79  
    Oracle connections and 88  
    Progress connections and 91, 96  
    Sybase connections and 101  
PCL printers 16  
pclstd.fonts file 16  
    updating 18  
PDF files xiii  
PDF formats 29, 54  
PeopleSoft Configuration Manager 83  
PeopleSoft databases  
    connecting to 82  
performance 2, 23, 49, 72  
Persistent Object process 5  
persistent objects 5  
    default directory 7  
PERSONAL\_FOLDER keyword  
    (ACPIMPRT) 120  
pobserv4 option 33, 34  
POBSRVR4.EXE 44  
POLLING INTERVAL keyword (ACPIMPRT)  
    122  
Port property 97  
PostScript font utility 124–126  
PostScript fonts  
    installing/uninstalling 16, 124  
    mapping on UNIX servers 18  
Postscript Printer Definition files 16  
PostScript printer models 16  
Postscript printers  
    setting up for UNIX servers 16  
-P parameter (Progress connections) 95  
PPD files 16  
primary keywords (ACPIMPRT)  
    ADM files 118  
    as literal values 118  
    overview 116  
    POB files 122  
Print processes 6  
printed documentation x, xi  
    syntax conventions xviii  
    typographical conventions xvii  
printers  
    export information and 109  
    import information and 113  
    UNIX accounts 13  
    Windows NT accounts 41, 59  
printing reports 2  
    from networks 13, 41  
    notification messages and 3  
    UNIX platforms 12, 16  
    Windows NT platforms 41, 59  
    with OLE objects 16, 43  
priority levels 120  
PRIV\_ROLES keyword (ACPIMPRT) 121,  
    123  
PRIV\_USERS keyword (ACPIMPRT) 121, 123  
privileges 2, 3  
    export information and 107  
    import information and 113  
    losing 35, 61  
    UNIX accounts 12, 35  
    Windows NT accounts 41, 61  
privileges templates 121  
process groups  
    exporting 109  
    importing 113  
process priority levels 120  
processes  
    adjusting view 22, 48  
    configuring for Windows NT servers 59  
    described 5  
    greeting for on UNIX servers 33, 34  
    restarting system 21, 46, 47  
    starting on UNIX servers 33  
    starting on Windows NT servers 59  
        automatically 60  
    stopping on UNIX servers 34  
    stopping on Windows NT servers 59  
PROCFG variable 90, 91

- product summary viii
- product updates xi
- profiles (Windows NT) 65
- programming interfaces x
- Progress MaxVarLenRegistry entry 90
- Progress Open Interface Broker 92, 93
  - startup parameters 93
- Progress8 databases
  - connecting to 89–95
    - as self-service clients 92
  - connecting to Progress9 tables with 98
  - defining environment variables 91, 93
  - setting up environment 90
  - startup parameters 92–93
- Progress9 databases
  - accessing dynamic link libraries 99
  - connecting to 95–100
  - defining environment variables 96, 99, 100
  - logging onto multiple 97
  - version compatibility 96
- PROMSGS variable 90, 91
- PROOIBRK variable 93
- PROOIDRV variable 93
- properties 18, 116
- PROSTARTUP variable 90
- PROTERMCAP variable 91
- providing report-specific online help xvi
- psstd.fonts file 16
  - updating 18

## Q

- queries
  - year conversions for 28, 53
- queue
  - clearing on UNIX servers 36
  - clearing on Windows NT servers 61, 62
- queueing requests 2
- quitting report server
  - UNIX platforms 33, 34
  - Windows NT platforms 59

## R

- rb\_client utility (UNIX) 85
- rbf option (ACEXPORT) 110
- rc.local 76
- RDBMS servers 2
- read buffer 23

- read privilege 4
- ReadOnlyQuery property 97
- Red Brick databases
  - checking connections 85
  - connecting to 83–85
  - installing Factory server 84
  - UNIX servers 84
- ref option (ACEXPORT) 110
- Registry editor 42, 43, 44, 45, 46, 53
- Registry entries (Windows NT)
  - 54
    - getting home directory 107, 111
    - setting Factory process default locale 43
    - setting Factory process timing 45
    - setting home directory 44
    - setting scheduled request default timings 52
  - specifying image search path 42
  - TCP/IP ports 56, 57
- release notes xi
- remote servers 12, 40
- rename option (ACIMPORT) 114
- replace option (ACIMPORT) 114
- Report Encyclopedia 2
  - accessing items 3
  - adjusting checkpoint information for 21, 45
  - auto-archiving contents 127
  - backing up 105
    - UNIX systems 36–37
    - Windows NT systems 62–65
  - clearing items 35, 60–62
  - creating with ACIMPORT 112
  - exporting items 104, 107, 124
  - importing items 104, 111, 115
  - items described 7
  - overview 7
  - ReportCast and 4
  - restoring 37, 64
  - securing 4
  - securing contents 127
  - storing items 7
  - updating 21, 46
  - upgrading 9, 107
- Report Encyclopedia utilities *See* command-line utilities
- report files
  - backing up on UNIX servers 36

