EMC® Smarts®
Service Assurance Manager

Version 9.2

Configuration Guide

P/N 300-999-851
REV 01

EMC²

Copyright © 1996 - 2013 EMC Corporation. All rights reserved. Published in the USA.

Published March, 2013

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC^2, EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (https://support.emc.com).

CONTENTS

Preface

Chapter 1 Overview

Service Assurance Manager components 20
Global Manager 20
Global Console 20
Business Dashboard 20
Adapter Platform 20
EMC Smarts Adapter for UIM/O 21
Business Impact Manager 21
EMC Smarts Notification Module 21
Companion UI 21
Watch4net 21
EMC Smarts Notification Module Cisco Syslog Processing Adapter 22
Information flow between components 23
Scenario 1 23
Scenario 2 24

Chapter 2 Configuration Overview

Types of configuration tasks 28
Creating user profiles 28
Configuring system parameters 29
Managing topology and notifications 29
Automating responses to notifications 29
How to configure the Global Manager 30
Privilege requirements 30
Global Manager Administration Console 30
Modifying configuration files 31
Starting and Configuring EMC Smarts NOTIF 34
Starting services 36
Starting services on Windows 37

Chapter 3 User Configuration for the Global Manager

Configuring a security message 40
Configuring the security message for the Global Console 40
Configuring security message for Business Dashboard viewlet and Web Console 40
User profiles 41
Creating a user profile 41
Deleting a user profile 44
Modifying a user profile 44
About the predefined user profiles 47
User accounts 48
Creating a user account 49
Deleting a user account 49
Creating a saved console 50
Notification lists 52

Notification list parameters ... 53
Creating a notification list ... 54
Modifying a notification list ... 55
Disabling a notification list ... 55
About Default notification lists ... 56
Processing notifications ... 56
Notification polling and Property Change ... 57
Configuring the Maintenance and Business Impact Manager ... 58
Schedule Maintenance option ... 59
Console operations ... 59
Interaction with EMC Smarts username/password privileges ... 59
Default user profiles ... 60
Console operations and groups ... 60

Chapter 4 System Configuration for the Global Manager

Defining Domain Manager parameters ... 66
Data exchange files or underlying domains ... 67
Defining system defaults ... 69
About acknowledging notifications ... 73
Archiving notifications ... 73
Configuring business templates ... 74
Configuring Global Manager Administration Console preferences ... 75
Configuring IPv6 Availability Manager connections ... 77
Examples of specifying the IP version ... 77
Audit logging ... 78
Configuring the Audit Log Adapter ... 79
Starting the Audit Log Adapter ... 80
Starting a new log file ... 80
Filtering the Audit Log ... 81
SAM performance throughput report ... 82
Notification lifecycle auditing ... 84
Configuring weights.conf for OTM ... 85
Hierarchical SAM configuration ... 86

Chapter 5 Configuring Domains

Configuring domains ... 88
Creating a domain ... 88
Modifying domains ... 91
Deleting domains ... 92
Configuring domain groups ... 93
Creating domain groups ... 93
Modifying domain groups ... 94
Deleting domain groups ... 96
Configuring domain types ... 97
Creating domain types ... 98
Modifying domain types ... 99
Deleting domain types ... 101
Managing overlapping elements from separate underlying domains ... 101
How the Global Manager applies tags ... 102
Configuring domain tagging ... 103
Creating domain tags ... 103
Modifying domain tags ... 104
Deleting domain tags ... 105

Reconfiguring the server 106

Chapter 6 Managing Notifications with the Global Manager

Overview of notifications 108
Understanding and managing notifications 109
States of a notification 110
A notification's life cycle 111
Acknowledging and archiving notifications 113
Configuration of acknowledgment and archiving of notifications 113
Notification types and incrementing OccurrenceCount 114
User-defined notification attributes 115
User-defined notification attributes from Adapter Platform or Global Manager 116

Chapter 7 Managing Topology with the Global Manager

Topology synchronization 118
Starting topology synchronization 118
Limitation of topology synchronization 119
Enable or disable topology autosync 120
Enable topology synchronizations at regular intervals 122
Get a list of topology drivers 122
Topology Timeout 123
Ensuring a consistent representation of topology 123
Same device classified differently in separate underlying domains 124
Same device named differently in separate underlying domains 125
Organizing topology with groups 126
General properties of groups 126
Properties of selective groups 127
Creating selective groups 129
Creating hierarchical groups 133
Purging of HierarchicalGroupManager instance 135
Custom icons for map nodes 136
Assigning icons to classes or instances 136
Assigning an icon to a class 137
Assigning an icon to an instance 137
Reassigning the derived icon to a class 138
Reassigning the derived icon to an instance 138

Chapter 8 Tool Configuration for the Global Manager

Types of program tools 140
Additional Service Assurance tools 140
How tools are invoked 141
Invoking server and client tools 141
Invoking automated tools 142
Information recorded to a notification's audit log 142
Security considerations for tools 142
Sample tool scripts 143
Creating tools 145
How data is passed to a tool script 146
Where to save tool scripts 149
Configuring with the Global Manager Administration Console 149
Context and Status Criteria for client and server tools 151

Modifying a tool 154
Running automated tools with sm_adapter 155
Running tools over X Windows 156
Adding domain configurations 156
Importing XML Domain Configurations 157
Importing XML configurations 159

Chapter 9 Escalation Configuration for the Global Manager

Overview of escalation 162
Escalation policy structure 162
Viewing an escalation policy 163
How notifications are scheduled for escalation 165
Effect of Global Manager restart on escalation 165
Developing escalation policies 166
Creating an escalation policy 166
Creating an escalation path 167
Adding escalation levels 168
Modifying escalation policies and paths 169
Enabling and disabling escalation paths 169
Modifying an enabled escalation path 170
Modifying disabled escalation paths or policies 171
Retiring an escalation path 173
Modifying escalation paths by using the Retire option 174
Removing or deleting escalation paths and policies 175
Testing and troubleshooting escalation policies 176
Viewing notifications being escalated 177

Chapter 10 Working with Filters

Overview 180
Building expression filters 180
Layout of the Filter Builder 181
Using the Filter Builder 182
ASL filters 186

Chapter 11 Importing and Exporting Configurations

Overview 190
About the default XML files 190
Running sm_config 193
Importing configurations 195
Defining objects in XML 195
Importing XML files to the Global Manager 197
Exporting configurations 198
Modifying individual repository objects 199
Enabling or disabling repository objects 199
Deleting repository objects 199
Importing tools implemented by EMC Smarts adapters 200

Chapter 12 SAM Configuration Parameters

System default parameters 202
ics-event-processing parameters 205
ics-nl-processing parameters 206
Console parameters 207

trap.d conf parameters 210
Environment variables 210

Appendix A ICIM Classes Used with Matching Criteria

Classes and attributes for matching managed systems 212
Attributes for matching notification properties 215

Appendix B Wildcard Patterns

Types of wildcard patterns 220

Appendix C XML Reference

Sample XML files 224
Structure of an XML document 224
Element 224
Attribute declaration (attribute list) 225
Value 225
Configuration elements 226
ics_config 226
Notification list (nlconfig) 227
Filter (filterconfig) 228
Column heading (columnheading) 230
Tool/Action (actionconfig) 232
Program name (program_name) 234
Timeout (actionconfig) 234
Display (display) 235
Trace (trace) 236
Status criteria (status_criteria) 236
Context criteria (context_criteria) 237
User (userconfig) 238
User profile (userprofileconfig) 239
Notification list (nl) 241
Console (console) 241
User (user) 242
Tool/Action (action) 242
Console operation (consoleoperation) 243
Escalation policy (policyconfig) 244
Filter (filterconfig) 245
Escalation path (pathconfig) 246
Retire time (retireTime) 247
Enable time (enableTime) 247
Escalation level (escalationlevel) 248
Map icon configuration 249
mapgifconfig element 249
repos element 249
image element 250
Global Manager configuration DTDs 250
ics-config.dtd 250
consoleoper-config.dtd 252
map-config.dtd 252

Index

FIGURES

	Title	Page
1	Information flow between Global Manager, Managers, adapters, and Global Console	24
2	Information flow between Global Manager, Adapter Platform, and Business Dashboard	25
3	Global Manager Administration Console—Configure System Defaults	69
4	Global Manager Administration Console—Configure Business Templates	74
5	Global Administration Console Preferences dialog box	76
6	Global Manager Administration Console—Configure Domain interface	91
7	Global Manager Administration Console—Configure Domain Group interface	95
8	Global Manager Administration Console—Configure Domain Type interface	99
9	Global Manager Administration Console—Configure Domain Tag interface	104
10	Global Manager Administration Console - Disable autosync	121
11	Import XML Domain Configuration dialog box	157
12	Import XML Configuration dialog box	159
13	Escalation policy structure	163
14	Escalation policy view	164
15	Filter Value Helper dialog box	183
16	Filter notifications with severity of 1or 2	184
17	Filter notifications of element class router	184
18	Filter notifications with severity of 1-2 or an impact of 5 or more	185
19	Filter showing notifications owned by all users other than “SYSTEM” or “maint”	185
20	Filter showing routers with severity 1	186

TABLES

	Title	Page
1	Global Manager configuration files	32
2	Nonmodifiable Service Assurance configuration files	33
3	Notification list parameters	53
4	Console operations and groups	60
5	Console operations allowed for all users	63
6	Parameters defining the domainconfig section	66
7	Data exchange file for each product	67
8	System Defaults parameters	70
9	SAM performance monitoring threshold setting parameters	83
10	Available domains	88
11	Domain parameters	91
12	Domain Group parameters	95
13	Default domain types	97
14	Domain type settings	98
15	Domain type configuration parameters	100
16	Domain Tag parameters	105
17	Description of Event State notification attribute	110
18	States of NOTIFICATION-Router_R1_Down	112
19	Performance factors of SAM topology synchronization	119
20	Advantages and disadvantages of each mode	122
21	Attributes for matching criteria	127
22	Target classes for selective groups	129
23	Sample server tool scripts	143
24	Sample client tools scripts	145
25	Attributes passed to all tool scripts	146
26	Attributes for element targets	146
27	Attributes for notification targets	147
28	Attributes for escalation targets	149
29	Server tool configuration parameters	150
30	Arguments to sm_adapter for automated action	155
31	Available domain type descriptors	157
32	Escalation-related entries in the Audit Log	176
33	Expression filter fields	180
34	Arguments to sm_config for managing configurations	193
35	System Defaults parameters	202
36	ics-event-processing parameters	205
37	ics-nl-processing parameters	206
38	Console parameters	207
39	Environment variables	210
40	System classes and their attributes	212
41	Notification attributes	215
42	Basic wildcard patterns	220
43	Compound wildcard patterns	221
44	XML value indicators	226
45	Attribute-list declarations for the nlconfig element	227
46	Attribute-list declarations for the filterconfig element	228
47	Attribute-list declarations for the criterion element	230
48	Attribute-list declarations for the columnheading element	231
49	Column heading attributes and default display names	231
50	Attribute-list declarations for the actionconfig element	233

51 Attribute-list declarations for the display element 235
52 Attribute-list declarations for the userconfig element 238
53 Attribute-list declarations for the userprofileconfig element 240
54 Attribute-list declarations for the actionconfig element 243
55 Attribute-list declarations for the policyconfig element..... 244
56 Attribute-list declarations for the pathconfig element..... 246
57 Attribute-list declarations for the escalationlevel element 248
58 Attribute-list declarations for the map element..... 249

PREFACE

As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC technical support professional if a product does not function properly or does not function as described in this document.

NOTICE

This document was accurate at publication time. Go to EMC Online Support (https://support.emc.com) to ensure that you are using the latest version of this document.

Purpose

This document provides instructions for the configuration of EMC Smarts Service Assurance Manager, specifically the Global Manager. Topics include creating user profiles, specifying sources of topology and events, and managing topology and notifications.

Audience

This guide is intended for administrators who are responsible for deploying, installing, and configuring the Global Manager. IT managers who seek to understand the role of the Global Manager in the context of an EMC Smarts solution may also find this guide useful.

In addition to the configuration guides for specific components, administrators should also read the *EMC Smarts Service Assurance Manager Deployment Guide* and the *EMC Smarts System Administration Guide* Guide.

EMC Smarts Service Assurance Manager installation directory

In this document, the term BASEDIR represents the location where EMC Smarts software is installed:

- For Windows, this location is: C:\InCharge*‹product›*.

On Windows operating systems, this product is, by default, installed to: C:\InCharge\SAM\smarts. This location is referred to as *BASEDIR*/smarts.

Optionally, you can specify the root of BASEDIR to be something other than C:\InCharge (on Windows), but you cannot change the ‹*product*› location under the root directory.

The *EMC Smarts System Administration Guide* provides more information about the directory structure of EMC Smarts software.

EMC Smarts Service Assurance Manager

The EMC Smarts Service Assurance Manager includes the following products:

- Service Assurance Manager (Global Manager), includes Business Impact Manager (BIM) and Failover System
- Global Console
- Business Dashboard
- Companion UI
- SAM Native Adapters:
  - Service Assurance Manager Notification Adapters (E-Mail Notifier Adapter, Script Notifier Adapter, SNMP Trap Notifier Adapter, Log File Notifier Adapter)
  - Adapter Platform (Adapter Platform server, Syslog Adapter, SNMP Trap Adapter, sm_ems command-line interface)
  - EMC Smarts Adapter for UIM/O
  - XML Adapter

Related documentation

In addition to this document, EMC Corporation provides a help system for command line programs as well as product documentation.

Help for command line programs

Descriptions of command line programs are available as HTML pages. The index.html file, which provides an index to the various commands, is located in the BASEDIR/smarts/doc/html/usage directory.

EMC Smarts documentation

Readers of this guide may find the following related documentation helpful:

- *Smarts Foundation EMC Data Access API (EDAA) Programmer Guide*
- *EMC Smarts Common Information Model (ICIM) 1.11 Reference for Non-Service Assurance Manager Products*
- *EMC Smarts Common Information Model (ICIM) Reference for Service Assurance Manager*
- *EMC Smarts Foundation Documentation Portfolio*
- *EMC Smarts System Administration Guide*
- *EMC Smarts ASL Reference Guide*
- *EMC Smarts Perl Reference Guide*
- *EMC Smarts Dynamic Modeling Tutorial*
- *EMC Smarts MODEL Reference Guide*
- *FlexNet Publisher License Administration Guide*

EMC Smarts Service Assurance Manager documentation

The following documents are relevant to users of the EMC Smarts Service Assurance Manager:

- *EMC Smarts Service Assurance Manager Release Notes*
- *EMC Smarts Open Source License and Copyright Information*
- *EMC Smarts Service Assurance Manager Introduction*
- *EMC Smarts Installation Guide for SAM, IP, ESM, MPLS, VoIP, and NPM Managers*
- *EMC Smarts Service Assurance Manager Deployment Guide*
- *EMC Smarts Service Assurance Manager Configuration Guide*
- *EMC Smarts Service Assurance Manager Operator Guide*
- *EMC Smarts Service Assurance Manager Dashboard Configuration Guide*
- *EMC Smarts Business Impact Manager User Guide*
- *EMC Smarts Failover System User Guide*
- *EMC Smarts Service Assurance Manager Notification Adapters User Guide*
- *EMC Smarts Service Assurance Manager Adapter Platform User Guide*

- *EMC Smarts Adapter for Unified Infrastructure Manager/Operations Configuration Guide*
- *EMC Smarts XML Adapter User Guide*
- *EMC Smarts Notification Module User Guide*
- *EMC Smarts Service Assurance Manager Troubleshooting Guide*
- *EMC Smarts Companion UI Online Help*
- *EMC Smarts Companion UI Open Source License and Copyright Information for GPLv3 as Included with a Distribution of SLES 11*
- *EMC Smarts Service Assurance Manager Documentation Portfolio*

The *EMC Smarts Documentation Catalog* provides documentation resources for other EMC Smarts products.

Suggestions for searching PDF files

You can search across multiple PDF files by using the Adobe Acrobat Reader software:

1. If the documentation is not accessible to all users of the EMC Smarts product, copy the contents of the BASEDIR/smarts/doc/pdf directory to a central location, such as a shared drive on your LAN, so that operators and others can view the documentation.
2. To search throughout the documentation library, open the Acrobat Reader software:
   a. Select **Edit › Search**, and type a word or phrase.
   b. Select **All PDF Documents** in, in the **Where would you like to search** option, and type the pathname of the location where the PDF documents reside.

If you have more than one EMC Smarts product installed, you can set up cross-product document searches by copying files from the BASEDIR/smarts/doc/pdf directory for each product into this common documentation directory path.

Conventions used in this document

EMC uses the following convention for special notices:

NOTICE

NOTICE is used to address practices not related to personal injury.

Typographical conventions

EMC uses the following type style conventions in this document:

Bold	Use for names of interface elements, such as names of windows, dialog boxes, buttons, fields, tab names, key names, and menu paths (what the user specifically selects or clicks)
Italic	Use for full titles of publications referenced in text
`Monospace`	Use for: • System output, such as an error message or script • System code • Pathnames, filenames, prompts, and syntax • Commands and options
`Monospace italic`	Use for variables.
`Monospace bold`	Use for user input.
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections — the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Pathname conventions

Directory pathnames are shown with forward slashes (/). Users of the Windows operating systems should substitute back slashes (\) for forward slashes.

Graphical conventions

If there are figures illustrating consoles in this document, they represent the consoles as they appear in Windows. Under UNIX, the consoles appear with slight differences. For example, in views that display items in a tree hierarchy such as the Topology Browser, a plus sign appears for Windows and an open circle appears for UNIX.

Manager

Unless otherwise specified, the term Manager is used to refer to EMC Smarts programs such as Domain Managers, Global Managers, and adapters.

Where to get help

EMC support, product, and licensing information can be obtained as follows:

Product information — For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at:

https://support.emc.com

Technical support — Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to:

`techpubcomments@emc.com`

CHAPTER 1
Overview

This chapter provides a brief overview of EMC Smarts Service Assurance Manager and illustrates its architecture.

The successful deployment of the Service Assurance Manager requires knowledge of your operations environment and the management tools already in place. You can integrate Service Assurance with third-party tools and existing installations of EMC Smarts applications.

The *EMC Smarts Installation Guide for SAM, IP, ESM, MPLS, VoIP, and NPM Managers* provides the hardware and software requirements, as well as the procedures for installing the Service Assurance Manager.

This chapter contains the following information:

- Service Assurance Manager components 20
- Information flow between components 23

Service Assurance Manager components

EMC® Smarts® Service Assurance Manager includes the following components.

Global Manager

Global Manager (Service Assurance Manager) is the software component at the heart of Service Assurance. The Global Manager consolidates topology and event information from multiple underlying domains and provides a central point for monitoring and managing distributed analysis domains.

Global Console

The Global Console is the primary tool for operators and administrators. The console enables operators to monitor the state of the managed environment and quickly respond to notifications. The Global Console also allows users to launch map and topology views. As the primary tool for users, the Global Console is typically installed on many hosts. The Global Manager Administration Console enables administrators to create and manage:

- User profiles
- Notification lists
- Tools
- Escalation policies

Business Dashboard

Business Dashboard displays views normally available from the Global Console as viewlets within your organization's web page or a third-party web portal. Viewlets™ can display a variety of information including:

- Notification log
- Status table
- Maps
- Summary
- Notification properties
- Containment

Adapter Platform

Adapter Platform is a software component that:

- Imports SNMP traps, topology, and even information from third-party applications or products
- Normalizes the imported data to the ICIM Common Information Model (ICIM)
- Provides this data to the Global Manager

EMC Smarts Adapter for UIM/O

The EMC Smarts Adapter for Unified Infrastructure Manager/Operations (referred to as the EMC Smarts Adapter for UIM/O) provides integration between the EMC Smarts Adapter for Unified Infrastructure Manager/Operations (herein referred to as UIM/O) and the Service Assurance Manager. The adapter retrieves alerts from UIM/O, creates notifications from the alerts, and sends the notifications to the Global Manager.

The *EMC Smarts Adapter for Unified Infrastructure Manager/Operations Configuration Guide* provides deployment information, configuration instructions, and client tool examples.

Business Impact Manager

Business Impact Manager extends the capabilities of the Global Manager to:

- Calculate the business impact of events
- Propagate the impacts to affected business elements (ServiceOfferings, BusinessProcesses, and ServiceSubscribers)

The propagated impacts are discrete notifications that are connected through an event causality chain to the root-cause problems.

EMC Smarts Notification Module

The EMC Smarts Notification Module (NOTIF) augments EMC Smarts solutions with event management features that are configured through a graphical user interface (the NOTIF Editor). NOTIF enables the user to easily optimize the flow of events and notifications sent through any EMC Smarts system. NOTIF can be installed on a SAM server and/or an Adapter Platform server. The internal event and notification processing features of the standard SAM Adapter Platform are replaced by NOTIF.

The *EMC Smarts Notification Module User Guide* includes information on NOTIF functionality and architecture.

Companion UI

EMC Smarts Companion UI supplements existing functionality in SAM, and EMC Smarts Network Configuration Manager (NCM).

The Companion UI is a VMware®-based appliance, that is, a virtual machine. The *EMC Smarts SolutionPack Installation and Configuration Guide* provides detailed information about installing and configuring the Companion UI.

Watch4net

Watch4net provides cross-domain performance monitoring of infrastructure and data center components, in real-time, from a single customizable Watch4net dashboard. The EMC Smarts Console SolutionPack provides consolidated access to various console components of SAM and CUI from Watch4net.

If you are deploying SAM, Watch4net, and Companion UI and you want single sign-on capability, consult the *EMC Smarts SolutionPack Installation and Configuration Guide* for instructions.

EMC Smarts Notification Module Cisco Syslog Processing Adapter

The EMC Smarts Notification Module Cisco Syslog Processing Adapter (referred to as the NOTIF Cisco Syslog Adapter) replaces the log file processing features of the standard Syslog Adapter. The NOTIF Cisco Syslog Adapter processes log file information into useful notifications, with or without the use of ASL scripts.

The *EMC Smarts Notification Module Cisco Syslog Processing Adapter Installation and User Guide* provides additional information on this adapter.

Information flow between components

The following scenarios describe the architectural flow of topology and notifications between the different Service Assurance Manager components.