- report files (*continued*)
  - backing up on Windows NT servers 62
  - caching 24, 50
  - exporting 107, 124
  - importing 111, 115
  - organizing 3
  - restoring from UNIX backups 36
  - restoring from Windows NT backups 62
- report server
  - See also* e.Reporting Server
  - adding soft fonts 16–19
  - connections *See* connections
  - default UNIX settings 27
  - processes described 4
  - running 72
    - AIX servers and 75
    - Solaris servers and 77
  - UNIX platforms 11
    - adjusting Factory process timing 20–21
    - implementing security 12
    - mapping fonts 18
    - printing reports 13, 16
    - restarting 33
    - setting up accounts 12–15
    - starting and stopping 33–35
  - upgrading 10
  - Windows NT platforms
    - adjusting Factory process timing 45
    - implementing security 40
    - setting up accounts 40–43
    - starting and stopping 59
  - Windows platforms 39
- report server API x
- Report Server Security Extension (RSSE) 127–130
  - example program for 105, 129
- Report Server TCP/IP configurations 31–33, 56–58
- ReportCast 4
- ReportCast channels ix, 10
  - exporting to 109
  - importing 112
  - removing notices 122
  - roles available for 121
  - UNIX file descriptors for 13
  - updating 122
- reporting solutions vii
- reports vii, 72
  - clearing 35, 60, 61
  - creating online help for xvi
  - generating for NT servers 45
  - generating for UNIX servers 20
  - printing *See* printing reports
  - running 2
  - storing 7
- repositories 107
- reqserv4 option 33, 34
- reqsrvr.sh 73
- REQSRVR4.EXE 44
- request directory 7
- Request process 5
- request queues 2
- Requester API x
  - requests option (ACDEFRAG) 106
  - requests option (ACTOC) 124
- requests 2, 3, 5
  - caching 24, 50
  - clearing queued 35, 61
  - export files and 109
  - handling 6
  - import files and 113
  - invalid 35, 61
  - setting buffer size 23, 48
  - storing 7
  - system restarts and 21, 47
  - UNIX servers 25
  - Windows NT servers 52
- resources
  - administration tasks for 8
  - UNIX accounts 12
  - Windows NT accounts 41
- response 21, 46
- restarting system processes
  - UNIX servers 21
  - Windows NT servers 46, 47
  - Windows servers 46
- restoring from backups
  - UNIX servers 36
  - Windows NT servers 62
- restoring Report Encyclopedia 37, 64
- resubmitting expired requests
  - UNIX servers 25
  - Windows NT servers 52
- result sets 72

- retry attempts 23
- rfolders option (ACTOC) 124
- ROLE keyword (ACPIMPRT) 117, 121
- roles 107
  - exporting 109
  - importing 112, 116, 117
- ROLES keyword (ACPIMPRT) 121
- root processes and accounts 12
- rsmail utility 66, 67
- RSSE *See* Report Server Security Extension
- RSSE.LOG 130
- running command-line utilities 104
- running reports 2

## S

- S parameter (Progress connections) 93
- sample programs 105
- scheduled requests
  - exporting 109
  - importing 113
  - removing from UNIX servers 25, 26
  - removing from Windows NT servers 52
- schedules 2, 3
  - clearing 35, 61
  - storing 7
- search extension API x
- searches (help topics) xv
- secondary keywords (ACPIMPRT)
  - ADM files 119
  - as literal values 118
  - overview 116
  - POB files 122
- secure read privilege 4
- security 3
  - administrative tasks for 8
  - page-level 4, 105, 129
  - UNIX servers 12
  - Windows NT servers 40
- security IDs
  - associating with ACLs 127
  - entering 130
  - mapping 129
- self-service clients (NT servers) 92
- Sender ID cache 24, 49
- sending mail
  - UNIX accounts 15, 38
  - Windows NT accounts 65–69
- sendmail program 15
- separators (date/time) 27
- serialization 128
- servers ix, 2
  - administrative tasks 8, 11, 39
  - AIX and ODBC drivers 75
  - connecting to *See* connections
  - crashing 33, 59
  - printing reports 41
  - shutting down on UNIX systems 34
  - shutting down on Windows NT systems 59
  - specifying SQL-specific 79
- Service property 97
- Services dialog box (Windows NT) 60
- services file
  - Informix databases and 77
  - Progress databases and 92
  - Progress8 databases and 91
- setenv variable 12, 74
- setnet program 77
- setting Factory process variables 20, 21
- shared libraries 86
- shared object files 86
- shell scripts (UNIX servers)
  - adjusting Factory process timing 20
  - adjusting process timing 22
  - decreasing read buffer 23
  - setting home directory 19
  - starting server processes 33
- shutdown script (UNIX servers) 34
- shutdown\_srvr.sh script 34
- SMALL\_ICON\_URL keyword (ACPIMPRT) 121
- socket port configurations 32
- soft fonts 16
- Software Development Kit x
- Solaris systems
  - Informix connections and 76
  - printing from 13
  - starting server on reboot 14
- SQL Client Configuration Utility 79
- SQL\*Net version-specific connections 87
- sqledit program 100

- sqlplus program 88
- Scribe reports
  - configuring driver for 51
  - printing 51
- SQRWPEXE 51
- starting report server
  - UNIX platforms 33
  - Windows platforms 59
- startup scripts (UNIX servers) 76
- StartUpParameters property 92, 93
- stopping report server
  - UNIX platforms 33, 34
  - Windows NT platforms 59
- store directory 7
- stored procedure component 72
- stored procedures 72, 82, 85, 96
- storing report items 7
- style sheets 23, 49
- SUB\_ROLES keyword (ACPIMPRT) 121
- subdirectories 7
- SUBSCRIBED keyword (ACPIMPRT) 121
- SUCCESS\_EMAIL\_NOTIFICATION
  - keyword (ACPIMPRT) 119
- SUCCESS\_SERVER\_NOTICE keyword (ACPIMPRT) 120
- Sun systems
  - DB2 clients 74
  - installing/uninstalling fonts 125
  - Progress9 clients 99
- SUPER\_ROLES keyword (ACPIMPRT) 121
- SV parameter (Progress connections) 93
- Sybase databases
  - connecting to 100–101
  - ODBC drivers and 72, 81
  - setting up environments 100
  - testing connections 101
- SYBASE variable 100
- sybinit utility 100
- symbolic names (Oracle tables) 87
- syntax conventions (documentation) xviii
- SysLocaleName setting 56
- system administrators *See* administrators
- System applet (Windows servers) 74
- system configuration files 7
- system directory 7
- system environment variables *See* environment variables

- system environments vs. user 73
- system failures 33, 59
- system identifiers (Oracle tables) 87
- system process restarts
  - UNIX servers 21
  - Windows NT servers 46, 47
- system-specific data 7

## T

- TCP/IP ports 31–33, 56–58
- temporary files 7
  - deleting 26, 52
  - UNIX servers 19
  - Windows NT servers 44
- testing connections
  - DB2 databases 74
  - MS/SQL databases 80
  - Oracle databases 88
  - Red Brick databases 85
  - Sybase databases 101
- testing mail notifications
  - UNIX servers 38
  - Windows NT servers 66, 67
- text files 116
  - sending as mail attachments 38, 66
- third-party analysis tools x
- third-party reports 6
- threads 24, 50
- time formats 27
- timeout values
  - configuring for UNIX 22, 23
  - configuring for Windows 48
- tmp directory 7
- tnsnames.ora file 87
- toplevel option (ACPIMPRT) 116
- triggers 96
- turning off RSSE serialization 128
- type option (ACEXPORT) 110
- typographical conventions xvii
  - syntax xviii

## U

- U parameter (Progress connections) 95
- ulimit command (UNIX) 13
- uninstalling PostScript fonts 125
- UNIX file descriptors 13