Scenario 1

The components described in Figure 1 on page 24 include:

- Service Assurance Manager:
  - Broker
  - Global Manager
  - Global Console
  - Report Manager
- Underlying Domains:
  - SAM Adapters
  - EMC Smarts IP Availability Manager
  - EMC Smarts IP Performance Manager

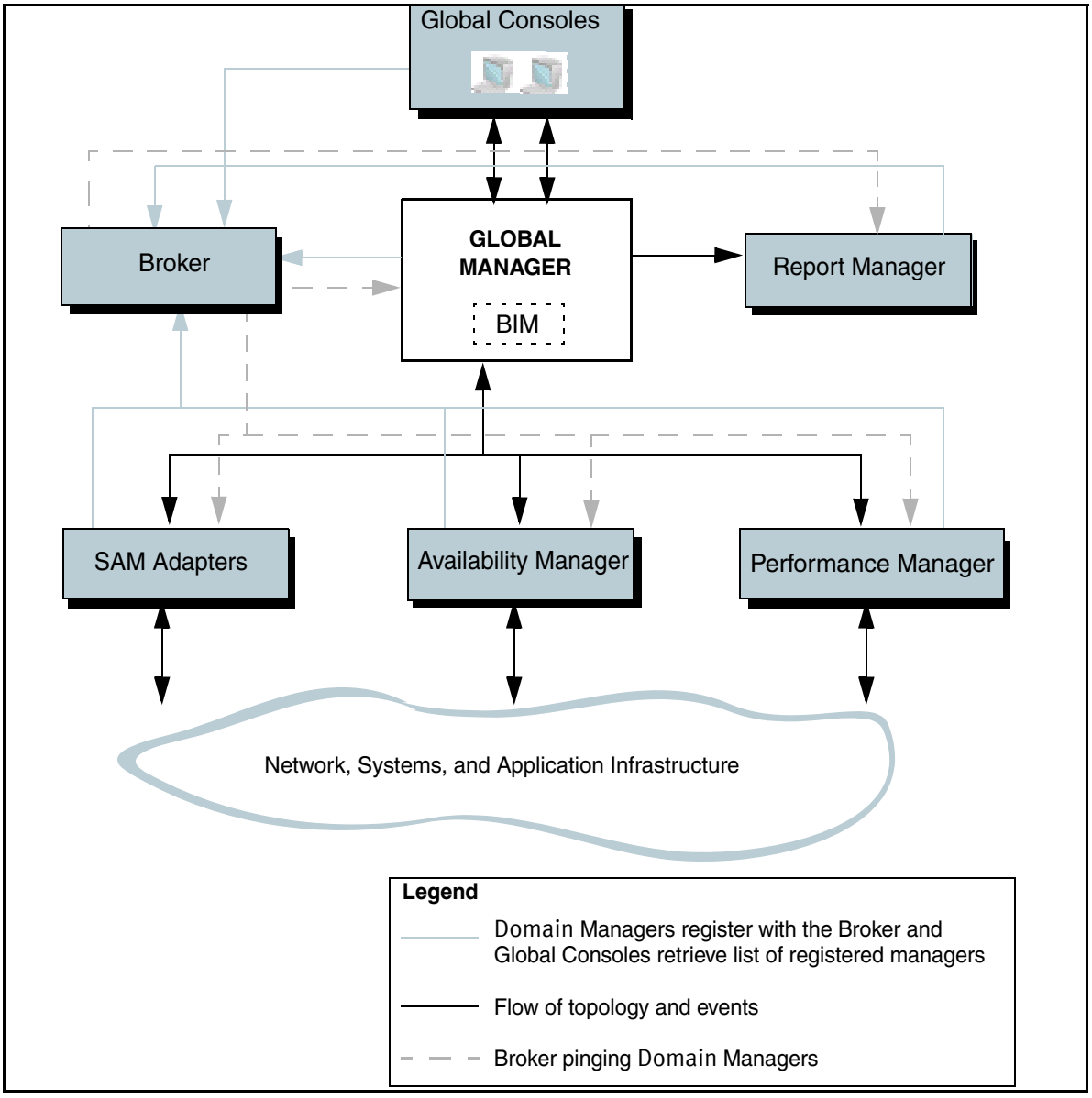

Figure 1 Information flow between Global Manager, Managers, adapters, and Global Console

Scenario 2

The second scenario, shown in Figure 2 on page 25, adds Adapter Platform and the Business Dashboard:

- Adapter Platform functions as an underlying domain by processing events and topology from disparate sources and sending this information to the Global Manager.
- Business Dashboard provides access to the Global Console by using standard web technologies such as HTML and Java.

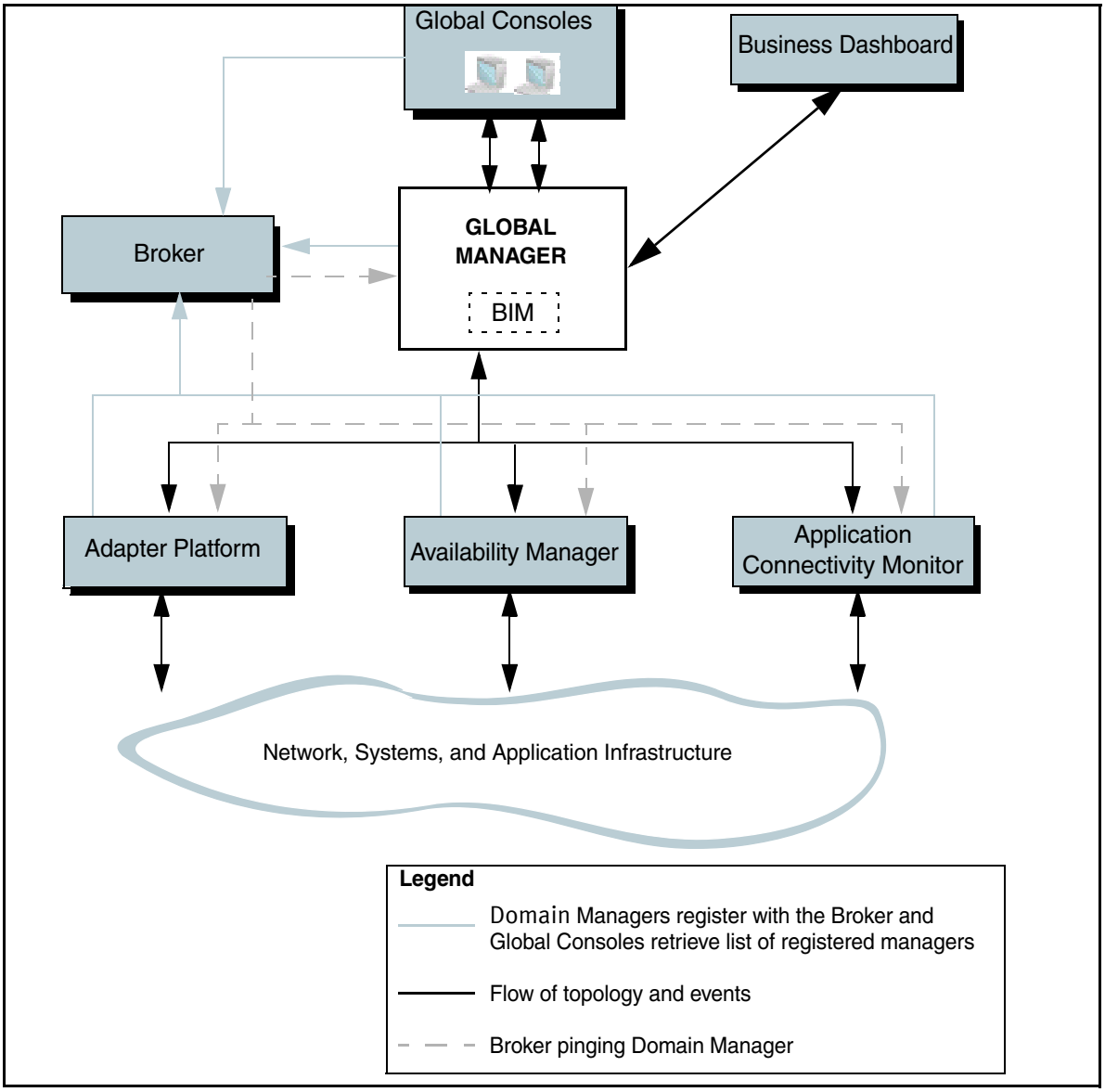

Figure 2 Information flow between Global Manager, Adapter Platform, and Business Dashboard

The Adapter Platform processes events and topology from third-party sources and provides this information to the Global Manager. Adapter Platform can receive events and topology from the following sources:

- SNMP traps — To process SNMP traps, you must configure the EMC Smarts SNMP Trap Adapter. The SNMP Trap Adapter can operate as a trap exploder, forwarding traps to multiple locations, or it can convert traps to notifications and forward the notifications to Adapter Platform. The NOTIF Editor may be used to graphically configure trap to notification management.
- System log files — To produce notifications from a system log file, you must configure the EMC Smarts Syslog Adapter. You must specify the appropriate system log file to process and make sure that the format of the log file matches the format described in the Syslog Adapter's ASL rule set. The Notification Module Cisco Syslog Processing Adapter may be used to manage Cisco type log messages into notifications.

- Third-party sources with sm_ems — To process events from a third-party source, you must configure the event source to invoke the **sm_ems** utility. You must invoke **sm_ems** so that it can notify or clear events, update the values of attributes, or create topology elements.

The *EMC Smarts Service Assurance Manager Adapter Platform User Guide* describes how to configure Adapter Platform, the SNMP Trap Adapter, the Syslog Adapter, and the **sm_ems** utility.

Business Dashboard

The Business Dashboard displays information about the managed environment through a web browser.

CHAPTER 2
Configuration Overview

After you install the Global Manager, one of the first configuration tasks is to specify the sources of topology and events and enable the Global Manager to begin working.

While there is much you can do to tailor Global Manager to your particular environment, the set of default configuration parameters provides a base from which to start. You can continue to configure the Global Manager while it is running.

This chapter contains the following information:

- Types of configuration tasks 28
- How to configure the Global Manager 30
- Starting and Configuring EMC Smarts NOTIF 34
- Starting services 36

Types of configuration tasks

This section organizes configuration options into groups of related tasks:

- Creating user profiles
- Specifying underlying domains and system-wide parameters
- Managing notifications and topologies
- Automating responses to notifications

Each of these groups includes several options:

- Creating user profiles
- Configuring system parameters
- Managing topology and notifications
- Automating responses to notifications

Creating user profiles

A user profile defines the environment for one or more users, that is, anyone who requires access to the Global Manager through the Global Console. You can create user profiles based on role (administrator/operator), job function (network management/helpdesk), or a combination of these. With a user profile, you specify the following:

- Notification lists determine which notifications the Global Manager shows to a particular user. You define one or more filters in a notification list to determine which notifications are available through that notification list. Chapter 6, "Managing Notifications with the Global Manager," provides additional information about notifications.
- Console operations determine a user's access privileges to various functions and commands provided by the Global Console. Such functions range from adding a view, to acknowledging a notification, and saving a console. You can select the console operations associated with each user profile. "Console operations" on page 59 lists the console operations and describes how to add or remove operations to a user profile.
- Tools provide a means by which users can respond to a notification. Such a response might include pinging a device to see if it is reachable or opening a trouble ticket. By defining the tools associated with a user profile, you determine what actions a user can invoke in response to a notification. Chapter 8, "Tool Configuration for the Global Manager," describes how to create and configure tools.
- Saved consoles provide users with a defined view of the network. This defined view can equal multiple displays of information. "Creating a saved console" on page 50 describes how to create a saved console and make it available to users over the network.

Configuring system parameters

System configuration tasks define the operation of the Global Manager:

- Specifying sources of topology and events and defining a smoothing interval and a minimum certainty for notifications received from each underlying domain.
- Setting parameters related to the archival of notifications.
- Defining the interaction between the Global Manager and Global Consoles.
- Defining the interaction between the Global Manager and underlying domains.

"Defining system defaults" on page 69 describes the system parameters.

Managing topology and notifications

Managing the topology and notifications maintained by the Global Manager requires an understanding of how the Global Manager synchronizes topology information and processes notifications.

You may need to perform one or more of the following tasks:

- Uniquely identify elements managed by two or more underlying domains (tagging). "How the Global Manager applies tags" on page 102 describes how to configure overlapping elements.
- Organize managed elements into groups. Chapter 7, "Managing Topology with the Global Manager," describes how to group topology elements.
- Customize the display of maps with new icons or map backgrounds. "Custom icons for map nodes" on page 136 describes how to add new icons or load map backgrounds.

The *EMC Smarts Notification Module User Guide* provides detailed information on how to manage notifications through the use of the NOTIF Editor.

Automating responses to notifications

Escalation policies provide a way to automatically respond to notifications. Chapter 9, "Escalation Configuration for the Global Manager," describes how to create escalation policies using the Global Manager Administration Console.

How to configure the Global Manager

You configure the Global Manager using the following methods:

- Global Manager Administration Console
- **sm_config** utility

Privilege requirements

To access the Global Manager Administration Console, you must have the following privileges:

- You must have All privileges, as determined by the serverConnect.conf file used by the Global Manager.
- You must have access to the Configure Global Manager Administration Console console operation.

To invoke the **sm_config** utility requires All privileges in the Global Manager's serverConnect.conf file as well as the necessary system privileges to edit installed files. Typically, administrator privileges are required to edit configuration files.

Global Manager Administration Console

The Global Manager Administration Console is a graphical interface for performing configuration tasks related to user profiles, escalation policies and custom map icons. The following wizards are provided to simplify configuration tasks:

- Notification List Wizard
- Automatic Tool Wizard
- Client Tool Wizard
- Server Tool Wizard
- User Profile Wizard
- User Wizard
- New Domain Wizard
- New Domain Group Wizard
- New Domain Type Wizard
- New Domain Tag Wizard

You can also create new users and notification lists through the User Profile Wizard.

Launching the Global Manager Administration Console

To open the Global Manager Administration Console:

From the **Global Console**, select **Configure** › **Global Manager Administration Console**.

The Global Administration Console is divided into two panes with a navigation tree in the left panel and a configuration panel in the right panel. Selecting an element in the tree displays content in the configuration panel. At the top level of the tree is the Global Manager you are configuring. Listed under the Global Manager are five elements configured through the console; Escalation Policies, Notification Lists, Tools, User profiles, and Users. Listed under each element are any instances of these elements. For example, the users default and maint are listed under the Users element.

To modify the icons displayed in the map consoles, from the **Global Manager Administration Console**, select **Configure** › **Edit Map Icons**.

The configuration of the Global Manager is stored in its repository file. Chapter 11, "Importing and Exporting Configurations," describes how to configure a Global Manager using the **sm_config** utility.

Modifying configuration files

As part of the EMC Smarts deployment and configuration process, you need to modify certain files. User-modifiable files include configuration files, rule set files, templates, and files (such as seed files and security configuration files) that contain encrypted passwords. Original versions of these files are installed into appropriate subdirectories under the *BASEDIR*/smarts/ directory.

Examples:

- The original versions of the Global Manager configuration files on UNIX are installed to /opt/InCharge/SAM/smarts/conf/ics
- The original versions of the Console configuration files on Windows are installed on C:\InCharge\SAM\smarts\conf\console

Original versions of files should not be altered. If a file requires modification, it must be stored as a local copy of the file in *BASEDIR*/smarts/local or one of its subdirectories.

Examples:

- A modified dxa-sam.conf file on UNIX should be saved to /opt/InCharge/SAM/smarts/local/conf/ics
- A modified properties.conf file on Windows should be saved to C:\InCharge\SAM\smarts\local\conf\console

EMC Smarts software is designed to first search for user-modifiable files in *BASEDIR*/smarts/local or one of its subdirectories. If a modified version of a file is not found in the local area, EMC Smarts software then searches appropriate nonlocal directories.

To facilitate proper file editing, EMC Corporation provides the **sm_edit** utility with every EMC Smarts product. When used to modify an original version of a file, this utility automatically creates a local copy of the file and places it in the appropriate location under *BASEDIR*/smarts/local. This ensures that the original version of the file remains

unchanged. You can invoke **sm_edit** from the command line. Optionally, you can configure Windows so that **sm_edit** is automatically invoked when user-modifiable files are double-clicked in Windows Explorer.

To invoke the **sm_edit** utility from the command line, specify the path and the name of the file to edit under *BASEDIR*/smarts. If multiple EMC Smarts products are running on the same host, invoke **sm_edit** from the bin directory of the product with the files to edit.

Example:

To edit the configuration file for the Global Manager, invoke the **sm_edit** utility as follows:

```
# /opt/InCharge/SAM/smarts/bin>sm_edit conf/ics/dxa-sam.conf
```

In this example, the **sm_edit** utility automatically creates a local copy of the dxa-sam.conf file in the /opt/InCharge/SAM/smarts/local/conf/ics directory and opens the file in a text editor. If a local version of the file already exists, the **sm_edit** utility opens the local version in a text editor. In addition, **sm_edit** creates any necessary subdirectories.

The *EMC Smarts System Administration Guide* provides additional information about the **sm_edit** utility.

A number of configuration files exist for the Global Manager. The *BASEDIR*/smarts/local/conf/ics directory contains configuration files related to the Global Manager.

Table 1 on page 32 lists the configuration files for the Global Manager.

Table 1 Global Manager configuration files

Directory	Files	Description
conf	serverConnect.conf clientConnect.conf brokerConnect.conf sm_cloneDir.conf	Security files for configuring access to Service Assurance Manager. The *EMC Smarts System Administration Guide* describes the syntax and configuration of these files.
conf/ics	service.data.template topology-group.data.template	Sample datafiles for importing services and groups into the Global Manager.
	actionconfig-sample.xml ics-config-sample.xml nlconfig-sample.xml policyconfig-sample.xml profileconfig-sample.xml userconfig-sample.xml mapgif-sample.xml	The XML files that define sample configurations for notification lists, user profiles, users, and custom map icons for the Global Manager.
	weights.conf	The configuration file for EMC Smarts Business Impact Manager. The *EMC Smarts Business Impact Manager User Guide* describes how to configure this file.
local/conf	runcmd_env.sh	Used to set environment variables for EMC Smarts software.

Configure Secure Broker

To configure Secure Broker, edit serverConnect.conf and clientConnect.conf files in the *BASEDIR*/smarts/local/conf/ics directory.

- Edit the serverConnect.conf file as follows:
  1. In the serverConnect.conf file, comment out the following line:

     ```
     <BROKER>:<DEFAULT>:<DEFAULT>:All
     ```

  2. Modify the BrokerNonsecure setting as follows:

     ```
     <BROKER>:BrokerNonsecure:Nonsecure:Ping
     ```

  3. Add the following line after the BrokerNonsecure line

     ```
     <BROKER>:SecureBroker:Secure:All
     ```

- Edit the clientConnect.conf file as follows:
  1. In the clientConnect.conf file, comment out the following line:

     ```
     *:<BROKER>:BrokerNonsecure:NonSecure
     ```

  2. Add a secure value as follows:

     ```
     *:<BROKER>:<PROMPT>:<PROMPT>
     ```

     where Secure value can be encrypted.

Table 2 on page 33 lists additional files used by the Global Manager. The files listed in Table 2 on page 33 should not be modified by the user.

Table 2 **Nonmodifiable Service Assurance configuration files**

Directory	Files	Description
conf/ics	ics-config.dtd ics-default.xml console_operations.xml consoleoper-ext.xml consoleoper_config.dtd	The XML and data type definition (DTD) files that define default notification lists, user profiles, users, and console operations for the Global Manager. These XML files provide the default configuration for the Global Manager.
conf/maps	instance_maptype-sample .xml map-config.dtd map_gif.xml	The XML and DTD files that define custom map icons.
conf/xml-if	icim_xml.dtd sample.xml	A sample XML file and the DTD file to define the XML standard used to import ICIM objects into EMC Smarts software. The *EMC Smarts XML Adapter User Guide* describes how to import topology using XML.

If your deployment involves adapters that integrate with any third-party products, there might be additional configuration and XML definition files. For more information regarding adapter configurations, refer to the guide for that adapter.

Starting and Configuring EMC Smarts NOTIF

After you install and configure the Global Manager, you must also configure NOTIF if you want to optimize the flow of events and notifications sent through your EMC Smarts system. To start and configure NOTIF:

1. Set an environment variable in the *BASEDIR* directory path to ensure that Java can be successfully started for NOTIF. In *BASEDIR*/smarts/local/conf/**runenv_cmd.sh**, add the following line:

   ```
   export SM_JAVA_ENABLED=YES
   ```

2. Configure the SNMP Trap Adapter to use the Notif-trap_mgr_parse.asl script instead of the default trap_mgr_parse.asl script in order for NOTIF to process SNMP traps.

For example:

```
./sm_trapd --name=TRAP-OI --server=INCHARGE-OI --config=icoi
--port=9013 --model=sm_actions --output
--rules=icoi-trapd/Notif-trap_mgr_parse.asl --daemon
```

NOTICE

The *--daemon* option only works on UNIX systems, not Microsoft Windows systems. Any customizations made to the original trap_mgr_parse.asl file in the BASEDIR/smarts/rules/icoi-trapd directory must be incorporated into the Notif-trap_mgr_parse.asl file in the BASEDIR/smarts/rules/icoi-trapd directory.

3. Configure the Syslog Adapter to use the Notif-SysLog_mgr.asl script instead of the default SysLog_mgr.asl script in order for NOTIF to process Cisco system log files.

   For example:

   ```
   sm_adapter.exe --name=SYSLOG-OI --server=INCHARGE-OI
   --tail=/var/log/syslog --model=sm_system --model=sm_actions
   --output icoi-syslog/Notif-SysLog_mgr.asl
   ```

   NOTICE

   One or more adapters can feed a single NOTIF server.

4. Start the NOTIF Editor by selecting **Start** › **Programs** › **InCharge** › **Smarts NOTIF Editor**.

   NOTICE

   You can also start the editor by double-clicking the **NotifGui.bat** file (for Windows systems) or the **NotifGui.sh** file (for UNIX systems) in the BASEDIR/smarts/notif/editor directory.

5. Use the NOTIF Editor to connect to the running SAM or Adapter Platform server. Select **Remote** › **Edit a server's settings** in the NOTIF Editor.

6. The **Connect to a Server** dialog box displays the list of available Adapter Platform and SAM server connections. Choose a server from the list of available server connections in the **Connect to a Server** dialog box and click **OK**.

   If no available server connections are displayed, do the following to populate the list of server connections:

   a. Click the **More** button in the **Connect to a Server** dialog box. This launches the **Manage Connections** dialog box where you can add server connections.

   b. Click the **Add** button in the **Manage Connections** dialog box.

      The **Input** dialog box displays.

   c. Enter a connection reference name (for example, "Remote Smarts NOTIF OI server") in the **Input** dialog box and click **OK**.

   d. Fill in the new connection record in the right pane of the **Manage Connections** dialog box.

   e. Click **OK** to save the connection setup.

   The **Remote Server Settings** dialog box displays, showing the server's current settings.

7. Select the **Activate Smarts NOTIF** checkbox, and then click **OK**.

   NOTICE

   You can also change other server settings in the **Remote Server Settings** dialog box if necessary.

8. Restart your SAM/Adapter Platform server.

   NOTICE

   Server setting changes made in the Smarts NOTIF Editor **Remote Server Settings** dialog box are persistent. When you change server settings in the **Remote Server Settings** dialog box and click **OK**, the changes are saved to the **Notif_Settings.import** file that is generated and saved to the **BASEDIR**/smarts/local/conf/‹*icoi* or *ics*› directory in the Adapter Platform or SAM server where Smarts NOTIF is running.

Server setting changes made in the Smarts NOTIF Editor are preserved even if the repository is deleted. For example, if the repository is erased because you used the --norestore option for server startup, the server uses the last saved settings from the Smarts NOTIF Editor saved in **BASEDIR**/smarts/local/conf/‹*icoi* or *ics*›/**Notif_Settings.import**. The *EMC Smarts Notification Module User Guide* includes information on how to use the Smarts NOTIF Editor.

Starting services

During installation, Smarts products may be installed as services. When installed as services, default values are specified for the parameters that are associated with the services.