- UNIX hosts 72, 100
- UNIX servers 11
  - adding Postscript printers 16
  - adjusting Factory process timing 20–21
  - adjusting system restart timing 21
  - administrative tasks described 9
  - attaching to
    - DB2 databases and 73, 74
    - Informix tables and 76, 77
    - ODBC drivers and 81, 84
    - Oracle tables and 87
    - Progress tables and 91, 92, 99
    - Sybase tables and 100
  - checking disk space for 15
  - clearing Report Encyclopedia 35
  - compressing backup files 37
  - configuring 14
  - configuring open server driver 25
  - configuring TCP/IP ports 31–33
  - connecting to 14
  - creating data files 21, 46
  - defining environment variables 27
    - Informix tables and 76
    - Oracle tables and 87
    - Progress tables and 91, 93, 99
  - defragmenting data files 106
  - exporting files 107
  - importing files 112
  - installing archive driver 126
  - installing Factory server 101
  - installing RSSE applications 128
  - installing/uninstalling fonts 124–126
  - locale-specific configurations 26–28
  - locale-specific View processes 29
  - mapping fonts 18
  - printing reports 12, 16
  - removing expired requests 25, 26
  - restoring backups 37
  - running report server 33–35
  - security 12
  - setting aging intervals 22
  - setting home directories 19
  - setting up accounts 12–15
  - setting up DB2 clients 74
  - setting up environments 19, 20, 26, 73
  - setting up Oracle clients 86, 87
  - temporary files 19

- testing mail notifications 38
  - troubleshooting connections 75
- UNIX tar command 37
- unused disk space 15, 42
- updates xi
- updating Report Encyclopedia 21, 46
- upgrades 9, 107
- user accounts 3, 8, 59
  - See also* accounts
  - export information and 107, 109
  - import information and 112, 116, 117
  - validating 5
- user addresses 15, 65
  - testing 38, 66
- user environments vs. system 73
- USER keyword (ACPIMPRT) 117, 119
- user names 72
- UserName property 97
- USERS keyword (ACPIMPRT) 121, 122
- USERSIDS.TXT 129, 130
- utilities *See* command-line utilities

## V

- v option 20
- validation 3, 5
- variables
  - setting Factory process 20, 21
  - setting for View processes 22, 48
- version option (ACIMPORT) 114
- version (report server)
  - displaying on UNIX platforms 19
  - displaying on Windows NT 44
  - upgrading and 10
- VERSION keyword (ACPIMPRT) 117, 119, 123
- view directory 7
- View process files 7
- View process variables 22, 48
- View processes
  - adjusting 22, 48
  - overview 5
  - setting retry attempts 23
  - UNIX locale configurations 29–31
  - Windows locale configurations 54–56
- Viewer viii
- viewing
  - DHTML reports 5, 23, 49

viewing (*continued*)  
  online documentation xiii  
  reports 4, 5  
Virtual Report Distribution 3  
Visigenic ODBC drivers 81

## W

wait times 24  
web browsers ix  
web pages *See* HTML reports  
web site (Actuate) xi  
-what option (ACTOC) 124  
Windows 95 connections 90, 91  
Windows hosts 72, 101  
Windows NT server  
  adjusting Factory process timing 45  
  adjusting system restart timing 46, 47  
  administrative tasks described 9  
  attaching to  
    DB2 databases and 73, 74  
    Informix tables and 77  
    MS/SQL tables and 79  
    ODBC drivers and 72, 81, 82, 84  
    Oracle tables and 87  
    Progress tables and 90, 92  
    Sybase tables and 100  
  backing up Report Encyclopedia 62–65  
  checking disk space for 42  
  clearing report server items 60–62  
  configuring 41, 43  
  configuring open server driver 50  
  configuring processes 59  
  configuring TCP/IP ports 56–58  
  defining environment variables 73  
    Informix tables and 76  
    MS/SQL tables and 79  
    Oracle tables and 88  
    Progress tables and 91, 93, 96  
    Sybase tables and 100  
  defining Registry entries 43, 45, 52, 107, 111

defragmenting data files 106  
exporting files 107  
importing files 111  
installing archive driver 126  
installing RSSE applications 128  
locale-specific View processes 54  
printing reports 43  
removing expired requests 52  
restoring backups 64  
running report server 59–60  
security 40  
setting aging intervals 47  
setting home directories 44  
setting up accounts 40–43, 68  
setting up mail notifications 65–69  
setting up Oracle clients 86  
temporary files 44  
testing mail notifications 66, 67  
troubleshooting connections 75  
Windows server 39  
Winzip utility 62

## X

X Logical Font (XLF) specification 18  
X Logical Font Description 18  
X Windows servers 12  
X Windows systems 12  
XML documents 29, 54  
XML translator 50  
xprinter directory 17

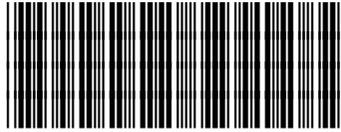
## Y

year conversions 28, 53  
year formats 54

## Z

zip files 62





\*100037163\*