The parameters for the Global Manager service:

```
sm_service install --force  --unmanaged  --startmode=runonce \
   '--name=ic-sam-server' \
   '--description=EMC Smarts Service Assurance Manager Server' \
   '/data/username/SAM_811_clean/SAM/smarts/bin/sm_server' \
   '-n' \
   'INCHARGE-SA' \
   '--config=ics' \
   '--port=0' \
   '--ignore-restore-errors' \
   '--output'
```

The parameters for the Adapter Platform service:

```
sm_service install --force
   --name=ic-sam-server
   "--description=EMC Smarts Service Assurance Manager Server"
   --startmode=runonce
   E:\data\SAM\9.2\SAM\smarts\bin\sm_server.exe  "-n"
      "INCHARGE-SA"
      "--config=ics"
      "--port=0"
      "--ignore-restore-errors"
      "--output"
```

The parameters for the SNMP Trap adapter service:

```
/opt/InCharge/SAM/smarts/bin/sm_service
   install --force --unmanaged --startmode=runonce
   --name=ic-trapd-receiver
   --description="EMC Smarts SNMP Trap Adapter"
   /opt/InCharge/SAM/smarts/bin/sm_trapd
      --name=TRAP-INCHARGE-OI
      --server=INCHARGE-OI
      --config=icoi
      --port=162
      --model=sm_actions
      --rules=icoi-trapd/Notif-trap_mgr_parse.asl
      --seed=seedfile
      --output

C:\InCharge\SAM\smarts\bin>sm_service
   install --force --name=ic-trapd-receiver
   --description="EMC Smarts SNMP Trap Adapter"
   C:\InCharge\SAM\smarts\bin\sm_trapd.exe
      --name=TRAP-INCHARGE-OI
      --server=INCHARGE-OI
      --config=icoi
      --port=162
      --model=sm_actions
      --rules=icoi-trapd/Notif-trap_mgr_parse.asl
      --seed=seedfile
      --output
```

The parameter for the Syslog adapter service:

```
/opt/InCharge/SAM/smarts/bin/sm_service
   install --force  --unmanaged  --startmode=runonce
```

```
   --name=ic-syslog-adapter
   --description="EMC Smarts Syslog Adapter"
   /opt/InCharge/SAM/smarts/bin/sm_adapter
      --name=SYSLOG-INCHARGE-OI
      --rserver=INCHARGE-OI
      --tail=/var/log/syslog
      --model=sm_system
      --model=sm_actions
      --output
      icoi-syslog/Notif-SysLog_mgr.asl

C:\InCharge\SAM\smarts\bin>sm_service
   install --force --name=ic-syslog-adapter
   --description="EMC Smarts Syslog Adapter"
   C:\InCharge\SAM\smarts\bin\sm_adapter.exe
      --name=SYSLOG-INCHARGE-OI
      --rserver=INCHARGE-OI
      --tail=/var/log/syslog
      --model=sm_system
      --model=sm_actions
      --output
      icoi-syslog/Notif-SysLog_mgr.asl
```

When Smarts products are installed as services, they start automatically upon system restart. The following sections describe additional methods to start Smarts services.

Starting services on Windows

To start an EMC Smarts service from the Windows desktop:

1. Select **Settings** › **Control Panel** › **Administrative Tools**.
2. Select **Services**.
3. Right-click the EMC Smarts service to select.
4. Select **Start** from the menu.

CHAPTER 3
User Configuration for the Global Manager

This chapter contains the following information:

- Configuring a security message 40
- User profiles 41
- User accounts 48
- Creating a saved console 50
- Notification lists 52
- Console operations 59

Configuring a security message

A security disclaimer message can be configured to display when users attempt to start the Global Console, Business Dashboard console, or the Web Console.

NOTICE

By default, the disclaimer dialog box is not enabled.

Configuring the security message for the Global Console

The file, security.warning, used to configure the disclaimer message is located in the *‹BASEDIR›*\smarts\local\conf directory. The file, security.warning, is a simple ascii text file containing the security warning message text. This text appears in the security warning dialog box.

NOTICE

The file used to configure the disclaimer message *must* be labeled security.warning. If the file does not exist or is labeled a different name, the security warning message will not appear when the console starts.

To configure the disclaimer message for the Global Console:

1. Verify that the security.warning file is located in the ‹*BASEDIR*›\smarts\local\conf directory.

   NOTICE

   The security.warning file must be manually created if it already does not exist in the ‹BASEDIR›\smarts\local\conf directory.

2. Modify the text in the security.warning file, if necessary.

Configuring security message for Business Dashboard viewlet and Web Console

For the security message to appear on either the Business Dashboard viewlet or the Web Console, the com.smarts.securityMessage parameter must be configured.

The value of the com.smarts.securityMessage parameter is the filename of the ascii file containing the security message. For example, if the security message file was created and saved as trespass.notice, the property would be specified as:

```
com.smarts.securityMessage=trespass.notice
```

NOTICE

If the parameter is not specified in the properties file, or if the files exists but has zero length, the warning message will not appear when the web console or dashboard starts. If the parameter is specified but the specified file cannot be found, a default security message appears.

The Adjusting Java Properties section of the *EMC Smarts Service Assurance Manager Operator Guide* provides additional information.

User profiles

User profiles provide a means by which an administrator can create and control the access and display of users who attach to the Global Manager through the Global Console, including the EMC Smarts web console and the Business Dashboard. For example, an administrator can create one user profile for operators who monitor the network infrastructure and a second user profile for operators who monitor application performance. All user accounts for a Global Manager are associated with a user profile.

A user profile includes the following properties:

- A notification list determines what notifications the Global Manager sends to the users.
- One or more users defined for the Global Manager. You must also ensure that these users have access to the Global Manager through the serverConnect.conf security file. The *EMC Smarts System Administration Guide* describes the configuration of security files.
- Server or client tools that enable users to invoke programs in response to notifications. Server and client tools, which are optional, are separately assigned to user profiles.
- One or more saved consoles that open when the user attaches to the Global Manager.
- Console operations, which are user actions on the console and determine a user's access privileges to various functions and commands provided by the Global Console.

You can create user profiles with the Global Manager Administration Console or by using the **sm_config** utility. Chapter 11, "Importing and Exporting Configurations," describes the **sm_config** utility.

Creating a user profile

When using the User Profile Wizard, you are prompted to specify the users, notification list, tools, saved consoles, and console operations associated with the user profile.

NOTICE

You can create users and notification lists from within the User Profile Wizard.

As you proceed through the wizard, you may notice that certain elements, such as users or tools appear grayed out. This indicates that these elements are not enabled. You can associate disabled elements with a user profile, however such elements are not available to the user profile. For example, if a tool is not enabled, it will not be available to a user until it is enabled. You can enable or disable elements by selecting them in the tree and selecting or clearing the Enable box in the Configure User Profile panel.

In addition, you can click Finish at any point and the wizard will create the user profile. If you are creating a new user profile, any options set after the point where you selected Finish will be blank. If you copied an existing user profile, the new profile will include the options of the copied profile.

NOTICE

Users may have to detach and reattach for changes to their user profile to take effect.

To create a user profile using the User Profile Wizard:

1. From the **Global Manager Administration Console**, click the **Launch User Profile Wizard** toolbar button.

   The **User Profile Creation Wizard** appears.

2. Type the name of the user profile and choose whether to create a new profile or to copy an existing profile, and click **Next**. If you choose to copy an existing profile, the remaining configuration options will be identical to the user profile that was copied.

   The name of a user profile cannot be the same as the name of a user. For example, you cannot create a user profile Operator and a user Operator. A suggested naming convention is to name the user profile with a –profile extension. For example, you could have a user profile called Operator–profile.

3. Select a notification list from the list box and click **Next**. Optionally, you could choose **New Notification List** from the menu to create a new notification list. "Notification list parameters" on page 53 describes how to create notification lists.

4. Assign users to this user profile. You can select multiple users by holding the **Ctrl** key when selecting users. Click the **Add** button to associate the user or users with the user profile.

   If the user does not exist, you can create a user by doing the following:

   a. Type the username in the **Create and add new user** field.

   b. Click **Add**.

   c. Click **Next**.

   NOTICE

   A user can belong to only one user profile. If the user you select is already a member of another user profile, the user is associated with the new user profile.

5. Select the appropriate server tools from the list of Available server tools and click the **Add** button to associate them with the user profile. You can select multiple tools by holding the **Ctrl** key while selecting. When finished, click **Next**.

   NOTICE

   Click the **Allow user to launch client or server tools only for a single selected object** option to allow users to only launch client or server tools for a single selected object.

6. Select the appropriate client tools from the list of Available client tools and click the **Add** button to associate them with the user profile. You can select multiple tools by holding the **Ctrl** key while selecting. When finished, click **Next**.

   NOTICE

   Click the **Allow user to launch client or server tools only for a single selected object** option to allow users to only launch client or server tools for a single selected object.

7. Select the saved consoles that are opened when users attach to the Global Manager, and click **Next**. If a saved console is not selected, the default NotificationLog console will appear when the user attaches to the Global Manager.

   If the appropriate console is not listed, type the name of the console in the **Add new console** field and click the **Add** button. You can specify the name of a saved console whether or not the saved console exists.

8. Select the console operations for users associated with this user profile. You can select **Other** to choose specific console operations or choose one of the default sets of console operations:

   - Read Only
   - Operator
   - Administrator

   The default sets of console operations cannot be modified.

   You can assign a modified version of one of the default sets of console operations by first selecting the set that provides most of the console operations to assign. Then, select **Other**. When you click **Next**, the console operations that correspond to the set you chose are selected but you can enable or disable individual console operations.

9. Click **Next**.

   The final screen of the User Profile Wizard shows all newly created elements, including the new user profile. By default, the user profile is enabled. You can disable it by clearing the box next to the profile name.

   NOTICE

   If you chose to copy an existing profile in step 2, and if the existing profile is disabled, then the user profile is disabled by default.

10. Click **Finish** to display a dialog box confirming that the user profile elements were created.

Deleting a user profile

To delete a user profile:

1. Select the user profile.
2. Select **Edit** › **Delete**.

You can disable a user profile without removing it. "Disabling a user profile" on page 44 describes how to disable a user profile.

NOTICE

You must not delete the default-profile user profile. When users attach to a Global Manager and their username is not assigned to a user profile, the default-profile user profile is applied. If the default-profile does not exist, an error message appears and the Global Console will not open.

Modifying a user profile

To modify a user profile:

1. Select the user profile.
2. Modify the properties of the user profile in the right panel:
   a. Change the notification list.
   b. Add or remove users.
   c. Add or remove server tools.
   d. Add or remove client tools.
   e. Add or remove saved consoles.
   f. Add or remove console operations.
3. Click **Apply**.

Changes to a user profile may not be available to console users until they restart the Global Console.

Disabling a user profile

You can disable a user profile without having to delete it. When users of a disabled profile attach to the Global Manager, they are assigned the default-profile. If the default-profile is deleted, users associated with a disabled user profile cannot attach to the Global Manager.

To disable a user profile:

1. Select the user profile.
2. Clear the **Enabled** checkbox.
3. Click **Apply**.

Changing the notification list for a user profile

A notification list determines what notifications a user sees in the Global Console. Changing the notification list associated with a user profile affects all the users of this profile.

The Default notification list filters out notifications where the value of the Owner attribute is maint. When the user maint takes ownership of a notification, that notification is removed from the display of users assigned the Default notification list.

There is a difference between taking ownership of a notification and acknowledging a notification. If a notification is:

- Owned and acknowledged and then recurs, the ownership and acknowledgement values are cleared in the new notification.
- Owned but not acknowledged and the notification recurs, the ownership is maintained and the recurring notification is also filtered from the operator's display.

"Notification lists" on page 52 describes the properties of the default notification lists.

To change the notification list for a user profile, select a new notification list from the Notification List menu. "Notification list parameters" on page 53 describes how to use the Notification List Wizard to create a notification list.

Adding or removing a user

A user can be associated with only one user profile. If you add a user to a user profile, the user is automatically removed from its previous user profile.

To add a user to a profile:

1. Select the user profile from the tree.
2. From the **Configure User Profile** panel, click **Modify List** in the **Users** section.

   The **Modify Users** dialog box appears.
3. Add or remove users from the user profile.
4. Click **OK**.
5. Click **Apply** at the bottom of the **Configure User Profile** panel. "User accounts" on page 48 describes how to create a user account.

To remove a user, select the user profile in the tree. From the **Configure User Profile** panel, select the user from the Users list and click **Remove Selected**. Click **Apply** at the bottom of the **Configure User Profile** panel.

When you remove a user from a user profile, the user is automatically associated with the default-profile until you assign the user to a different user profile. "Creating a saved console" on page 50 describes the default user profiles.

Adding or removing a tool

Client and server tools are configured separately, however the process for adding or removing a client or server tool from a user profile is the same.

To add a tool to a user profile:

1. Select the user profile in the tree.
2. From the **Configure User Profile** panel, click **Modify List** in the **Server Tools** or **Client Tools** section.

   The **Modify Server Tools** or **Modify Client Tools** dialog box appears.
3. Add or remove tools from the user profile.
4. Click **OK**.
5. Click **Apply** at the bottom of the **Configure User Profile** panel.

To remove a tool:

1. Select the user profile in the tree.
2. From the **Configure User Profile** panel, select a tool and click **Remove Selected**.
3. Click **Apply** at the bottom of the **Configure User Profile** panel.

Chapter 8, "Tool Configuration for the Global Manager," describes how to configure tools.

Adding or removing a saved console

A saved console provides users with a preconfigured view of the network, helping users focus on aspects of the managed system they are to monitor.

To add a saved console to a user profile:

1. Select the user profile in the tree.
2. From the **Configure User Profile** panel, click **Modify List** in the **Saved Consoles** section.

   The **Modify Saved Consoles** dialog box appears.
3. Add or remove saved consoles from the user profile.
4. Click **OK**.
5. Click **Apply**.

To remove a saved console:

1. Select the user profile in the tree.
2. From the **Configure User Profile** panel, select a saved console and click **Remove Selected**.
3. Click **Apply** at the bottom of the **Configure User Profile** panel.

"Creating a saved console" on page 50 describes how to create a saved console.

Adding or removing console operations

Console operations are the functions and commands a user can invoke through the Global Console. By adding or removing console operations, an administrator can determine the level of access a user has to console operations. "Console operations" on page 59 lists the various console operations.

To add or remove console operations for a user profile:

1. Select the user profile in the tree.
2. From the **Configure User Profile** panel, click **Modify List** in the **Console Operations** section.

   The **Console Operations** dialog box appears.
3. Select or clear console operations for the user profile.

   NOTICE

   In addition, you can select one of the three default groups of console operations and apply it to the user profile.
4. Click **OK** to close the dialog box.
5. Click **Apply** at the bottom of the **Configure User Profile** panel.

About the predefined user profiles

Service Assurance includes four editable user profiles: admin-profile, oper-profile, default-profile, and maint-profile. The default-profile, admin-profile, and oper-profile user profiles include the default notification list and the Notification Log console.

default-profile

The default-profile user profile provides read-only operations and no tools. This profile ensures that any user that is not associated with a user profile or whose user profile is disabled, can attach to the Global Manager. The user must still authenticate with the Global Manager.

maint-profile

The maint-profile also serves a special purpose, which is to take ownership of notifications that are not relevant to operators. For example, if notifications are generated for interfaces that should not be managed, or for components that are known to be faulty but are not scheduled to be fixed, an administrator can log in as the user maint and take ownership of these notifications.

admin-profile

The admin-profile includes all console operations access groups, server tools, and client tools, assuming the user has been given "All" privileges on the server side using the serverConnect.conf configuration file.

oper-profile

The oper-profile includes:

- All client and server tools
- Details about notifications
- Maps
- Notification management, such as acknowledge
- Taking ownership of a notification
- Groups configurations
- Restricted access to configuration and management of user console and views

User accounts

A user account provides a unique identifier for each person who attaches to the Global Manager with the Global Console. The username provides a method for tracking who performs the following operations on notifications:

- When users acknowledge or unacknowledge a notification, their username is added to the notification's Owner attribute and recorded in the notification's audit trail.
- When users take ownership, their username is added to the notification's Owner attribute and this information is recorded in the notification's audit trail.
- When users invoke a tool on a notification, the user's name is recorded to the notification's audit trail.

When a user attaches to a Global Manager, the console automatically tries to associate the user with a user profile. If the user is not assigned to a profile, the default-profile is used.

NOTICE

Users must authenticate with a matching username and password in the serverConnect.conf file used by the Global Manager. The *EMC Smarts System Administration Guide* describes the syntax and configuration of the security files.

Creating a user account

You can create new users in one of the following ways:

- Through the User Profile Wizard when you create a user profile as described in "User profiles" on page 41.
- Through the User Wizard as described below.
- Through the **sm_config** utility as described in Chapter 11, "Importing and Exporting Configurations".

To create a new user with the User Wizard:

1. Select the **Launch User Wizard** toolbar button.

   The **User Creation Wizard** appears.
2. Type the name of the new user.

   The username must be unique.
3. Click **Next**.
4. Select a user profile for this user from the list box.
5. Do one of the following:
   - Click **Next** to display a confirmation screen.
   - Click **Finish** to create the user.
6. Add the username and password in the serverConnect.conf file used by the Global Manager.

Deleting a user account

To delete a user:

1. Select the user in the tree.
2. Select **Edit** › **Delete**.

NOTICE

Do not disable or remove the Admin user account. If the Admin user account is removed or disabled, but other users still have the same privileges, none of the applications can be shut down.

Creating a saved console

You can create a console layout for the Global Console, save it to the Global Manager, and associate it with a user profile so that it automatically opens for users associated with that profile. In addition, you can save the console to the Global Manager so that users can open it from the Global Console by selecting File › Open Remote.

There are two methods to set up access to a saved console:

- Specify the console name when you set up a new user profile, the saved console opens automatically when the user attaches to the Global Console.
- Save a console to a directory that users can access. Console users can use the Open Remote As command to open consoles in this directory.

The *EMC Smarts Service Assurance Manager Operator Guide* describes how to set up and save a console.

Saving a remote console for user access

You can save a console to one of the following locations on the host where the Global Manager is running:

- Use *BASEDIR*/smarts/local/consoles for all users.
- Use *BASEDIR*/smarts/local/consoles/*‹user›* for a specific user.
- Use *BASEDIR*/smarts/local/consoles/*‹user_profile›* for all users assigned a specific user profile.

NOTICE

The *BASEDIR*/smarts/local/consoles/*‹user_profile›* directory must be manually created by the administrator.

When a console is saved by selecting File › Save As › Save Remote As, the console is saved to the *BASEDIR*/smarts/local/consoles/*‹USER›* directory, where *‹USER›* is the EMC Smarts username. Users cannot see or open the consoles saved in another user's directory unless it is manually copied into:

- Their own console directory.
- The *BASEDIR*/smarts/local/consoles directory.
- The *BASEDIR*/smarts/local/consoles/*‹user_profile›* directory that corresponds to the user profile associated with their username.

Providing a saved console in a user profile

To provide a saved console in a user profile:

1. Configure the console layout and select **File** › **Save As** › **Save Remote As**.

   The console is saved to a directory that corresponds to your username under *BASEDIR*/smarts/local/consoles. For example, if you are logged in as the user "admin", the console is saved to the *BASEDIR*/smarts/local/consoles/admin directory.

   When you select **Save Remote As**, you are prompted to select the Domain Managers to which the saved console should automatically reattach in the event of a disconnect. The list includes those Domain Managers to which you are currently attached. By choosing to automatically reattach to a Domain Manager, you can omit the attach and detach console operations from the user profile.

2. To make the console available to all Global Console users, move the saved console from its current directory to the *BASEDIR*/smarts/local/consoles directory.

   When a saved console is located in this directory, a user can also open the console using **File** › **Open Remote**.

3. Add the console to the user profiles as described in "Adding or removing a saved console" on page 46.

4. To make the console available to users of a particular user profile, move the saved console from its current directory to the *BASEDIR*/smarts/local/consoles/‹*user_profile*› directory.

   NOTICE

   The *BASEDIR*/smarts/local/consoles/‹*user_profile*› directory must be manually created by the administrator.

Notification lists

A notification list determines what notifications a client of the Global Manager receives. A notification list includes a filter that the Global Manager uses to process notifications before sending the notifications to the client. In addition to a filter, you can also use a notification list to change the name of column headings in the Notification Log of the Global Console.

NOTICE

Changes to a notification list are not available to console users until they reattach to the Global Manager.

Clients that use notification lists include:

- Global Consoles — The notification list is a property of the user profile. "Creating a user profile" on page 41 describes how to create user profiles.
- Notification adapters — The notification list used by a notification adapter is specified in the NLSubscription section of the adapter's configuration file. The *EMC Smarts Service Assurance Manager Notification Adapters User Guide* provides additional details on notification adapters.
- Adapter Platform — A Global Manager uses a notification list to receive notifications from the Adapter Platform. The notification list used by Global Manager is specified in the dxa-oi.conf file.

NOTICE

Changes to the notification list are made using the Global Console and not by directly editing the dxa-oi.conf file.

Because a notification list controls what notifications a client receives, it affects more than the display of notifications in a Notification Log. A notification list also affects the display of maps in the Global Console. For example, if a notification list filters out all notifications for a router, the status of that router will appear normal, regardless of its actual condition.

The filters specified as part of a notification list should not be confused with the filtering capabilities of the Global Console. The Global Manager performs the filtering defined in a notification list. The notifications that are filtered out by a notification list are not sent to a client. Filters in the Global Console can be used to further refine the filter of a notification list. The *EMC Smarts Service Assurance Manager Operator Guide* describes filtering performed by the Global Console.

You can create or modify a notification list using the Global Manager Administration Console or the **sm_config** utility. This section describes using the Global Manager Administration Console. Chapter 11, "Importing and Exporting Configurations," describes how to use the **sm_config** utility.

Notification list parameters

A notification list includes the properties described in Table 3 on page 53.

Table 3 Notification list parameters

Field	Description
Name	Name of the notification list, must be unique among notification lists.
Filter and ASL Filter	Type of filter used to filter notifications. You can specify two types of filters: an expression filter or an ASL filter: • An expression filter matches a wildcard expression against the value of a notification attribute. You create an expression filter using the filter builder. "Building expression filters" on page 180 describes expression filters. • An ASL filter is specified by checking the ASL Filter checkbox and typing the name of the ASL file. The ASL filter file must be located in the BASEDIR/smarts/local/rules/ics directory. You must also specify the ics directory with the name of the filter, for example, ics/myASLfilter.asl. If no filter is specified, the notification list matches all notifications.
Column Heading	Used to modify the column headings that are displayed in the Notification Log. "Attributes for matching notification properties" on page 215 lists notification attributes.

You can use expression filters and ASL filters in combination, as well as specify multiple expressions in an expression filter. However, each filter definition sheets can contain only expression filters or an ASL filter.

You cannot have expression filters and ASL filters on the same sheet:

- When an expression filter and an ASL filter are used in combination (they are on two separate sheets), a notification must match either the expression filter or the ASL filter to be sent to the client.
- When multiple expression filters are specified on the same sheet, the notification must match each expression filter to be sent to the client.
- When multiple expression filters are specified on multiple sheets, the notification must match each expression filter on sheet 1 or each expression filter on sheet 2, and so on.

By default, there are two notification lists:

- Default — Specifies an expression filter.
- Maintenance — Does not specify a filter, it matches all notifications.

"About Default notification lists" on page 56 describes the default notification lists.

Creating a notification list

To create a notification list using the Global Manager Administration Console:

1. Select the **Launch Notification List Wizard** toolbar button.

   The **Notification List Creation Wizard** appears.

2. Type a unique name for the Notification List.

3. Choose to create a new notification list or copy an existing notification list:

   - If a new notification list is being created, all of the filter properties are empty.
   - If you copy a notification list, the notification list properties contain the same values as the copied list.

4. Click **Next**.

5. Create an expression filter or type the name of an ASL Filter prefaced by the name of the *BASEDIR*/smarts/local/rules directory in which it is located, and click **Next**.

   NOTICE

   The ASL filter must exist in the specified location, as the Global Manager loads the filter when you exit the wizard. The filter builder, used to create expression filters, is also used to configure tool and escalation policies. Chapter 10, "Working with Filters," provides additional information.

6. Edit the column headings that are displayed in the notification log. The left column lists the attributes included in a notification. The right column lists the columns names as they are currently displayed. You can edit the values in the right column by double-clicking a field.

7. Do one of the following:

   - Click **Next** to view the confirmation panel.
   - Click **Finish** to create the notification list.

   The new notification list is displayed in the Global Manager Administration Console.

Modifying a notification list

To edit the filters and the column headings for a notification list:

1. Select a notification list in the tree.

   The **Configure Notification List** panel appears.

2. Click **Edit Filter** to modify the filter.

   The **Edit Filter** dialog box appears. Chapter 10, "Working with Filters," provides additional information.

3. Click **OK** when you have finished editing the filters.
4. Edit the display headings as needed. Double-click the display heading in the right column to edit the text. The new display heading is visible when users reattach to the Global Manager.
5. Click **Apply**.

NOTICE

To provide a value for a UserDefined attribute, you must configure a hook script to populate the field. "User-defined notification attributes" on page 115 provides information about hook scripts.

Disabling a notification list

You can disable a notification list if you do not want it active but do not want to delete it. When users attach to a Global Manager and their notification is disabled, their console will not receive any notifications. The console will receive notifications when the notification list is enabled.

To disable a notification list:

1. Select the list from the tree in the **Global Manager Administration Console**.
2. Clear the **Enabled** checkbox.
3. Click **Apply**.

The name of the disabled notification list appears dimmed.

NOTICE

The Default notification list is used by the default-profile, admin-profile, and oper-profile user profiles and if this default notification list is disabled then user attached to this profile will not be able to receive any notification. Whenever changes are made to enable or disable of notification list through an admin profile, SAM server re-initializes all notifications to console. The resent notifications are based on changes made by the admin. So if the admin disables Default Notification List, when server re-initializes the connection, it will not find the default notification list or any other notification list attached to admin-profile. This will cause the console to not receive any notifications.

About Default notification lists

Two notification lists, Default and Maintenance, are included with the Global Manager and defined in the ics-default.xml file. The Default notification list is used by the default-profile, admin-profile, and oper-profile user profiles. If a user profile is disabled, users of that profile are assigned the Default user profile, which includes the notification list.

The Default notification list uses an expression filter that specifies the following conditions:

- Matches any notification in which the value of the Owner attribute is not maint.
- Matches any notification in which the value of Owner attribute is not SYSTEM.

A notification must match both conditions before it is sent to a client. The Maintenance notification list does not specify a filter. Because of this, it matches all notifications.

"About the predefined user profiles" on page 47 describes the purpose of the default notification lists and how they are incorporated into the default user profiles.

Processing notifications

The Service Assurance Manager processes notifications sent from:

- The Adapter Platform Manager (OI) to the INCHARGE-SA (SAM) server
- An Aggregate SAM server to a Presentation SAM server

If your SAM Server is communicating with an older version SAM, or OI, you must configure the usePre8.0 parameter included in the dxa-oi.conf file to true. Configuring the usePre8.0 parameter to true, instructs the SAM server to communicate with pre 8.0 SAM Servers and OIs.

To configure the SAM Server to communicate with older versions of SAM and OI:

1. Using a text editor, open the dxa-oi.conf file.
2. Locate the following lines of text:

   ```
   # Underlying OI is pre8.0, so we must use a different transport....
   usePre8.0 TRUE
   ```

3. Uncomment (delete the pound sign) the line, for example:

   ```
   Underlying OI is pre8.0, so we must use a different transport....
   usePre8.0 TRUE
   ```

4. Save and close the file.

If your Service Assurance Manager environment requires even faster notification processing, you can configure your Service Assurance Manager to provide multiple threads to process the notifications. Processing notifications using multiple threads adds an additional layer of complexity to your configuration, could possibly use additional system resources, and is only recommended for specific use if your system is failing to keep up with the notifications.

To configure your Service Assurance Manager to process notifications using multiple threads:

1. Using a text editor, open the dxa-oi.conf file.
2. Locate the following lines of text:

```
# Subscribe to a Notification List
# NLName/n
sub     Default/n
```

3. Uncomment (delete the pound sign) the line.
4. Add the number of threads, for example:

```
Subscribe to a Notification List
NLName/n
sub     Default/n4
```

   In this example, the SAM Server creates four threads to process the notifications from the adapter platform.

5. Save and close the file.

Notification polling and Property Change

The SAM Console polls the SAM server to request batched notification updates at a default polling interval of 10 seconds. Therefore, operations that update notifications are shown with an average delay of a few seconds.

In SAM 8.0 and above, an option has been added to the SAM Console to choose between Notification polling (HyperNotification) and PropertyChange (subscription based notification updates from server) notification updates.

Notification Polling

Notification polling **provides better performance over a simple Notification transfer.** To enable Notification polling, set the new Console property *com.smarts.disableHyperNotification* to **FALSE** by typing the following in the command line prompt:

sm_gui -Dcom.smarts.disableHyperNotification=False

By default, this property is set to **FALSE** making Notification polling the default form of receiving notification updates.

Notifications are fetched in bulk at regular intervals. This interval can be changed by setting the environment variable, **SM_SAM_TO_SAM_HYPER_NOTIF_SLEEP** in the runcmd_env.sh file. The default value for SM_SAM_TO_SAM_HYPER_NOTIF_SLEEP parameter is 10. Setting the value to 5 will reduce the interval for fetching notifications and any mismatch in notification timestamp in a hierarchical environment.

For example, to set an interval of 5 seconds, add the following line in the runcmd_env.sh file:

```
SM_SAM_TO_SAM_HYPER_NOTIF_SLEEP=5
```

NOTICE

If the interval is set to less than 5 seconds, the performance of the SAM server is impacted. Hence, an interval of less than 5 seconds is not recommended.

Polling Change

To enable PropertyChange notification updates, set the Console property *com.smarts.disableHyperNotification* to **TRUE** by typing the following in the command line prompt:

sm_gui -Dcom.smarts.disableHyperNotification=True

Setting this parameter to True, also helps avoid creation of multiple entries for a single session connected using dmctl.

The -Dcom.smarts.disableHyperNotification parameter was introduced in the SAM 8.1 HF 3 release. The hyper notification feature allows the console to poll the SAM server to request batches of notification updates. The feature requires dedicated separate connections between the server and console. While hyper notification increases the number of connections between the server and console for SAM 8.x versus SAM 7.x, it at the same time improves the console performance and notification processing.

It is recommended that hyper notification is not disabled as it would put additional load on the server and console, resulting in degradation of SAM performance. Exercise caution while using the -Dcom.smarts.disableHyperNotification parameter.

Configuring the Maintenance and Business Impact Manager

To configure the MBIM- Maintenance and Business Impact Manager:

1. Start the MBIM domain manager server.
2. Configure an MBIM domain.

   "Configuring domains" on page 88 provides detailed information for configuring domains and reconfiguring SAM.

   Connecting SAM to the MBIM domain provides automatic MBIM notifications. When MBIM is disconnected, a notification appears in the console and the maintenance adapter is automatically disabled.
3. From the **Domain Manager Administration Console**, attach to the MBIM domain.
4. Select the MBIM object (the root of the display).
5. Click **Topology** › **Add Source**.
6. Type the instance of SAM to use as a source of topology for the MBIM domain.

Schedule Maintenance option

To inform fellow operators that a specific managed element is entering its scheduled maintenance period, the In Maintenance option for a specific notification is used.

Enabling maintenance tracking

To enable maintenance tracking:

1. Log in to the console.
2. In the **Topology Browser Console**, right-click on the name of the Service Assurance Manager server at the top of the topology tree which is the topology source for MBIM.
3. Click **Enable Maintenance**.

Disabling maintenance tracking

To disable maintenance tracking:

1. Log in to the console.
2. In the **Topology Browser Console**, right-click on the name of the Service Assurance Manager server at the top of the topology tree which is the topology source for MBIM.
3. Click **Disable Maintenance**.

Console operations

Console operations are user actions on the console and determine a user's access privileges to various functions and commands provided by the Global Console. Examples of console operations include taking ownership or acknowledging a notification, adding a view to the console, or viewing the details of a notification. An administrator can assign one or more of these tasks to a user profile.

Interaction with EMC Smarts username/password privileges

An administrator needs All privileges, conferred by the Global Manager's serverConnect.conf file and console operations access privileges to perform the following administrative tasks:

- Display the Configure menu in the Global Console
  - Open the Global Manager Administration Console
  - Open the Domain Manager Administration Console
  - Open the Group Definition Console
  - Open the Topology Builder Console
  - Edit Map Icons
- Save View as Dashboard viewlet
- Edit attributes in underlying domains
- Select Correlate Now or Recompute Codebook in underlying domains
- Manage or unmanage elements in underlying domains

With console operations, an administrator can assign one or more of these tasks to a user profile.

NOTICE

Some features require a separate license, such as Business Services Maps. An administrator can assign any feature to a user, but the feature may not be enabled.

Default user profiles

The user profiles included with Service Assurance support the following console operations:

- The admin-profile includes all console operations groups, giving users assigned to this profile administrative access.
- The maint-profile includes the Detail Information, Map, Views, and Event Management console operations groups, giving users operator access.
- The default-profile includes the Detail Information, Map, and Views console operations groups, giving users read-only access.
- The oper-profile includes the Detail Information, Map, Views, and Event Management console operations groups, giving users operator access.

Console operations and groups

Table 4 on page 60 lists the console operations and their assigned group. To facilitate the configuration of user profiles, the following sets of groups are provided:

- Read Only includes the Detail Information, Map, and Views groups
- Operator includes the groups in Read Only and Event Management
- Administrator includes all groups

When one of these sets is selected, the associated groups and operations are automatically selected.

The administrator can then customize the selected groups or operations as described in "Creating a user profile" on page 41.

Table 4 Console operations and groups (page 1 of 4)

Group/Console operation	Description
Detail Information	
Browse	Opens a Topology Browser view from the Global Manager and displays the element as the root.
Browse Detail	Opens a Topology Browser view from the underlying Domain Manager and displays the element as the root.
Expand Map Node	Expands a map node to display connected nodes.
Show Containment	Displays an element's components.
Map	

Table 4 Console operations and groups (page 2 of 4)

Group/Console operation	Description
Physical Connectivity	Opens a map displaying physical connectivity.
IP Connectivity	Opens a map displaying IP connectivity.
VLAN Connectivity	Opens a map displaying VLAN connectivity.
Membership	Opens a map showing group elements (IP or VLAN).
Business Services	Opens a map displaying business elements.
Application	Opens a map displaying application elements.
Network Protocol Manager Connectivity	Opens a map that depicts topology and connectivity for: • BGP Connectivity • OSPF Connectivity • OSPF Network Connectivity • EIGRP Connectivity • EIGRP Network Connectivity • ISIS Connectivity • ISIS Network Connectivity
MPLS Connectivity	Opens a map that depicts MPLS topology.
Multicast Connectivity	Opens a map that depicts Multicast topology.
Multicast	A Multicast map depicts multicast topology.
Storage Network Connectivity	Opens a map that depicts storage network topology.
Views	
Notification Log	Provides the ability to access a Notification Log view.
Topology Browser	Provides the ability to access a Topology Browser view.
Summary	Provides the ability to access a Summary view.
Status Table	Provides the ability to access a Status Table view.
Map	Open a Map Console.
View Configuration	
Notification View Filter	Create a filter for a Notification Log.
Summary Filter	Create a filter for Summary Views.
Edit Summary Parameters	Edit a Summary View configuration.
Add Summary	Add a Summary to a Summary View.
Delete Summary	Remove a Summary from a Summary View.
Status Table Configuration	Edit Status Table configuration.
Console	

Table 4 Console operations and groups (page 3 of 4)

Group/Console operation	Description
Save	Save the console to wherever it was originally loaded from, local or remote. If a console has not been previously saved, the system will prompt the user for information on where to save the console.
Save Remote As	Save a console to the Global Manager.
Open Local	Open a console from the local system.
Open Remote	Open a Console from the Global Manager.
InCharge Manager Attach	Attach to a Domain Manager.
InCharge Manager Detach	Detach from a Domain Manager.
Views Management	
Add View	Provides the ability to add a view to a console, and also open a new console (for example, Notification Log Console). **Notice:** You can only add views or open a console for views for which you have permission to access.
Delete View	Remove a view from the console.
Copy View	Copy a view in the console.
Copy View as New Console	Copy a view in the console and open as a new console.
Set View Context Listening	Set listening context for active view.
Event Management	
Take Ownership	Take ownership of a notification.
Release Ownership	Release ownership of a notification.
Schedule Maintenance	Schedule maintenance on a managed element.
Acknowledge	Acknowledge a notification.
Unacknowledge	Unacknowledge a notification.
Notification Properties	Open a notification properties dialog or add notification properties view to console.
Administration	
Configure Group	Open a Groups Definition Console.
Configure Global Manager Administration Console	Open a Global Manager Administration Console.
Configure Domain Manager Administration Console	Open a Domain Manager Administration Console.
Configure Map Icons	Specify custom icons for map nodes using the Global Manager Administration Console.
Launch Topology Builder	Open the Topology Builder Console.

Table 4 Console operations and groups (page 4 of 4)

Group/Console operation	Description
Save View as Viewlet	Save a view as a viewlet for the InCharge Business Dashboard.

Console operations allowed for all users

There are additional console operations that are always available and cannot be removed from the console. Table 5 on page 63 lists of the more useful of these console operations.

NOTICE

There are other operations that cannot be removed from the console that are not included in this table.

Table 5 Console operations allowed for all users

Console view/console operation	Description
Notification Log View/Console	
Sort Columns	Sort notifications by selected column.
Select Columns	Select columns to display in the log.
Save log contents to file	Save the log to a file on the local system.
Freeze Display	Freeze the log to prevent display of new notifications.
Summary View/Console	
Layout	Set the number of summaries per row.
Move Summary Left	Move a summary one summary to the left.
Move Summary Right	Move a summary one summary to the right.
Topology Browser View/Console	
Select	Control the set of classes or instances displayed in the tree.
Refresh tree	Refresh the tree view.
Make Root	Make the selected instance the root of the tree.
Map View/Console	
Save map	Save a map to the Global Manager.
Edit map filter	Edit classes displayed in a map.
Set background	Load a background image for a map.

CHAPTER 4
System Configuration for the Global Manager

This chapter contains the following information:

- Defining Domain Manager parameters 66
- Defining system defaults 69
- Configuring business templates 74
- Configuring Global Manager Administration Console preferences 75
- Configuring IPv6 Availability Manager connections 77
- Audit logging 78
- SAM performance throughput report 82
- Notification lifecycle auditing 84
- Configuring weights.conf for OTM 85
- Hierarchical SAM configuration 86

Defining Domain Manager parameters

A Domain Manager refers to an underlying analysis application, such as IP Availability Manager, Storage Insight for Availability, Adapter Platform, or other Global Manager that serves as a source of event and topology data. You specify these domains and the parameters that control the import of event and topology data in the domainconfig section of the ics-default.xml file.

The domainconfig section is divided into one or more domaintype subsections, each of which defines the configuration for one or more underlying domains. The following example shows the syntax of a domainconfig section when the underlying domain is IP Availability Manager:

```
<domainconfig>
        <domaintype
            name="AM-Suite"
            description="Type for AM-Suite"
            dxaConfFile="dxa-conn.conf"
            MinimumCertainty="0.01"
            SmoothingInterval="65">
            <hook enable="False">ics/dxa-sample-hook.asl</hook>
            <domainname>INCHARGE-AM-1</domainname>
            <domainname>INCHARGE-AM-2</domainname>
        </domaintype>
```

Table 6 on page 66 describes the domainconfig parameters.

Table 6 Parameters defining the domainconfig section (page 1 of 2)

Parameter	Description
name	Defines the domain name.
description	Defines the description of the domaintype.
dxaConfFile	Defines the data exchange file that corresponds to the underlying domain. A data exchange file ensures that Global Manager receives the correct event and topology data. Table 7 on page 67 specifies the correct data exchange file for each product.
MinimumCertainty	Defines the minimum value that the Certainty attribute must have before an event is sent to the Global Manager from the underlying domain. Events with a Certainty value below the threshold are discarded. This value must be a number between 0.0 and 0.99. The default value is 0.01 for IP Availability Manager and IP Performance Manager. Otherwise the default is 0.24. **Notice:** The MinimumCertainity parameter does not apply to underlying Adapter Platform servers or other Global Managers.

Table 6 Parameters defining the domainconfig section (page 2 of 2)

Parameter	Description
SmoothingInterval	Defines the time, in seconds, an event must be active before it is sent to the Global Manager. The default value is 65 seconds. **Notice:** The SmoothingInterval parameter does not apply to underlying Adapter Platform servers or other Global Managers.
hook	[Optional] Defines the name of an ASL script that modifies a notification. Typically, this is used to add information to one of the user-defined fields of a notification. Hook scripts must be located in the BASEDIR/smarts/local/rules directory. You must prefix the name of the script with the directory in which it is located, typically this is the ics directory. For example, the ics/dxa-sample-hook.asl hook script is located in the BASEDIR/smarts/local/rules/ics directory. **Notice:** The default ASL hook scripts are placeholders for a script you must provide. A hook script is not required unless you need to perform additional processing on the notifications coming from the underlying domain. "User-defined notification attributes" on page 115 describes possible uses for hook scripts.
domainname	Defines the name of the underlying domain. You can specify multiple domains with the same configuration by adding name parameters to the domaintype section. However, the value of each name parameter within a domaintype section must be unique.

Data exchange files or underlying domains

Table 7 on page 67 lists the data exchange files that should be used for each product.

Table 7 Data exchange file for each product (page 1 of 3)

Product	Component	Data exchange file
Service Assurance Manager	Global Manager	dxa-sam.conf
	Adapter Platform	dxa-oi.conf
	Adapter for BMC Patrol	dxa-bmc.conf
	SDI Adapter	dxa-sdi.conf
	MBIM server	dxa-sam-from-mbim.conf
IP Manager 7.0	IP Availability Manager	dxa-conn.conf
	IP Performance Manager and IP Server Performance Manager	dxa-perf.con
	IP Availability Manager and IP Performance Manager	dxa-conn-perf.conf
	IP Discovery Manager	dxa-dm.conf

Table 7 Data exchange file for each product (page 2 of 3)

Product	Component	Data exchange file
IP Manager 6.x	IP Availability Manager	dxa6X-conn.conf
	IP Performance Manager and IP Server Performance Manager	dxa6X-perf.conf
	IP Availability Manager and IP Performance Manager	dxa6X-conn-perf.conf
	IP Discovery Manager	dxa6X-dm.conf
IP Manager 4.x	IP Availability Manager	dxa4X-conn.conf
	IP Performance Manager	dxa4X-perf.conf
	IP Availability Manager and IP Performance Manager	dxa4X-conn-perf.conf
Network Protocol Manager	Network Protocol Manager for BGP	dxa-bgp.conf
	Network Protocol Manager for OSPF	dxa-ospf.conf
	Network Protocol Manager for EIGRP	dxa-eigrp.conf
	Network Protocol Manager for IS-IS	dxa-isis.conf
Storage Insight Manager	Storage Insight for Availability	dxa-storage-analysis.conf dxa-storage-topology.conf
VoIP Availability Manager	VoIP Availability Manager	dxa-sam-from-voip.conf
Application Insight	Application Connectivity Monitor	dxa-sam-from-acm.conf
	Application Insight	dxa-sam-from-ai.conf
MPLS Manager	MPLS Manager	dxa-mpls-vpn-ics-analysis.conf
		dxa-mpls-vpn-ics-monitoring.conf
		dxa-mpls-vpn-ics-topology.conf
		dxa-mpls-vpn-ics.conf
ATM/Frame Relay Availability Manager	ATM/Frame Relay Availability Manager	dxa-sam-from-atmfr.conf
	XD Manager for ATM/FR-to-IP AM	dxa-sam-from-xdatmfrip.conf
Multicast Manager	Multicast Manager	dxa-mcast.conf
Optical Transport Manager	Optical Transport Manager for SONET	dxa-sam-from-sonet.conf
	Optical Transport Manager for WDM	dxa-sam-from-wdm.conf
	XD Manager OTM to IP	dxa-xd-otm-ip.conf
Application Connectivity Monitor 1.0	Application Connectivity Monitor	dxa-app-poller.conf
Application Services Manager 1.0	Application Services Manager	dxa-asm10.conf
Application Services Manager 1.1	Application Services Manager	dxa-asm.conf

Table 7 Data exchange file for each product (page 3 of 3)

Product	Component	Data exchange file
CiscoWorks Device Fault Manager	CiscoWorks Device Fault Manager	dxa-dfm.conf
CiscoWorks Voice Health Monitor	CiscoWorks Voice Health Monitor	dxa-vhm.conf
Security Infrastructure Manager	Security Infrastructure Manager	dxa-sim.conf

Defining system defaults

System defaults are system-wide settings that affect the Global Manager and its clients.

To configure system defaults:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration.**
4. **Click System Defaults**.

   The **Configure System Defaults** interface appears, as illustrated in Figure 3 on page 69.

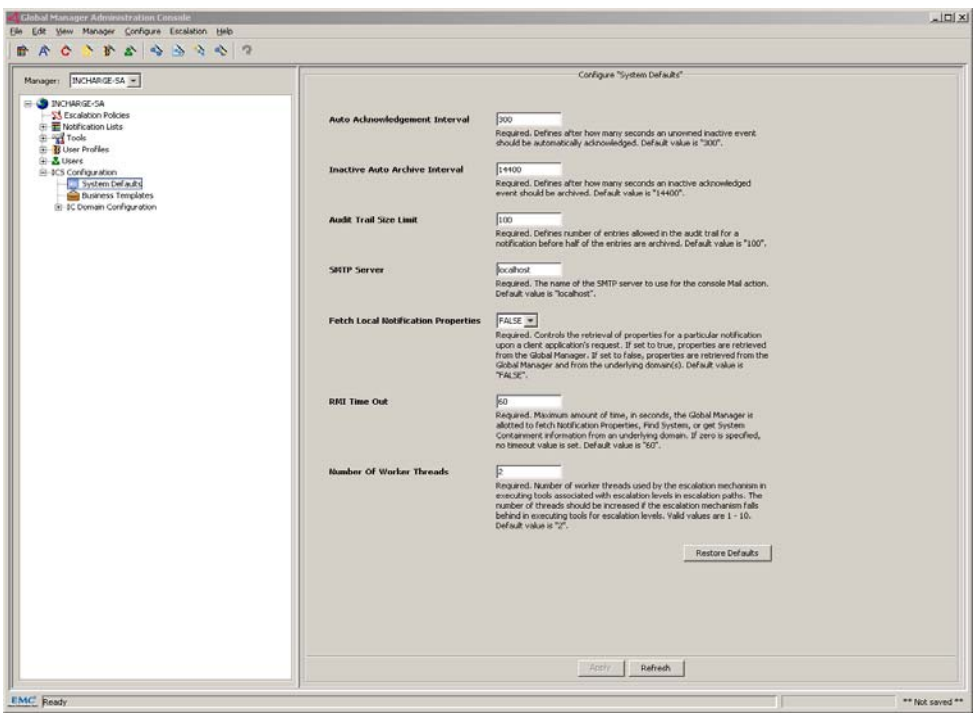

Figure 3 Global Manager Administration Console—Configure System Defaults

Table 8 on page 70 describes the System Defaults parameters.

Table 8 System Defaults parameters (page 1 of 3)

Parameter	Description
Auto Acknowledgement Interval	Defines interval, in seconds, after which an inactive and unowned notification is acknowledged. Notifications that are acknowledged by the Global Manager are owned by the user SYSTEM. **Default**: 300 seconds
Inactive Auto Archive Interval	Defines interval, in seconds, after which an inactive and acknowledged notification is archived. If this value is set to zero, archiving is disabled and notifications will not be deleted, causing Global Manager to use more memory. **Default**: 14400 seconds (4 hours)
Audit Trail Size Limit	Defines the number of audit log entries for each notification that are saved and visible in the Global Console before the log contents are archived. When this limit is reached, half of the entries are written to the notification archive. **Default**: 100 entries
SMTP Server	Defines the name of the SMTP mail server through which mail messages sent by the Global Console's Mail tool are sent. This mail server must be reachable from the host where the Global Manager is running. This value can be overridden in the Global Console. **Default**: "localhost"
Fetch Local Notification Properties	Defines how the properties of a particular notification are retrieved when requested by a client application. If set to "true", properties are retrieved from the Global Manager. If set to "false", properties are retrieved from the Global Manager and from the underlying domain. **Notice:** The values configured for this parameter affect the performance of the Service Assurance Manager. Client applications include the Global Console, the dmctl utility, and adapters. **Default**: False
RMI Time Out	Defines the maximum amount of time, in seconds, the Global Manager is allotted to retrieve Notification Properties, Find System, or get System Containment information from an underlying domain. If a negative value is specified, no time-out value is set. **Default**: 60 seconds
Number Of Worker Threads	Defines the number of worker threads used by the escalation mechanism in executing tools associated with escalation levels in escalation paths. Increase the number of threads if the escalation mechanism falls behind in executing tools for escalation levels. Valid values are 1 - 10. **Default**: 2

Table 8 System Defaults parameters (page 2 of 3)

Parameter	Description
Broadcast Event State on Restart	Defines whether notifications that result from internal Global Manager events are propagated to clients that subscribe to them. Internal Global Manager events include system disconnects and system reconfigurations. If set to True, notifications resulting from internal Global Manager events are propagated. If set to False, the events are not propagated. When configuring this parameter, the value must be identical for all Service Assurance Manager servers within your Service Assurance Manager deployment. For changes to this parameter configuration to take effect, the Service Assurance Manager servers must be restarted. **Notice:** For optimum benefit, configure this parameter as early in your Service Assurance Manager deployment as possible, before all source domains are added to the Service Assurance Manager server. The quickest way to apply this parameter configuration is to restart the Service Assurance Manager server. However, if you are configuring this parameter in a production environment, apply the parameter configuration to the Service Assurance Manager server by making a minimal change to the configuration of each of the underlying source domain types, and clicking the **Reconfigure** button. One example of a minimal configuration change is to modify the value for the Smoothing Interval parameter. A minimal configuration change is required for each domain type. This results in the Service Assurance Manager server disconnecting from each underlying domain, rereading all the settings including the new modification, and then reconnecting to the underlying domains. **Default**: True
Hierarchical Acknowledge	Defines whether acknowledging or unacknowledging a notification in the server sends the acknowledge (or unacknowledged) information to the underlying servers. If set to True, the notification in the underlying servers are also acknowledged or unacknowledged. If set to False, the notification in the underlying servers are not changed. **Default**: True
Companion UI Hostname	Defines the location of the related Companion UI vApp. The accepted values are the Domain Name System (DNS) host or the IP address of the vApp (for example, companionUI or companionUI.mycompany.com).

Table 8 System Defaults parameters (page 3 of 3)

Parameter	Description
Use DNS Lookup in This Domain	Defines whether the system performs Domain Name System (DNS) searches when searching for computer systems. If set to True, the system queries the DNS in this domain. **Notice:** When performing a DNS search, perform the search from the Adapter Platform (OI). **Default**: True
Use DNS Lookup in Underlying Domains	Defines whether the system uses Domain Name when searching for computer systems. If set to True, the system queries the DNS in underlying domains. **Default**: True
Use DNS Domains in DNS Lookup	Defines whether DNS domains are used when the system performs a DNS search. If set to false, only original domain names are used during the DNS search. **Default**: False
DNS Domain Names	Defines the list of domain names the system uses when performing a DNS search. **Notice:** When entering multiple domain names, separate each domain name with a comma.
Underlying Domains Order	Defines the order in which the underlying domains are searched. **Notice:** When entering multiple domain names, separate each domain name with a comma.
Fully Qualified Domain Name Suffixes	Defines which domain names returned as a result of performing a DNS search are considered to be fully qualified. Domain names matching suffixes entered in this parameter are considered fully qualified.

5. Type or select your system defaults configuration options.
6. Click **Apply**.

NOTICE

System defaults cannot be deleted. EMC Corporation cannot guarantee that any repository modification made since the last time the repository was saved will be implemented during a Windows Service restart. To guarantee that repository modifications are implemented during a Windows Service restart, you must shut down the service completely and then restart the service.

About acknowledging notifications

When a notification is acknowledged, an entry is appended to the notification's Audit Log. In addition, the values of two notification attributes are set: Owner and Acknowledged. The value of the Owner attribute is set to the username of whomever acknowledged the event. The value of the Acknowledged attribute is set to TRUE.

If a notification is unacknowledged, another entry is appended to the notification's audit log. The value of the Owner attribute is set to the EMC Smarts username of whomever unacknowledged the notification and the value of the Acknowledged attribute is set to FALSE.

Auto acknowledgement is a facility designed for notifications that clear before an operator is able to acknowledge them. Usually a notification will clear shortly after it appears. Instead of requiring operators to manually acknowledge such notifications, the Global Manager automatically acknowledges unowned notifications after they remain inactive (clear) for the time specified by the AutoAcknowledgementInterval. Notifications acknowledged by the Global Manager have their Owner attribute set to SYSTEM.

After a notification is acknowledged, it is eligible for archiving. The Global Manager will not archive an active or unacknowledged notification.

Archiving notifications

Old notifications are periodically removed from the Global Manager's repository and archived to a file. The notification archive includes the values of the notification's attributes at the time it is archived, including the contents of the audit log. This archive is intended to provide a record of information before it is deleted.

The *EMC Smarts Report Manager User Guide* describes how to generate historical reports based on notification data. Report Manager does not use the notification archive for reporting, but maintains the notification information in a relational database.

Archived notifications are written to the file *‹global_manager›*.archive, where *‹global_manager›* is the name of the Global Manager. By default, this is INCHARGE-SA so the resulting notification archive is named INCHARGE-SA.archive. This file is written to the *BASEDIR*/smarts/local/logs directory.

NOTICE

Over a period of time, a busy Global Manager can generate a sizeable archive file. It may be necessary to periodically rotate the notification archive file.

Configuring business templates

Business templates are used to load business entities. Two types of business entities are topology-group/service data files and Business Impact Manager weights files.

Both topology-group/service data files and Business Impact Manager weights data can be loaded using text or XML files.

To configure business templates:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Click **Business Templates**.

   The **Configure Business Templates** interface appears, as illustrated in Figure 4 on page 74.

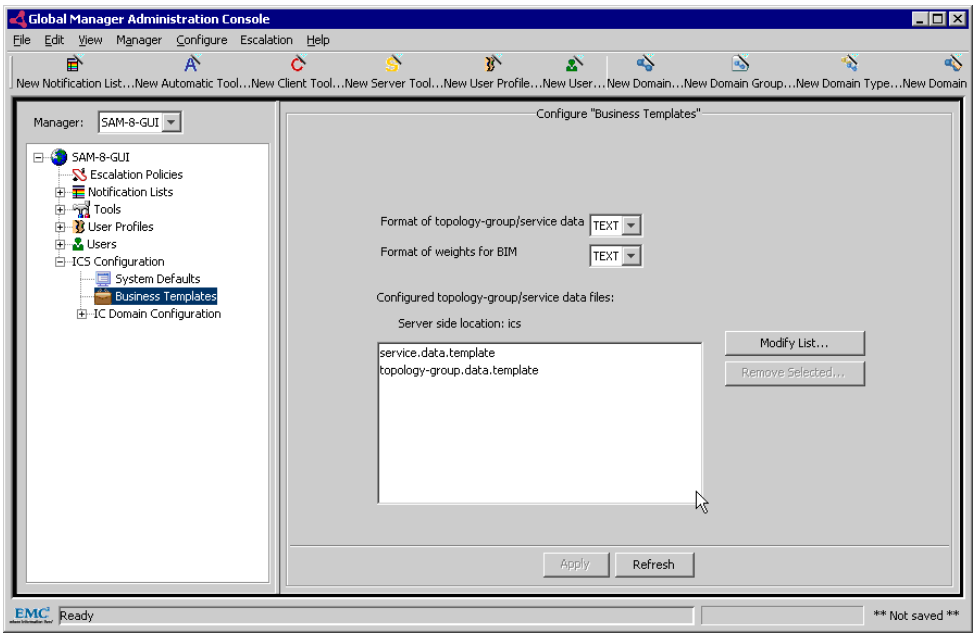

Figure 4 Global Manager Administration Console—Configure Business Templates

5. Select the format for the type of data file being configured.

   NOTICE

   TEXT is the default format file type for both topology-group/service data files and the Business Impact Manager weights files.

6. Click **Modify List** to add business template files.

   NOTICE

   Click **Remove Selected** to delete a selected business template file from the list.

7. Click **Apply**.

   NOTICE

   Business templates cannot be deleted.

Configuring Global Manager Administration Console preferences

The following preferences are configured using the Global Manager Administration Console Preferences dialog box:

- DXA configuration file cannot be found at the server location
- Hook script cannot be found at the server location
- Business template file cannot be found at the server location
- Prompt for confirmation when closing the console with a server which requires reconfiguration
- Prompt for confirmation when exiting the application with a server which requires reconfiguration
- Remind to reconfigure after changing Domain configuration
- Remind to reconfigure after changing Domain Group configuration
- Remind to reconfigure after changing Domain Type configuration
- Remind to reconfigure after changing Domain Tag configuration

To configure Global Manager Administration Console Preferences:

1. From the **Global Manager Administration Console**, select **View** › **Global Administration Console Preferences**.

   The **Global Administration Console Preferences** dialog box appears, as illustrated in Figure 5 on page 76.

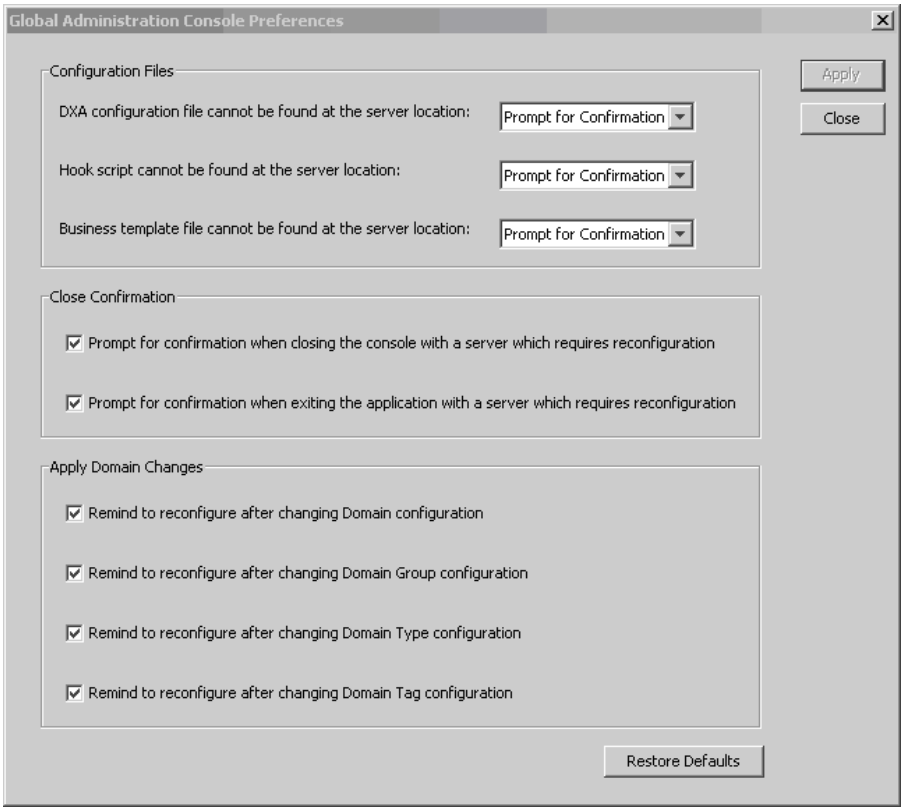

Figure 5 Global Administration Console Preferences dialog box

2. Select or click your desired configuration options.
3. Click **Apply**.

NOTICE

Click **Restore Defaults** to restore the default configuration settings.

Configuring IPv6 Availability Manager connections

The SM_IP_VERSIONS environment variable allows you to define the Internet Protocol (IP) version used when SAM is running in an IPv6 network.

To configure this variable:

1. Using a text editor, edit the runcmd_env.sh file. The runcmd_env.sh file is located in the *BASEDIR*/smarts/local/conf/ directory.
2. Add the following line:

   SM_IP_VERSIONS=<*value*>

   The available values for the variable are:

   - V4 — IPv4 connections only. This is the default value.
   - V6 — IPv6 connections only.
   - V4V6 — IPv4 connections preferred over IPv6 connections.
   - V6V4 — IPv6 connections preferred over IPv4 connections.
3. Save and close the file.
4. Edit the com.smarts.ip_versions Java property. The value for this property must match the value entered for the SM_IP_VERSIONS environment variable.

The *EMC Smarts System Administration Guide* provides detailed environment variable information.

Examples of specifying the IP version

Code examples that specify the IP version when connection, include:

```
frame:v4
frame:v6
frame:v4v6
frame:v6v4
frame:v6:3026    ## 3026 is the port
[1:1::1111]:3026 ## 3026 is the port
```

The IP version in the following example is redundant but allowed:

```
[1:1::1111]:v6:3026 ## 3026 is the port
[1:1::1111]:v4v6:3026 ## 3026 is the port
```

Audit logging

Entries written to the audit log include specific user interactions with EMC Smarts analysis servers.

Audit logging is enabled on the Service Assurance Manager and analysis servers when a modification is made to the servers from the Global Console.

Actions that a user may perform in the Global Manager Administration Console that are captured in the audit logs include:

- Escalation Policies
  - Add an escalation policy
  - Delete an escalation policy
  - Modify an escalation policy
  - Disable an escalation policy
  - Enable an escalation policy
- Notification Lists
  - Add a new notification list
  - Delete a notification list
  - Modify a notification list
    - Modify filters
    - Modify display headings
  - Disable a notification list
  - Enable a notification list
- Tools
  - Add a new tool
  - Delete a tool
  - Modify a tool
  - Allow different user profiles access to the tool
  - Disable a tool
  - Enable a tool
- User Profiles
  - Add a user profile
  - Delete a user profile

- Modify a user profile
  - Modify notification list
  - Modify users associated with profile
  - Modify tools the user profile can access
  - Modify default console
  - Modify console operations the user profile can access
- Disable a user profile
- Enable a user profile

This section includes the following topics:

- "Configuring the Audit Log Adapter" on page 79
- "Starting the Audit Log Adapter" on page 80
- "Starting a new log file" on page 80
- "Filtering the Audit Log" on page 81

Configuring the Audit Log Adapter

To generate audit log output on a server you must add the SM_AUDIT_CLIENT_ACTIONS environment variable to the runcmd_env.sh file. The runcmd_env.sh file is located in the *BASEDIR*/smarts/local/conf directory.

To add the SM_AUDIT_CLIENT_ACTIONS environment variable to the runcmd_env.sh file:

1. Invoke the **sm_edit** utility from the *BASEDIR*/smarts/bin directory:

   ```
   sm_edit conf/runcmd_env.sh
   ```

   NOTICE

   If multiple EMC Smarts products are running on the same host, you should ensure you invoke the **sm_edit** utility from the bin directory of the product whose files you want to edit. The *EMC Smarts Service Assurance Manager Configuration Guide* contains additional **sm_edit** utility information.

2. Add the following environment variables and corresponding values using the following syntax:

   ```
   SM_AUDIT_CLIENT_ACTIONS=TRUE
   SM_AUDIT_ALL_CONNECTIONS=1
   ```

   NOTICE

   Set the SM_AUDIT_CLIENT_ACTIONS environment variable to true to have all console actions entered into the audit logs. Set the SM_AUDIT_ALL_CONNECTIONS environment variable to 1 to have all operations including command line actions entered into the audit logs.

3. Save and close the runcmd_env.sh file.
4. Restart any EMC Smarts program currently running for the new environment variables to take effect.

Starting the Audit Log Adapter

To start the Audit Log Adapter, type the following command:

```
▼BASEDIR/smarts/bin/sm_server -n <adapter_name>
--config=audit-parse --output▲
```

Where:

- *‹adapter_name›* is the name of the Audit Log Adapter. It is recommended that you name the Audit log Adapter relative to the server for which it is performing audit parsing, for example, INCHARGE-SAM-AUDIT.

NOTICE

▼▲ Indicates that this command must be typed as one line. Separate each option with a space when typing the command.

To name the Audit Log Adapter a name other than ‹*adapter_name*›:

1. Copy the Sam/Smarts/conf/audit-parse/adapter-name.conf file to the sam/smarts/local/conf/audit-parse directory.
2. Rename the adapter-name.conf file with the same name as the Audit Log Adapter.

   For example, if the Audit Log Adapter name is Audit-log, the adapter-name.conf file should be renamed Audit-log.conf.

Starting a new log file

The roll_sam_auditlog command is invoked through the dmctl utility, which requires that you attach to a Domain Manager or adapter with administrative privileges. The syntax of the command is:

```
serverName exec roll_sam_auditlog
```

For example, on Windows to start a new log file for a Domain Manager named InCharge, type the following command and press **Enter**:

```
C:\InCharge\SAM\smarts\bin\dmctl -s InCharge exec roll_sam_auditlog
```

On UNIX systems, the equivalent command is:

```
opt/InCharge/SAM/smarts/bin/dmctl -s InCharge exec roll_sam_auditlog
```

When roll_sam_auditlog is invoked, the Domain Manager or adapter writes an informational message to the end of the current log file, repeats the steps it executed at startup, and creates a new log file and a backup log file. All subsequent logging information is recorded to the new file.

Filtering the Audit Log

By default, the SAM audit log performs an audit and logs SAM functions in the .audit file without filtering. When logging of client actions is enabled, the audit log provides information about client actions that changed the state of the server, such as put, insert, remove or delete requests. The audit ignores requests for information that do not change the state of the server, for example, get requests.

Many operations do not change the state of a server, and hence do not need to be logged or audited. The conf/ics/sam-audit-filter.import file contains a list of operations that are known to not change the state of the server. If a new operation is added that does not change the state of the server or if you do not want a particular operation logged, then the operation may be added to the file to suppress logging that operation when it is invoked. Each operation to be suppressed is listed on a separate line as a string containing the class name, two colons, and the operation name.

In order to exclude redundant functions from the .audit file, a filtering criteria is available. To exclude functions from the audit log, add them to the conf/ics/sam-audit-filter.import file. After the file is updated, restart SAM.

SAM performance throughput report

SAM performance throughput report provides users with information on notifications processed, notified, cleared and changed within a specified time interval. The report explains the possible slowdown in SAM performance and provides a better view into the health of SAM. The report, referred to as the ‹SAM-SERVER-NAME›.perf file, is available under the ‹*BASEDIR*›/smarts/local/logs directory.

This report provides information on notification processing attributes such as:

- Average notification processing time
- Average impact refresh calculation time
- Number of topology synchronizations that have occurred during the default monitoring duration
- Average time for processing hook files
- Number of connected events
- Number of disconnect events
- Number of suspended events
- Number of cleared events
- Number of timer alarms
- Number of events with heart beats
- Number of rejected events
- Number of accepted events
- Number of notified events
- Average time taken to push an event to the NOTIF processing queue

The monitoring period for the report is controlled by the PERF_ARCHIVE_INTERVAL environment variable. The default time interval for this variable is set to 3600 seconds (that is, one hour). At an interval of every one hour, the above mentioned performance statistics are written to the ‹SAM-SERVER-NAME›.perf file.

Each record in the file is in the following format:

```
<DATA-TIME> - <DOMAIN-MANAGER-NAME> : Performance metrics
```

Provided below is an example of the SAM performance throughput report:

```
December 19, 2011 4:23:19 PM GMT+05:30 - EMC-IT :
Avg_NotificationProcess_Time = 0     Avg_ImpactRefresh_Time = 0
No_Toposync = 0 Property_Control = 0Attrib_Change = 0       Connect
= 0    Suspend = 0    Clear = 0      Avg_HookFileProcess_Time = 0
Avg_NotifProcess_Time = 0       Timer_Alarm = 60    Disconnect = 0
Heart_Beat = 0  Reject = 0      Accept = 0      Notify = 0
Total_Count = 60
```

You can modify the default time interval set for the PERF_ARCHIVE_INTERVAL environment variable in the ics-event-processing.asl and the ics-nl-processing.asl files. These files are located under the ‹*BASEDIR*›/local/rules/ics folder or the ‹*BASEDIR*›/rules/ics folder. The

modification must be carried out in both the ics-event-processing.asl and the ics-nl-processing.asl files. Use the **sm_edit** utility to modify the values to these files. For example:

```
BASEDIR/smarts/bin>sm_edit ../rules/ics/ics-event-processing.asl
```

You must reconfigure the domain for the changes to take effect.

Performance threshold event in SAM

As part of performance monitoring, few threshold values have been defined in the DEFINE_PERF_THRESHOLD_VALUES rule, available in the ics-event-processing.asl and ics-nl-processing.asl files. You can find these ASL files in the ‹*BASEDIR*›/local/rules/ics folder or the ‹*BASEDIR*›/rules/ics folder. For example, the default number of threads for disconnected events in an hour is one. If case more than one disconnected event is found, SAM generates a performance monitoring severity 2 notification in SAM with relevant details. This notification will get auto-cleared after an hour as defined in the PERF_ARCHIVE_INTERVAL environment variable.

Table 9 on page 83 lists the SAM performance monitoring threshold setting parameters. This feature is not enabled by default. You can enable this feature by setting the value for the PERF_THRESHOLD_EVENT flag to TRUE in both the ics-event-processing.asl and ics-nl-processing.asl files.

Table 9 **SAM performance monitoring threshold setting parameters**

Parameter	Default Values
Number of Topology Synchronization	The maximum is one topology synchronization operation every 10 minutes for a given domain manager.
Number of disconnect Events	The maximum is one disconnected event in an hour for a given domain manager.
Average Impact Refresh Calculation Time	The average time taken must not be more than 5 seconds.
Average Notification Processing Time	The average time taken must not be more than one second.
Average Hook file processing Time	The average time taken must not be more than two seconds.
Average Time taken to push an Event to NOTIF processing queue	The average time taken must not be more than two seconds.

You can modify the default values for the performance monitoring threshold parameters by modifying the values for the parameters both in the ics-event-processing.asl and the ics-nl-processing.asl files. Use a text editor or the **sm_edit** utility to modify the values. For example:

```
BASEDIR/smarts/bin>sm_edit ../rules/ics/ics-event-processing.asl
```

You must reconfigure the domain for the changes to take effect.

Notification lifecycle auditing

Notification lifecycle auditing feature helps you identify any kind of delay in the event or notification processing lifecycle. If this feature is enabled the audit log includes additional information such as the time spent by a notification in a hook script, in addition to notify, refresh impact, NOTIF processing, change and clear events.

Auditing entries are displayed for the following stages of event or notification processing:

- Notification creation and updating of notification attributes
- Topology element linking
- Hook script invocation
- Impact and CausedBy relationship update
- NOTIF processing

The feature is disabled by default and can be manually enabled when required for that specific domain by setting the variable *detailedEventAudit* to TRUE.

To enable notification lifecycle auditing, type the following command:

```
#dmctl -s <SAM-Server> put InChargeDomain::InChargeDomain_<omain
Name>::detailedEventAudit TRUE
```

For example,

```
#dmctl -s <SAM-Server> put
   InChargeDomain::InChargeDomain_INCHARGE-AM::detailedEventAudit TRUE
```

The feature can be disabled for each domain by setting the parameter *detailedEventAudit* to FALSE using the same command:

```
#dmctl -s <SAM-Server> put
   InChargeDomain::InChargeDomain_INCHARGE-AM::detailedEventAudit
   FALSE
```

The values of the EventAudit and NotificationAudit flags in the DXA configuration files associated with the underlying domain are internally changed based on the value set for detailedEventAudit.

NOTICE

EMC recommends that you use the variable detailedEventAudit to enable or disable notification lifecycle auditing. Modifying values in the DXA configuration files can cause a conflict and there is a chance that the value of detailedEventAudit will get overwritten when the event driver restarts.

The audit trail has a size limit and is by default set to 100 entries. *Defining system defaults* section in the *EMC Smarts Service Assurance Manager Configuration Guide* describes the procedure to modify the default audit trial size limit.

Configuring weights.conf for OTM

NOTICE

This is an Optical Transport Manager (OTM) product-specific configuration in SAM, and hence not required to be carried out by other product users.

The SAM weights.conf file must be configured in the SAM installation area. After you install SAM, the following new, weights files are installed in SAM installation area:

- ‹BASEDIR›smarts/conf/ics/weights.conf
- ‹BASEDIR›smarts/conf/ics/weights-otm.conf

NOTICE

The weights-otm.conf contains the names of the classes which need to be appended to weights.conf for Impact count calculation.

1. Use sm_edit to open the ‹BASEDIR›smarts/conf/ics/weights.conf and the ‹BASEDIR›smarts/conf/ics/weights-otm.conf.
2. Copy the following configuration items from ‹BASEDIR›/smarts/conf/ics/weights-otm.conf:
   - ClassWeight OpticalNetworkElement 0
   - NotificationClassWeight HighOrder_Circuit 1
   - NotificationClassWeight LowOrder_Circuit 1
   - NotificationClassWeight ClientCircuit 1
   - NotificationClassCheck TopologicalLink
   - NotificationClassCheck DropSideTopologicalLink
   - NotificationClassCheck BBFiberLink
   - NotificationClassCheck BBTopologicalLink
   - NotificationClassCheck BBDropSideTopologicalLink
   - NotificationClassCheck Equipment
   - NotificationClassCheck Card
   - NotificationClassCheck EquipmentProtectionGroup
   - NotificationClassCheck FiberLink
   - NotificationClassCheck FiberBundle
   - NotificationClassCheck OpticalNetworkElement
   - NotificationClassCheck HighOrder_Trail
   - NotificationClassCheck LowOrder_Trail
   - NotificationClassCheck Transponder
   - NotificationClassCheck Amplifier

- NotificationClassCheck PTP
- NotificationClassCheck Conduit

3. Paste the configuration items to ‹BASEDIR›/conf/ics/weights.conf.
4. Save and close conf/ics/weights.conf.
5. Close conf/ics/weights-otm.conf without saving.

EMC Smarts Optical Transport Management Installation Guide provides detailed instructions on installing OTM.

Hierarchical SAM configuration

In a typical SAM configuration, when the server is started it does not contain the dynamic library to load all custom classes, for example, UIM_O_AdapterManager. In a custom hierarchical SAM configuration, this could result in generating ASL errors.

The following configuration must be made in the hierarchical SAM area to avoid the generation of ASL errors:

1. Copy dyn_uii.ldm and dyn_uii.mdl files from the Service Assurance Manager BASEDIR/smarts/model/ics files to the BASEDIR/smarts/local/model/top-sam folder. Here, top-sam is the custom created folder for the top-level SAM.
2. Restart the server.

CHAPTER 5
Configuring Domains

This chapter contains the following information:

- Configuring domains 88
- Configuring domain groups 93
- Configuring domain types 97
- Managing overlapping elements from separate underlying domains 101
- Configuring domain tagging 103
- Reconfiguring the server 106

Configuring domains

This section provides detailed information on configuring domains and includes the following topics:

- "Creating a domain" on page 88
- "Modifying domains" on page 91
- "Deleting domains" on page 92

Table 10 on page 88 defines the domains available.

NOTICE

The domains will be available in the Console only if SAM is started without an RPS file. For domains with existing RPS files, the domains will not be available. The INCHARGE-AM domain is enabled by default. All other domains are disabled.

Table 10 Available domains

Domain	Description
INCHARGE-AM-PM	Availability and Performance Manager
INCHARGE-AM	Availability Manager
INCHARGE-DISCMGR	Discovery Manager
INCHARGE-MPLS-ANALYSIS	Analysis Server for MPLS 2.0
INCHARGE-MPLS-MONITORING	Monitoring Server for MPLS 2.0
INCHARGE-MPLS-TOPOLOGY	Topology Server for MPLS 2.0
INCHARGE-OI	Adapter Platform
INCHARGE-PM	Performance Manager
INCHARGE-MBIM	MBIM Manager
NCM	Network Connectivity Monitor

NOTICE

You can use **sm_config** utility to import domain configurations. However, EMC Corporation recommends that you create and modify domains using the Domain Creation Wizard and the Global Manager Administration Console. "Creating a domain" on page 88 and "Modifying domains" on page 91 provide detailed information.

Creating a domain

To create a domain:

1. From the **Global Manager Administration Console**, click **Edit** › **New Domain**.

   The **Domain Creation Wizard** appears.

2. Type the name of the domain server in the **Domain Name** field.

3. Type the description of the domain in the **Domain Description** field, and click **Next**.

   NOTICE

   To create a domain using the Copy Existing option, select an existing domain server from the **Copy Existing** list, type a **Domain Name**, and click **Finish**.

4. Click the domain type. Available options include:

   - **Use Default Type**

     NOTICE

     This option is available only when copying an existing domain.

   - **Create New Type**

     If this option is selected, type the name of the domain type in the **Type Name** field.

     NOTICE

     To copy an existing domain type, select an existing domain type from the **Copy Existing** list.

   - **Select Existing Type**

     If this option is selected, select a domain type from the **Select Type** list.

5. Click **Next**.
6. Type the tag string in the **Tag String** field.

   NOTICE

   If a tag string is not specified, go to step 12.

7. Click **Next**.
8. Click the appropriate option. Available options include:

   - **Use Default Tag**

     NOTICE

     This option only appears when a copied domain is selected or when the selected domain has an assigned tag.

   - **Create New Tag**

     NOTICE

     Type a name in the **Tag Type Name** field. To copy an existing tag, select a tag from the **Copy Existing** list.

- **Select Existing Tag**

NOTICE

Select a tag from the **Select Tag** list.

9. Click **Next**.
10. Type description in the **Tag Description** field.
11. Type a matching tag pattern in the **Matching Pattern** field.
12. Click **Next**.
13. Select the groups this domain belongs to from the **Available Groups** list.
14. Click the group's role. Available options include:
    - **Topology Server**
    - **Monitoring Server**
    - **Analysis Server**

    NOTICE

    This step is only required if configured groups exist.

15. Click **Add** to move the group to the **Selected Groups** list.
16. Click **Next**.

    The instance that will be created when the domain creation process is complete is displayed.

    NOTICE

    By default, the domain is enabled. If the domain is part of a domain group, it is only enabled if the domain group it belongs to is enabled. To disable the domain, clear its checkbox.

17. Click **Finish**.

Domain naming conventions

To allow the console to easily suppress names of component domain servers, domain names must comply with the following naming convention:

Group Name-Role[-Sub-Role]

Where:

- *Group Name* — Identifies the function of the group (for example, MPLS) as well as the region or other further designation if multiple groups with the same function are being deployed. For example, MPLS-NA might designate the MPLS server group monitoring North America.

- *Role* — Must be one of MONITORING, TOPOLOGY, or ANALYSIS. This is the key field the graphical user interface uses to identify component servers.
- *Sub-Role* — Distinguishes multiple servers within the same role. For example, MPLS-Monitoring-Snmp and MPLS-Monitoring-Ping.

Modifying domains

To modify a domain:

1. Open the **Global Manager Administration Console**.
2. From the **Manager List**, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domains**.
6. Click the appropriate domain.

   The **Configure Domain** interface appears, as illustrated in Figure 6 on page 91.

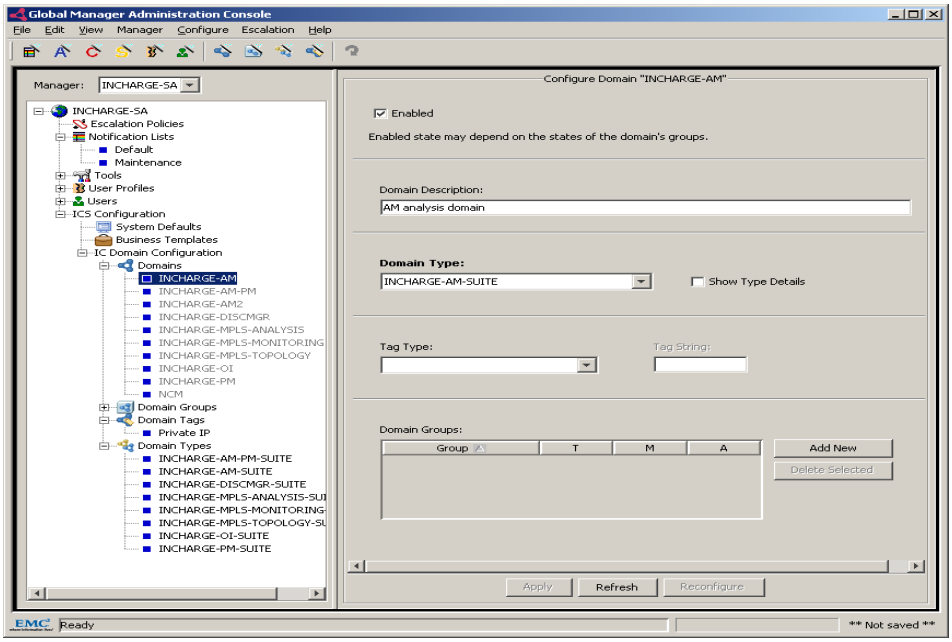

Figure 6 Global Manager Administration Console—Configure Domain interface

7. Type the appropriate configurations.

Table 11 on page 91 describes the available Domain parameters.

Table 11 Domain parameters

Parameter	Description
Enable	Enables the domain.
Domain Description	Provides a description of the domain.
Domain Type	Defines the domain type.

Table 11 Domain parameters (continued)

Parameter	Description
Show Type Details	Displays the domain type information.
Tag Type	Defines the tag type.
Tag String	Defines the tag string.
Domain Groups	Defines the domain groups in which the domain is included. The table includes the following columns: • T — Topology • M — Monitoring • A — Analysis **Notice:** Click **Add New** to display the **Add Domain Group** dialog box. Select a **Domain Group** and click **Delete Selected** to remove a domain group.

8. Click the **Apply**.
9. Click **Reconfigure**.

NOTICE

Configuration changes are not applied until the **Reconfigure** button has been clicked. Clicking the **Reconfigure** button starts the data synchronization process. Click the **Reconfigure** button only after all updates to domains, domain tags, domain groups, and domain types have been made.

Deleting domains

To delete domains:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain**.
6. Click the appropriate domain.
7. Select **Edit** › **Delete**.

   The selected domain is deleted.

NOTICE

Domains that refer to other domains cannot be deleted.

Configuring domain groups

A typical domain group comprises three servers—a topology server, an analysis server, and a monitoring server.

NOTICE

For more complex deployments, contact Custom Engineering.

The Configure Domain Group interface consists of a description of the domain group and three sections listing the topology servers, monitoring servers, and analysis servers. A typical MPLS deployment includes one of each type of server. The pane included on the left of the interface displays the default configuration of the Service Assurance Manager (SAM) server.

NOTICE

Most domain objects included in the default configuration are disabled (except INCHARGE-AM). The MPLS Group is disabled by default.

This section includes the following topics:

- "Creating domain groups" on page 93
- "Modifying domain groups" on page 94
- "Deleting domain groups" on page 96

Creating domain groups

Use the Domain Groups Creation Wizard to create a new domain group. Using the wizard, you can:

- Copy an existing domain group and type a name for the new group. The wizard automatically creates the domains in the group.
- Manually create the component domains from within the wizard.

Creating a domain group using the Copy Existing option

To create a domain group using the Copy Existing option:

1. From the **Global Manager Administration Console**, click **Edit** › **New Domain Group**.

   The **Domain Group Creation Wizard** appears.
2. Select a group to be copied from the **Copy Existing** list.
3. Type a name in the **Group Name** field.
4. Type a description in the **Group Description** field.
5. Click **Next**.

6. Click **Default Configuration**.

   NOTICE

   To customize the configuration, click **Customized Configuration**. This option is for advanced users only.

7. Click **Finish**.

   The configuration objects are created in the server.

Manually creating a domain group

To manually create a domain group:

1. From the **Global Manager Administration Console**, click **Edit** › **New Domain Group**.

   The **Domain Groups Creation Wizard** appears.

2. Type a name in the **Group Name** field.
3. Type a description in the **Group Description** field.
4. Click **Next**.
5. Click **Default Configuration**.

   NOTICE

   To customize the configuration, click **Customized Configuration** and complete the Select Topology Server options. If the Default Server option is selected, the wizard creates a default server for this group. The default server does not include any domain tagging information. The default server's domain type is copied from the corresponding domains of the copied domain group.

6. Click **Next**.

   The instance that will be created once the domain group creation process is complete appears in the wizard.

   NOTICE

   By default, the domain group is enabled. If a domain is part of the domain group, it is also enabled. To disable the domain group, clear its checkbox.

7. Click **Finish**.

Modifying domain groups

To modify a domain group:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain Groups**.

6. Click **INCHARGE-MPS**. The **Configure Domain Group** interface appears, as illustrated in Figure 7 on page 95.

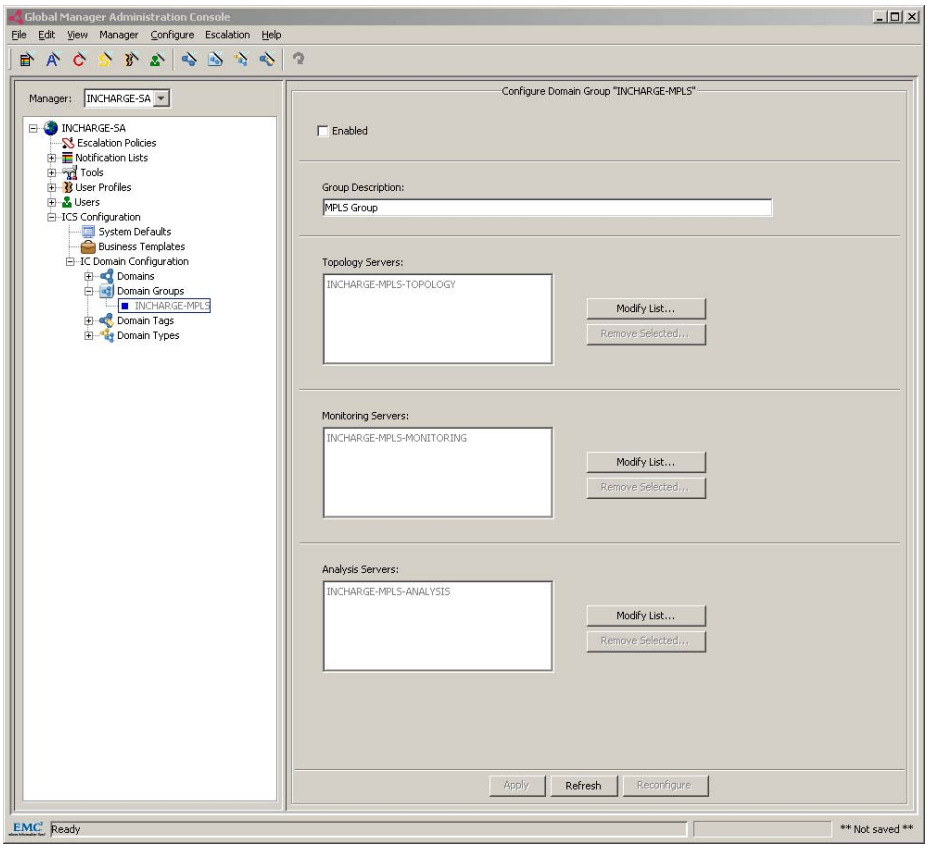

Figure 7 Global Manager Administration Console—Configure Domain Group interface

7. Type the appropriate configurations.

   Table 12 on page 95 describes the available Domain Group parameters.

Table 12 **Domain Group parameters**

Parameter	Description
Enable	Enables the domain group. Enabling a domain group enables all the domains within it. Disabling the group disables domains included in the group as long as the domains are not members of any other enabled group. Disabling or enabling a domain, however, has no effect on groups of which that domain is a member.
Group Description	Provides a description of the domain group.
Topology Servers	Defines the topology servers.
Monitoring Servers	Defines the monitoring servers.
Analysis Servers	Defines the analysis servers.

NOTICE

Click the **Modify List** button to display the **Modify List of Topology Services** dialog box. To remove a server from a list, select the server in the list and click the **Removed Selected** button.

8. Click the **Apply**.
9. Click **Reconfigure**.

NOTICE

Configuration changes are not applied until the **Reconfigure** button has been clicked. Clicking the **Reconfigure** button starts the data synchronization process. Click the **Reconfigure** button only after all updates to domains, domain tags, domain groups, and domain types have been made.

Deleting domain groups

To delete domain groups:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain Groups**.
6. Click the appropriate domain group.
7. Select **Edit** › **Delete**.

   The selected domain group is deleted.

NOTICE

Domain groups that refer to other domains cannot be deleted.

Configuring domain types

A domain type is a configuration object that captures all the settings relevant to a particular type of domain with which the Service Assurance Manager server connects. One domain type exists for each class of server communicating with the Service Assurance Manager; for example, AM, NPM, MPLS, and SIA.

Properties of a domain type include:

- DXA file
- Hook scripts
- Minimum certainty
- Smoothing intervals

NOTICE

The domain type does not instantiate a domain object.

Table 13 on page 97 defines the default domain types.

Table 13 Default domain types

Domain type	Description
INCHARGE-AM-PM-SUITE	Availability and Performance Manager
INCHARGE-AM-SUITE	Availability Manager
INCHARGE-DISCMGR-SUITE	Discovery Manager
INCHARGE-MPLS-ANALYSIS-SUITE	Analysis Server for MPLS 2.0
INCHARGE-MPLS-MONITORING-SUITE	Monitoring Server for MPLS 2.0
INCHARGE-MPLS-TOPOLOGY-SUITE	Topology Server for MPLS 2.0
INCHARGE-OI-SUITE	Open Integration Server
INCHARGE-PM-SUITE	Performance Manager
INCHARGE-SAM-MBIM-SUITE	Maintenance and Business Impact Manager
NCM-SUITE	NCM OEM

NOTICE

Additional domain types previously configured in the ics.conf file are available as separately loadable XML files located in the conf/ics directory.

The following topics are included in this section:

- "Creating domain types" on page 98
- "Modifying domain types" on page 99
- "Deleting domain types" on page 101

Creating domain types

To create a domain type using the Domain Groups Creation Wizard:

1. From the **Global Manager Administration Console**, click **Edit** › **New Domain Type**.

   The **Domain Groups Creation Wizard** appears.

2. Select a type to be copied from the **Copy Existing** list.
3. Type a name in the **Type Name** field.
4. Type a description in the **Type Description** field.
5. Click **Next**.
6. Edit the domain type settings, and click **Next**.

   Table 14 on page 98 defines the available domain type settings.

Table 14 Domain type settings

Option	Definition
DXA File	Defines the DXA configuration file.
Hook Script: Enable	Enables the hook script. **Notice:** If the hook script is enabled, type the path for the hook script in the field.
Minimum Certainty	Defines the minimum certainty. **Default value**: 0.01
Smoothing Interval	Defines the smoothing interval. **Default value**: 65

7. Select the servers from the **Available Domains** list.
8. Click **Add** to move the server to the **Selected Domains** list.

   This defines the servers associated with the new domain type.

NOTICE

Domains cannot be removed from a domain type—they can only be reassigned to another domain type. By adding domains, you are only reassigning the domains to a different domain type. Newly added domains display in green text. You can only delete domains displayed in green text. After clicking **Apply**, the domains cannot be deleted.

9. Click **Next**.
10. Click **Finish**. The configuration objects are created in the server.

Modifying domain types

To modify a domain type:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain Types**.
6. Click the appropriate **INCHARGE Domain Type**. The **Configure Domain Type** interface appears, as illustrated in Figure 8 on page 99.

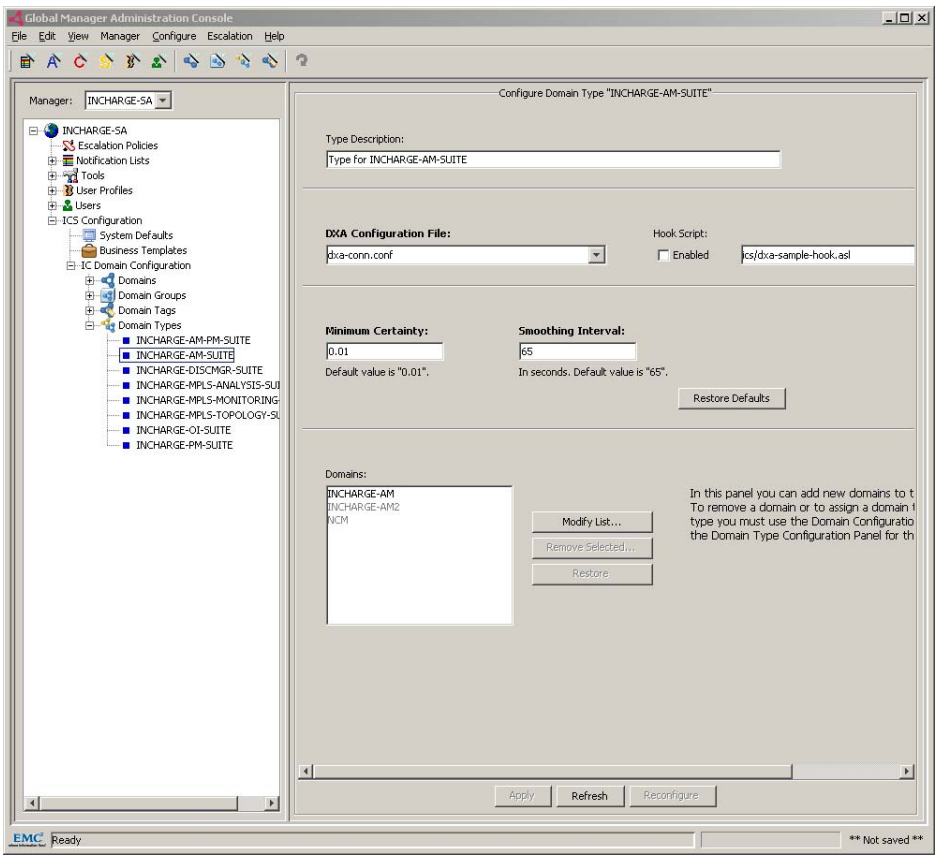

Figure 8 Global Manager Administration Console—Configure Domain Type interface

7. Type the appropriate configurations.

Table 15 on page 100 defines the available domain type configuration parameters.

Table 15 Domain type configuration parameters

Parameters	Definition
Type Description	Defines the domain type.
DXA Configuration File	Defines the DXA configuration file.
Hook Script: Enable	Enables the hook script. **Notice:** If the hook script is enabled, type the path for the hook script in the field.
Minimum Certainty	Defines the minimum certainty. **Default value**: 0.01
Smoothing Interval	Defines the smoothing interval. **Default value**: 65
Domains	Defines the domains associated with the domain type. **Notice:** Click **Modify List** to add domains, select a domain from the list and click **Remove Selected** to remove domains, and click **Restore** to restore previously configured domains. Domains cannot be removed from a domain type, they can only be reassigned to another domain type.

8. Click **Apply**.
9. Click **Reconfigure**.

NOTICE

Configuration changes are not applied until the **Reconfigure** button has been clicked. Clicking the **Reconfigure** button starts the data synchronization process. Click the **Reconfigure** button only after all updates to domains, domain tags, domain groups, and domain types have been made.

Deleting domain types

To delete domain tags:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain Types**.
6. Click the appropriate domain type.
7. Select **Edit** › **Delete**.

   The selected domain type is deleted.

NOTICE

Domain types that refer to other domains cannot be deleted.

Managing overlapping elements from separate underlying domains

A Global Manager collects topology information from multiple underlying domains. In some cases, two or more of these domains can manage elements with the same name. Two such examples are IP networks, which are named using the network address, and partitions which are assigned names by the underlying domain.

For example, a service provider might use a private IP network address, such as 10.0.0.0, to provide IP addresses to different customers. When the same range of IP addresses are assigned to multiple customers, different topology elements may have the same IP address, and thus the same name.

Partitions are created and named by IP Availability Manager. All IP Availability Managers use the same convention to name partitions. If the topologies of two domain managers each include partitions, it is possible that one or more of the partitions have the same name, as defined by the Name attribute. The *EMC Smarts IP Manager User Guide* provides more information about partitions.

By default, the Global Manager treats elements with the same name from different underlying domains as a single instance, consolidating the relationships and attributes of the two elements to a single topological instance. The values for the attributes are taken from the underlying domain that performs the most recent topology synchronization.

You can, however, configure the Global Manager to manage elements of the same name from different domains as distinct elements. By specifying a tag for one or both of the underlying domains, the Global Manager will create two separate topology elements, each containing the attributes and relationships of the respective objects in the underlying domains. The tag that you specify is appended to the name of the objects in the tagged domain and displayed in the Global Console.

To tag managed elements, you need to complete two tasks:

- Specify a tag.
- Specify a matching pattern that will match the IP addresses and partitions to be tagged.

How the Global Manager applies tags

The Global Manager can apply tags to both topology elements and notifications. The Global Manager finds matching elements by comparing the matching pattern against the Name and DisplayName attributes of both topology elements and notifications from the relevant underlying domain. If they match the filter, both are applied to the tag. Additionally, a notification on a matching instance will cause the Name, DisplayName, InstanceDisplayName, and InstanceName of the notification to be tagged.

As a result, you should construct a matching pattern to match the values of attributes you want tagged. For topology elements, this can include the Name and DisplayName attributes. For notifications, this can include the DisplayName, InstanceDisplayName, and InstanceName attributes.

For example, the pattern IP*-172.16.*|172.16.* could match the following:

- Elements of the IPNetwork class, which are prefixed with “IPNET-”. For IPNET-172.16.0.0, the following would result:
  - DisplayName of 172.16.0.0 [tag-3]
  - Name of IPNET-172.16.0.0_tag-3.
- Elements of the system classes, such as Host, that are named using the system’s IP address. For host 172.16.1.107, the following would result:
  - DisplayName of 172.16.1.107 [tag-3]
  - Name of 172.16.1.107_tag-3.
- Notifications generated for these elements would also be tagged. For a Host Down notification, the following would result:
  - DisplayName of Host Down 100%: 172.16.1.107 [tag-3]
  - InstanceDisplayName of 172.16.1.107 [tag-3]
  - InstanceName of 172.16.1.107_tag-3.

NOTICE

If the system is named by its hostname, this matching pattern would not apply.

Configuring domain tagging

Domain tags allow the Service Assurance Manager to tag a subset of the topology of a given domain. Domain tags distinguish a topology subset from topology from an underlying domain which may have instance names that overlap but represent unique instances in the network topology.

A domain tag consists of a unique identifier and a wildcard expression that identifies matching instances.

The following topics are included in this section:

- "Creating domain tags" on page 103
- "Modifying domain tags" on page 104
- "Deleting domain tags" on page 105

Creating domain tags

To create a domain tag:

1. From the **Global Manager Administration Console**, click **Edit** › **New Domain Tag**.

   The **Domain Tag Creation Wizard** appears.

2. Type a name in the **Tag Type Name** field.
3. Type a description in the **Tag Description** field.

   NOTICE

   To copy an existing tag, select a tag from the **Copy Existing** list.

4. Click **Next**.
5. Type a matching pattern for the domain tag in the **Matching Pattern** text box.
6. Click **Next**.
7. From the **Available Domains** list, select domains for which the domain tag applies.

   NOTICE

   For each domain selected, type a tag string in the **Tag String** field. If a domain has a tag string already associated with it, you can replace the domain tag associated with the tag string with this new one.

8. Click the **Add** button to move the selected domains to the **Selected Domains** list.
9. Click **Next**.

   Instances that will be created when the create domain tag process is completed appear in this screen.

10. Click **Finish**.

Modifying domain tags

To modify domain tags:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain Tags**.
6. Click the appropriate domain tag.

   The **Configure Domain Tag** interface appears, as illustrated by Figure 9 on page 104.

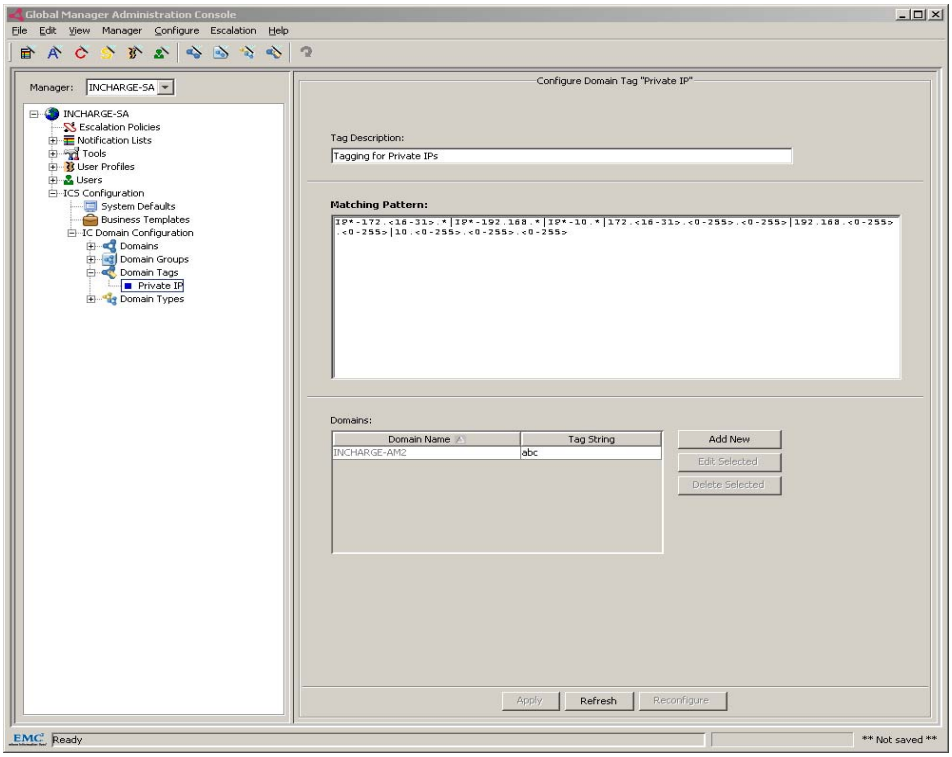

Figure 9 Global Manager Administration Console—Configure Domain Tag interface

7. Type the appropriate configurations.

Table 16 on page 105 describes the available Domain Tag parameters.

Table 16 Domain Tag parameters

Parameter	Description
Tag Description	Provides a description of the domain tag.
Matching Pattern	Defines the domain tag matching pattern. **Notice:** Wildcard expressions are valid.
Domains	Defines the domains with which the domain tag is associated. **Notice:** Click **Add New** to display the **Add New Domain** dialog box. Select a domain and click **Edit Selected** to display the **Edit Domain** dialog box. Select a domain and click **Delete Selected** to delete the selected domain.

8. Click **Apply**.
9. Click **Reconfigure**.

NOTICE

Configuration changes are not applied until the **Reconfigure** button has been clicked. Clicking the **Reconfigure** button starts the data synchronization process. Click the **Reconfigure** button only after all updates to domains, domain tags, domain groups, and domain types have been made.

Deleting domain tags

To delete domain tags:

1. Open the **Global Manager Administration Console**.
2. From the **Manager** list, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domain Tags**.
6. Click the appropriate domain tag.
7. Select **Edit** › **Delete**.

   The selected domain tag is deleted.

NOTICE

Domain tags that refer to other domains cannot be deleted.

Reconfiguring the server

Any modifications to domains, domain groups, domain tags, and domain types are not implemented until the server has been reconfigured. You should reconfigure the server only after all modifications have been applied—not after each individual modification. When you click the Launch Server Reconfiguration toolbar button, the reconfigure process begins.

Click **View** › **Message Log** to display the Manager Message Log window. Configuration errors are displayed in this window.

NOTICE

The Service Assurance Manager server applies all reconfigurations during the restart process.

To reconfigure the server:

From the Global Manager Administration Console, click the **Launch Server Reconfiguration** toolbar button.

NOTICE

Click the **Launch Server Reconfiguration** toolbar button only after all modifications have been applied.

CHAPTER 6
Managing Notifications with the Global Manager

This chapter contains the following information:

- Overview of notifications 108
- Understanding and managing notifications 109
- Acknowledging and archiving notifications 113
- Notification types and incrementing OccurrenceCount 114
- User-defined notification attributes 115

Overview of notifications

The Global Manager stores the topology and event information that it receives from the underlying domains in its repository. The topology and event information are stored as objects, that is as instances of the classes defined in the ICIM model. The notifications that Global Manager sends to clients are themselves objects in the Global Manager's repository.

Notification objects, similar to other objects in the Global Manager's repository, have attributes that describe their properties. For notifications, these attributes include the time the event occurred, the type of event, the name of the object where the event occurred, and much more information. "Attributes for matching notification properties" on page 215 provides a complete list of notification attributes.

Notification attributes are used throughout Service Assurance for a variety of purposes:

- Notification attributes correspond to the columns in the Notification Log of the Global Console.

  NOTICE

  Attribute names are not exactly the same as the column headings of the Notification Log.

- You can specify a matching pattern against the values of notification attributes for filtering a notification list.
- Twenty notification attributes are user-defined. You can write an ASL script that populates these attributes with additional information.

NOTICE

This chapter discusses how you can manage notifications using the Global Manager. In addition to the standard notification processing provided by the Global Console, you can also manage notifications creation in a more extensive manner from any event source using the NOTIF Editor. This editor allows for real-time reconfiguration of event and notification processing without server restarts. The *EMC Smarts Notification Module User Guide* provides detailed information on configuring events and notifications using the NOTIF Editor.

Understanding and managing notifications

This section describes the states of a notification. It also describes the attributes of a notification and how a change in the notification's state affects the values of these attributes. Finally, this section describes the acknowledgement and archiving of notifications.

The following notification attributes are related to a notification's state:

- Event State
- First Notify Time
- Last Notify Time
- Last Change Time
- Last Clear Time
- Count

NOTICE

The connotation for the Last Notify Time attribute differs between the one displayed in the SAM notification console and the IP event console. In the IP event console, Last Notify Time indicates the current timestamp of the console, compared to the actual notification time displayed in SAM. Hence, the Last Notify Time from the IP event console must not be used for correlation of the Last Notify Time between SAM and IP.

States of a notification

A notification's state is defined by the Event State attribute, which has five possible values. Table 17 on page 110 describes each possible value of the Event State attribute and shows its relationship to the Active notification attribute.

Table 17 Description of Event State notification attribute

Event State value	Description	Active value
ACTIVE	Event that caused the notification is occurring. The notification is in the "notify" state.	TRUE
WAS_ACTIVE	Global Manager is disconnected from the underlying domain that was the source for the notification. **Notice:** The WAS_ACTIVE attribute is not applicable when the Broadcast Event State on Restart Flag is configured to FALSE. "The WAS_ACTIVE and SUSPENDED event states" on page 110 describes the suspended state. "Defining system defaults" on page 69 provides detailed information on the Broadcast Event State on Restart Flag system default.	TRUE
SUSPENDED	Global Manager is no longer receiving information about an active notification. **Notice:** The SUSPENDED attribute is not applicable when the Broadcast Event State on Restart Flag system default is configured to FALSE. "The WAS_ACTIVE and SUSPENDED event states" on page 110 describes the suspended state. "Defining system defaults" on page 69 provides detailed information on the Broadcast Event State on Restart Flag system default.	TRUE
INACTIVE	Event that causes the notification is no longer occurring. The notification is in the clear state.	FALSE
UNINITIALIZED	State of the notification object when it is first created and before any state is assigned to it.	FALSE

The WAS_ACTIVE and SUSPENDED event states

The WAS_ACTIVE event state is used to identify those notifications that are active under two conditions:

- A Global Manager disconnects from the underlying domain that is the event source. In this case, all active notifications from the disconnected domain are marked WAS_ACTIVE.
- A Global Manager is started from a saved repository file. In this case, all active notifications are marked WAS_ACTIVE.

Notifications marked as WAS_ACTIVE remain in the WAS_ACTIVE state until the Global Manager can verify their status.

If the Global Manager is not able to reconnect to the underlying domains that generated these notifications in 1800 seconds (30 minutes), the value of the Severity attribute is set to 4 ("Severity" on page 217 provides additional information), changing the color of the notification in a Notification Log to blue. The value of 1800 seconds is referred to as the detachTime.

When the connection between the Global Manager and the underlying domains is reestablished, the Global Manager does the following:

- Notifications that are still active in the underlying domain have their Event State changed to ACTIVE and their Severity value updated accordingly. The Last Change attribute is assigned the current time and the Last Notify attribute value remains unchanged.
- Notifications that have cleared in the underlying domain remain in the WAS_ACTIVE state until the attachTime has elapsed. The default value of attachTime is 6000 seconds (100 minutes). However, the actual value that is used to calculate when to clear the notifications is determined as follows:
  - When the uptime for the underlying domain is greater than the attachTime, then the value is 240 seconds plus the smoothing interval. The smoothing interval is specified in the corresponding Domain Type. This typically occurs when the server was restarted.
  - When the uptime for the underlying servers is less than the attachTime, the value is attachTime minus uptime plus the smoothing interval plus 20 minutes. For temporary disconnects, this value is typically used to determine the state of WAS_ACTIVE notifications. This typically occurs when the server was restarted and there was a disconnect.

The SUSPENDED state indicates that the Global Manager is no longer able to retrieve information about an active notification. When an underlying Domain Manager is not able to get to the source of a notification, it suspends the notification. The Domain Manager sends this message to the Global Manager, which in turn suspends the notification. A Domain Manager may not be able to get to the source of a notification because an SNMP agent is not responding or because it received unexpected error values in an SNMP request.

Notifications suspended by the Global Manager have the value of the Severity attribute set to 4, changing the color of the notification in a Notification Log to blue.

A notification's life cycle

Each notification has a unique name. A notification's name consists of the class name and instance name where the event occurred, and the event name. The notification naming convention is:

```
NOTIFICATION-ClassName_InstanceName_EventName
```

For example, the notification NOTIFICATION-Router_R1_Down identifies the event Down, which occurred in instance R1 of the Router class.

The Notification Log Console will not list more than one notification with this name. If the event clears and occurs again before it is archived, the Count field is increased accordingly.

The following notification attributes are related to a notification's state:

- First Notify — Is the first time a notification occurs.
- Last Notify — Is the last time a notification occurs.
- Last Clear — Is the last time a notification cleared.
- Last Change — Is the last time a notification changed.
- Count — Is the number of occurrences of an event starting from First Notify until Last Notify.

Table 18 on page 112 illustrates the states of the notification NOTIFICATION-Router_R1_Down. For the First Notify and Last Notify attributes, the timestamp value is provided by the underlying domain that diagnosed the event. Values for Last Change, Count, and Last Clear (not shown) are provided by the Global Manager.

Table 18 States of NOTIFICATION-Router_R1_Down

Time	User/system actions	Notification state	Time of first notify	Time of last notify	Time of last change	Count
2:00		Notify	1:00	1:00	2:00	1
2:15		Clear	1:00	1:00	2:15	1
2:25		Notify	1:00	2:25	2:25	2
2:35		Clear	1:00	2:25	2:35	2
3:00	Acknowledge		1:00	2:25	3:00	2
3:05	Archived		1:00	2:25	3:00	2
3:55		Notify	3:55	3:55	3:55	1

In this example:

At 1:00, an Availability Manager generated a Down event.

At 2:00, the Global Manager connected to the Availability Manager and the Global Manager created the notification NOTIFICATION-Router_R1_Down for the Down event. The Global Manager retains the original timestamp of the event and stores it in the notification—First Notify and Last Notify are assigned the values of 1:00.

At 2:15, the notification is cleared.

At 2:25, the Availability Manager generated the same Down event again, the Availability Manager was already connected to the Global Manager, and the Global Manager reactivates the notification. The First Notify remains 1:00 and the Last Notify reads 2:25.

At 2:35, the notification is cleared.

At 3:00, the operator acknowledged the notification.

Notice at 3:05, the notification is archived and the archive action records the most recent values for First Notify, Last Notify, and Last Change. A notification cannot be archived until it has been acknowledged. Acknowledgement does not change a notification's state unless the notification attribute ClearOnAcknowledge is set to True for a specific notification.

At 3:55, the Availability Manager generated a Down event again and the Global Manager created the notification again. After a notification is archived, the Global Manager treats a recurrence of that notification as though it were a first occurrence. The value of notification attributes start all over again: new First Notify time, Count starts at 1, and so on.

Acknowledging and archiving notifications

Regardless of its state, a notification can be marked as Acknowledged. An operator can acknowledge a notification through the Global Console. When an operator acknowledges a notification, the operator becomes the owner of the notification. Acknowledging a notification does not change its state. If an inactive acknowledged notification is re-notified, the value of the Acknowledged attribute is set to FALSE and the value of the Owner attribute is cleared.

By default, the Global Manager automatically acknowledges cleared (inactive) and unowned notifications after 5 minutes. When Global Manager acknowledges a cleared notification, the owner is set to SYSTEM. You can configure this interval through the AutoAcknowledgementInterval.

Using a notification list, you can filter acknowledged notifications so that they do not appear in an operator's display.

After a notification has been acknowledged and is cleared (inactive), it is eligible to be archived. An archived notification is removed from the Global Manager's repository and written to an archive file. The notification archive file is named *‹global_manager›*.archive and located in the *BASEDIR*/smarts/local/logs directory.

When a notification is archived, the Global Manager treats a recurrence of that notification as though it were the first occurrence. The value of notification attributes start over again: new First Notify time, Count starts at 1, and so on.

The Global Manager archives only notifications that are both inactive and acknowledged. You can configure the acknowledgement and archival of notifications through the AutoAcknowledgementInterval and InactiveAutoArchiveInterval settings.

Configuration of acknowledgment and archiving of notifications

The following parameters, defined using the Global Administration Console, control how and when the Global Manager acknowledges or archives notifications. "Defining Domain Manager parameters" on page 66 describes these parameters.

NOTICE

Any changes made to the parameters of the ics-default.xml file require that you click the Reconfigure button to make those changes take effect. "Reconfiguring the server" on page 106 provides detailed information.

AutoAcknowledgementInterval

AutoAcknowledgementInterval controls when a cleared (inactive) notification is automatically marked as acknowledged by the Global Manager. The interval is calculated from the time when a notification clears. If a notification recurs, or is unacknowledged, the notification is rescheduled for automatic acknowledgement. The default value is 300 seconds (5 minutes). When this interval is set to 0, the Global Manager will not automatically acknowledge cleared notifications.

InactiveAutoArchiveInterval

InactiveAutoArchiveInterval controls when a cleared (inactive) and acknowledged notification is archived by the Global Manager. The interval is calculated from either the time the notification is cleared or the time a notification is acknowledged (whichever is later). If a notification recurs, or is unacknowledged, the notification is rescheduled for archival. The default value is 14400 seconds (4 hours). When this interval is set to 0, the Global Manager will not automatically archive cleared and acknowledged notifications.

Retain local objects with an associated notification instance

If attributes such as, elementname and elementclassname are found empty for notifications in the Notification Log Console, it is likely that corresponding local objects were deleted during topology synchronization with underlying domain managers but notifications were not cleared.

To avoid local objects that have an associated notification instance from being deleted, set the following property in the *local/conf/runcmd_env.sh file*:

```
SM_KEEP_LOCAL_OBJECTS=true
```

By default, it is set to **FALSE**.

Notification types and incrementing OccurrenceCount

The ICIM defines two types of notifications: momentary and durable. A momentary notification has no duration, it describes an event that happened at a specific time. An authentication failure is an example of a momentary event. A durable notification describes an event that is active over a period of time. While the event is active, the problem it causes is still in effect. An example of a durable event is a link failure. You can determine a notification's type by checking the value of the EventType notification attribute.

A notification's type determines how the value of the OccurrenceCount attribute is increased by a Global Manager. In addition, the source of the notification and the conditions under which a notification is sent, also determine how a notification's count is increased.

For example, consider the scenario where notification N_1 is active in a Global Manager and there are two different sources (S1 and S2) for the notification:

- For durable notifications, the OccurrenceCount represents how many times the notification has become active (notified) during the notification's life cycle. The count is incremented when the notification reoccurs after it was cleared and renotified.

  Using the example above, when S1 notifies, the Global Manager increments the OccurrenceCount to 1. If S2 then notifies, the OccurrenceCount remains 1.

- For momentary notifications, the OccurrenceCount represents how many times the Global Manager has received a notify message regarding the notification. The current state of the notification has no effect.

  Using the example above, when S1 notifies, the Global Manager increments the OccurrenceCount to 1. If S2 then notifies, the OccurrenceCount increments to 2.

  The Global Manager would increment the OccurrenceCount on a re-notify by 1, regardless of whether the notification was active. In fact, if notification N_1 occurred 50 times in the underlying domain, the Global Manager would increase the value of the OccurrenceCount by 50.

If the Event Type is durable or the value for the SM_OCCURANCECOUNT flag is set to FALSE in the runcmd_env.sh file while the event type is momentary, the server always ignores any re-occurrences of an active notification that has occurred during the disconnected period. If the Event Type is momentary and the value for the SM_OCCURANCECOUNT flag is set to TRUE in the runcmd_env.sh file, the server will update the re-occurrences of an active notification that has occurred during the disconnected period.

User-defined notification attributes

Notification objects created by the Global Manager include twenty user-definable attributes. By default, these attributes have no value.

To populate the attributes with values, you may create an ASL program and specify it in the HookScript field of a DomainType subsection. "Defining Domain Manager parameters" on page 66 describes the HookScript field.

A sample ASL program, dxa-sample-hook.asl, is included in the *BASEDIR*/smarts/rules/ics directory. This ASL program receives a handle to the notification object. You need to provide any additional functionality. Your ASL script should query the Global Manager, Adapter Platform, or underlying domain for additional information.

For example, you can populate one of the columns with the number of affected customers (instances of the ServiceSubscriber class) by counting the number of impacted subscribers.

"Notification list parameters" on page 53 describes the method to give the column names in the Global Console more meaningful names.

User-defined notification attributes from Adapter Platform or Global Manager

When a Global Manager receives notifications from Adapter Platform or another Global Manager, these notifications include 20 user-definable notification attributes. The values for these user-defined notification attributes should be populated in the underlying Adapter Platform or Global Manager.

The user-defined attributes, and their values, are automatically passed to a Global Manager when the default data exchange file is specified in the corresponding Domain Type, for example:

- dxa-oi.conf for notifications received from Adapter Platform
- dxa-sam.conf for notifications received from Global Manager

"Defining Domain Manager parameters" on page 66 describes how to configure a Global Manager to receive events and topology from underlying domains.

CHAPTER 7
Managing Topology with the Global Manager

This chapter contains the following information:

- Topology synchronization 118
- Topology Timeout 123
- Ensuring a consistent representation of topology 123
- Organizing topology with groups 126
- Purging of HierarchicalGroupManager instance 135
- Custom icons for map nodes 136

The *EMC Smarts Business Impact Manager User Guide* describes how to augment the topology with business and service-related elements.

Topology synchronization

A Global Manager imports topology information from the underlying domains specified for all enabled domains. To maintain an up-to-date representation topology, the Global Manager synchronizes its topology with an underlying domain when any one of the following occurs:

- The underlying domain is reconnected after a connection loss or a restart.
- The Global Manager is started.
- The underlying domain performs a full or incremental discovery, rediscovers an object, or a manual discovery is initiated.
- A change is made to the configuration of the underlying domains. When the Global Manager is reconfigured, it automatically synchronizes its topology with the updated underlying domains.

Topology synchronization consists of the following two parts:

- Import of raw topology data from remote domains – This involves retrieving of raw data from the domain and creating topology elements and relationships. If topology and relationships already exist, no action is taken.
- Post processing – This involves reconciliation of the SAM topology. SAM maintains an internal data structure to keep track of topology elements imported from underlying domains which allows SAM to identify and purge topology objects and relationships which no longer exists in the underlying domains.

Starting topology synchronization

To manually start topology synchronization, invoke the following dmctl command from the *BASEDIR*/smarts/bin directory:

```
▼dmctl -s SAMServername invoke GA_Driver::<domain-name>_Topo-Driver
   start▲
```

NOTICE

▼▲ The command must be typed as one line.

Limitation of topology synchronization

The following are the limitations of SAM topology synchronization:

- Redundant syncs
- Costly operation

Redundant syncs

Topology synchronization is triggered whenever there is a change in topology of an underlying domain. But SAM may import only a subset of the topology from the underlying domain based on dxa configuration. As a result, topology synchronization may be triggered even though the imported topology has not changed in the underlying domain. Such topology synchronizations are not useful as no change is propagated to SAM and can be avoided by disabling autosync mode. Instead topology synchronizations can be triggered manually at regular intervals. One of the easiest ways to identify redundant syncs is to find out number of instances added or deleted during topology synchronizations. If it is zero, then it is likely that redundant syncs are triggered between SAM and underlying domain.

Costly Operation

Topology synchronization is a costly operation on SAM server and it is recommended to configure topology synchronization judiciously. Performance of SAM topology synchronization is dependent on the factors mentioned in Table 19 on page 119.

Table 19 Performance factors of SAM topology synchronization

Performance factor	Dependency of SAM topology synchronization
Size of topology	Time taken to import topology is directly proportional to size of topology.
Frequency of sync	More frequent syncs increase load on SAM server.
Resources(CPU/memory)	Concurrent topology synchronization works better on a multi core machine.
Network latency	Topology import slows down due to network latency.

Memory growth due to Topology Synchronization

When reconciliation of SAM topology and import of topology from other underlying domains are in progress at the same time, a growth in memory is observed.

During topology synchronisation in SAM, a separate thread is created for synchronization of each domain. Whenever topology changes in the underlying domains, topology synchronization is triggered by SAM to keep local SAM topology in synchronization with the topology in the underlying domains. The underlying domain updates the lastProbeFinishedAt attribute of ICF_TopologyManager whenever there is a change in topology. SAM subscribes to the attribute and when the attribute is updated after a topology change (like discovery in AM), SAM triggers topology synchronization with underlying domain. Each time topology synchronization is triggered, classes mentioned in dxa configuration are imported from underlying domains and SAM topology is reconciled. In other words, topology synchronization is always full and not incremental in nature.

Topology Synchronization can be run in the following three modes:

- Concurrent Synchronization – This is the default mode.
- Serial Synchronization – This can be enabled by setting SM_ENABLE_SERIAL_SYNCS=TRUE in runcmd_env.sh
- Fixed Interval – This disables topology synchronization and invokes drivers at regular interval.

Enable or disable topology autosync

There are two methods by which you can disable auto synching of the topology:

- "Enable or disable topology autosync using the Console" on page 120
- "Enable or disable topology autosync using the configuration file" on page 121

Enable or disable topology autosync using the Console

To enable or disable topology autosync using the Global Manager Administration Console:

1. Open the **Global Manager Administration Console**.
2. From the **Manager List**, select the SAM Manager.
3. Expand **ICS Configuration**.
4. Expand **IC Domain Configuration**.
5. Expand **Domains**.
6. Click the appropriate domain.

7. The **Configure Domain** interface appears, as illustrated in Figure 10 on page 121.

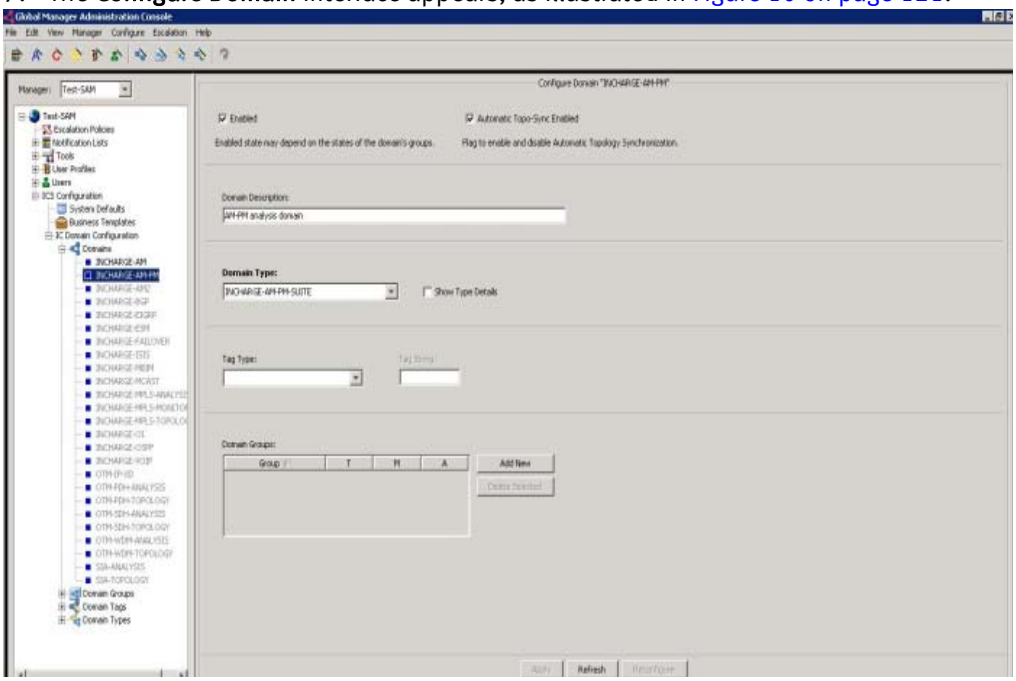

Figure 10 Global Manager Administration Console - Disable autosync

8. Select or clear the **Automatic Topo-Sync Enabled** check box based on whether you want to enable or disable autosync of the topology.

   NOTICE

   Automatic synchronization of topology is enabled by default.

9. Click **Apply › Reconfigure**.

Enable or disable topology autosync using the configuration file

NOTICE

Autosync of topology is enabled by default.

To disable topology autosync using the configuration file:

1. Add the following command at the end of the dxa file used for the domain:

   `topoUpdate ICF_TopologyManager::ICF-TopologyManager::fakeAttr`

   The sm_edit utility that is available in the bin directory, can be used to edit the dxa file.

2. Reconfigure or restart the server.

Enable topology synchronizations at regular intervals

To enable topology synchronizations at regular intervals add the following command to a cron script:

```
dmctl -b <broker-name> -s <server-name> invoke
GA_DaemonDriver::<Underlying-Domain-Name>_Topo-Driver start
```

Get a list of topology drivers

To get a list of topology drivers, execute the following command:

```
dmctl -b <broker-name> -s <server-name> getInstances
GA_DaemonDriver
```

The advantages and disadvantages of running topology synchronization in each mode are

Table 20 Advantages and disadvantages of each mode

Mode	Advantages	Disadvantages
Concurrent Synchronization	• Topology is updated with little or no delay. • Post processing is refined which improves the performance of server.	• Heavy load on server as topology synchronizations are triggered at higher frequency. • Memory growth might happen if overlapping topologies are synchronized at the same time.
Serial Synchronization	• Server Load is less on server as only one sync will be active at any point. • No issues related to memory growth.	• Topology may get updated with delay. Time delay is unpredictable as depends on how many domains are synchronizing at the same time. • Post processing gets invoked at the end of topology synchronization which might degrade the performance in large topology.
Fixed Interval	• Topology synchronization can be scheduled based on server load pattern. • Time delay in topology update is predictable.	• Topology is updated with delay.

mentioned in Table 20 on page 122.

If there are overlapping topologies between two or more underlying domains and top level SAM is configured such that it is synchronizing with underlying domains concurrently, then it is recommended to either run the server in serial mode or start topology drivers at regular intervals. The rule of thumb is to use the serial mode if there is more than 50% overlapping topology when synchronizations are triggered concurrently. When topology synchronization is set to operate in serial mode(non-default option), both import and post processing operations are considered as one atomic operation, and multiple topology synchronizations with underlying domains are executed serially.

Only one topology synchronization thread is executed at any point. This reduces the load on the server. The downside is a delay in completion of topology synchronization operation if multiple synchronizations are triggered at the same time.

Topology Timeout

The timeout (default timeoutTopoValue) parameter defines the default timeout to be applied to the front end and back end to which this driver is connected. If multiple drivers are connected, the maximum value across all of them defines the default. The timeout is always in seconds. A value of 0, requests no timeout. The value cannot be negative. If multiple drivers connect to the same front end or back end, a value of 0 requesting no timeout can be replaced by a non-zero value. If you want to force there to be no timeout, set the timeout to some very large value. For instance, you may leave the timeout value at the default value of 600 seconds. When the topodriver calls the remote server, it waits for 600 seconds to receive a response from the server. During topology synchronization if for some reason the underlying source domain stops responding to SAM, after a wait of 600 seconds, SAM will time-out and resume topology synchronization after the next topology discovery cycle.

The *TimeOutTopo ‹n›* parameter is present in the dxa configuration file of the corresponding domain (for example, dxa-sam.conf for SAM domains). The timeout value can be modified in the corresponding underlying domain dxa configuration file.

Ensuring a consistent representation of topology

A Global Manager imports topology information. Because the topology information comes from disparate sources, it is important that the Global Manager present a correct and consistent representation of the topology. This is especially true when two or more domains manage the same devices.

Newer versions of EMC Smarts applications include an expanded and updated list of certified devices. This may mean that a device classified as Uncertified or Node by an older version of the EMC Smarts software is classified as a Router or Switch by a newer version of the software. For best results, upgrade existing EMC Smarts applications to the most recent version. If that is not possible, use the same version of the software for all the underlying applications.

Two issues can arise when the Global Manager receives inconsistent topology information:

- The Global Manager imports information about two or more devices with the same name but the devices are classified differently in their respective underlying domains.
- The Global Manager imports the same device from two or more underlying domains but the device is named differently in each underlying domain.

NOTICE

The Global Manager does not import instances of the Unsupported and Undiscovered classes. However, the topology of a Global Manager may include instances, such as Hosts, with a value of Undiscovered for their Certification attribute.

Same device classified differently in separate underlying domains

The Global Manager can receive conflicting topology information when two underlying domains discover the same device but classify it differently. The scenarios where this may occur are as follows:

- A device has the same name, but is an instance of different classes in two or more underlying domains. In addition, the device is classified as Host, Uncertified, or Node in one or more of the underlying domains. When this occurs, the Global Manager replaces an instance of a less specific class, Host, Uncertified or Node, with an instance of a more specific class. Incoming event information for the device is consolidated to the instance in the Global Manager's topology.

  For example, device1.mydomain.com is classified as Uncertified in one underlying domain and classified as a Router in a second underlying domain. The Global Manager classifies the device as a Router in its topology. All incoming notifications related to device1.mydomain.com are associated with the Router instance in the Global Manager's topology.

- A device has the same name, but is an instance of different classes in two or more underlying domains. In this case, the device is not classified as Host, Uncertified, or Node in any underlying domain. The Global Manager classifies the device according to the first topology information it receives from an underlying domain. Relationship information for the device is updated during consecutive topology synchronizations. Incoming event information for the device is consolidated to the instance in the Global Manager's topology.

  For example, device2.mydomain.com is classified as a Probe in one underlying domain and a Router in a second underlying domain. If the first underlying domain synchronizes first with the Global Manager, the Global Manager classifies the device as a Probe in its topology. However, any relationship information for the router from the second underlying domain is preserved by the Global Manager and added to the Probe instance. All incoming notifications related to device2.mydomain.com are associated with the Probe instance in the Global Manager's topology.

NOTICE

If there is only one domain, the Global Manager will change the classification of an object when the domain changes it. For example, if IP Availability Manager discovers a device as a Probe, the Global Manager will create a Probe. If IP Availability Manager rediscovers the same device as a Switch, then Global Manager will delete the Probe and re-create the instance as a Switch.

Same device named differently in separate underlying domains

The Global Manager creates its topology based on the names of the devices it imports from the underlying domains. When the Global Manager imports the same device from two or more underlying domains with the same name, it creates a single corresponding device in its own topology. The Global Manager associates any incoming events from the underlying domains that are related to this device with the single device in its own topology.

When the Global Manager imports the same device from two or more underlying domains and each domain gives the device a different name, the Global Manager creates unique elements in its topology for each device.

For example:

- One underlying domain discovers a device and gives it the name device3.
- Another underlying domain discovers the same device but gives it the name device3.mydomain.com.

When the Global Manager receives topology information from these underlying domains, it creates two instances in its topology — one named device3 and one named device3.smarts.com.

Same device with different DisplayName

When two or more underlying domains manage the same instance and the value of DisplayName is different in each underlying domain, the value of DisplayName in the Global Manager changes each time it synchronizes with the other underlying domain. The same issue applies to notification display names when an alarm is generated on the instance by one of the underlying domains.

To fix the display name to a single value, specify the keyword update in the underlying domains from which the value of DisplayName should be taken in the smarts/conf/ics/ics-default.xml file, as shown in the following example:

```
<domainconfig>
        <domaintype
            name="MPLS"
            description="Type for MPLS"
            dxaConfFile="dxa-mpls-vpn-conn.conf"
            MinimumCertainty="0.24"
          update=”True”
            SmoothingInterval="65">
            <hook enable="False">ics/dxa-sample-hook.asl</hook>
            <domainname>INCHARGE-MPLS</domainname>
</domaintype>
```

Organizing topology with groups

Grouping provides a method by which you can organize topology elements. With Service Assurance Manager, you can create groups and organize topology elements to more efficiently manage large numbers of elements.

Before you start, it might be useful to devise a strategy to organize topology elements into groups. Common strategies include organizing by the following:

- Business units
- Geographical regions
- Resources

General properties of groups

A group is a user-defined collection of instances from the Global Manager's topology. A group consists of members or child groups. A member is a topological element such as a switch. A child group (or subgroup) is another group, which may be a collection of members or additional subgroups. A group that contains child groups is referred to as the parent group.

Parent and child groups are organized into a tree structure. At the root of each tree is a top-level group. Each top-level group is a distinct organization of groups and members—its configuration or removal does not effect other top-level groups.

In the Map Console, top-level groups are displayed directly beneath the icon for the Global Manager:

- A member is also a member of the groups above it within the same group hierarchy.
- Within a single group hierarchy, an element cannot be a member of more than one group. An element can belong to two or more groups that descend from different top-level groups.
- You cannot create a circular group where a group is specified as a subgroup of itself or one of its subgroups.

Before we explain how to create groups, you may find it helpful to understand how groups are displayed in the Map Console.

How groups are displayed in the Map Console

The Map Console only displays the members of a group when that group does not contain any child groups. If a group contains one or more child groups as well as members, only the child groups are displayed.

When you create groups to display them in the console maps, create a catch-all group within a parent group. The catch-all group should contain all the members that do not belong to any of the other subgroups. This technique prevents a parent group from containing both members and child groups.

Types of groups

The Global Manager supports two types of groups: selective groups and hierarchical groups:

- A selective group is a group whose members are determined by a matching pattern that you specify through the Global Console.
- A hierarchical group is specified in a data file which is then imported by the Global Manager.

In addition to the methods by which they are created, there exist several other differences between selective and hierarchical groups.

Properties of selective groups

Selective groups contain three properties that distinguish them from hierarchical groups: matching criteria, priority, and target classes. These properties help to determine what elements become members of a selective group.

Matching criteria

Matching criteria are attributes defined in the ICIM model that you use to determine what elements are eligible to become a member of a group. When you create a selective group, you specify a matching pattern that is compared against the attributes of the element. If the pattern matches the value of the specified attribute, the element becomes a member of the group. A matching pattern comprises one or more characters and wildcards. If you do not specify a matching pattern, all managed elements that pass the target class filter match the group. Priority will determine if any elements become members.

Table 21 on page 127 lists attributes against which you can apply a matching pattern. The group's target class determines which attributes are available to match against. For example, if the target class is IPNetwork, attributes that describe a managed system, such as Certification, are not listed.

Table 21 Attributes for matching criteria (page 1 of 2)

Attribute	Description
Certification	Level of certification assigned to this device during discovery. Values include: UNCERTIFIED, GENERIC, TEMPLATE, CERTIFIED, or VALIDATED.
CreationClassName	Name of the class of which the managed element is an instance. This is used as the ClassDisplayName attribute in notifications affecting this element.
Description	A brief description of the element.
DisplayClassName	Same as creation class name.
DisplayName	Name of the managed element. For systems, DisplayName and Name are usually the same.

Table 21 Attributes for matching criteria (page 2 of 2)

Attribute	Description
IsManaged	Determines if the system is managed by the underlying domain. Value is TRUE or FALSE. **Notice:** Unmanaged elements do not appear in the Global Manager topology.
Location	A brief textual description of the system's physical location.
Model	Vendor's name for the system.
Name	Name of the managed element. For systems, Name and DisplayName are usually the same.
PrimaryOwnerContact	Information on how to contact the system's owner.
PrimaryOwnerName	Name of the system's owner.
ServiceName	Name of external system used to import attributes and events.
SystemName	Name of the system that contains this managed element.
Type	Classifies the type of system. Possible values include: Bridge, Host, Hub, Node, Other, Probe, Router, RSFC, RSM, Switch, and TerminalServer.
Vendor	Name of the system's manufacturer.

"Wildcard Patterns" on page 219 describes wildcards for matching patterns.

Priority

Priority distinguishes between groups at the same level of the hierarchy with the same parent. Each such group is automatically assigned a different priority. When a topology element matches the pattern of two different groups, it becomes a member of the group with the higher priority. Because of this, you should assign a higher priority to a group with a stricter matching pattern. If a group has a high priority and it matches all the topology elements, it will contain all the available members.

Target classes

A target class acts like a filter, allowing only those elements that are instances of the target class, or one of its subclasses, to become members of the group. Managed elements must pass the target class filter before they are compared against the matching criteria.

When you create a child group, the child group should have the same target class as its parent. The exception to this rule is when the new target class is a subclass of the parent group's target class. For example, the ICIM_ManagedSystemElement class is near the top of the ICIM hierarchy. All of the other target classes are a subclass of ICIM_ManagedSystemElement. If ICIM_ManagedSystemElement is the target class of a parent group, you can select a different target class for a child group.

Similarly, VLAN, NetworkConnection, and IPNetwork are subclasses of ICIM_LogicalLink. If ICIM_LogicalLink is the target class of the parent, you can choose one of these three classes as the target class for a child group.

Table 22 on page 129 lists the target classes you can assign to a selective group when you create it. Classes are listed in the order that they appear in the ICIM class hierarchy. The description indicates when a class is a subclass of another target class.

Table 22 Target classes for selective groups

Target class	Description
ICIM_ManagedElement	Base class for the ICIM system element hierarchy. This is the broadest target class.
UnitaryComputerSystem	Represents a single computer system. This is the superclass for the Bridge, Host, Hub, Probe, MSFC, Router, RSFC, RSM, Switch, TerminalServer, and Node classes.
ApplicationService	Represents service provided by software. Examples include email, web server, and database applications.
LogicalLink	Represents a link between two endpoints. Examples include database transactions, IP networks, and cables.
VLAN	Virtual LAN typical in switched networks. VLAN is a subclass of LogicalLink.
NetworkConnection	A connection between two routers, typically a virtual circuit. NetworkConnection is a subclass of LogicalLink.
IPNetwork	Subnet of an IP network. IPNetwork is a subclass of LogicalLink.
ServiceSubscriber	Customer who receives services provided through a service offering.
ServiceOffering	A service provided to customers.

Creating selective groups

You create and edit selective groups by using the Global Console. Creating or editing groups through the console requires administrator privileges from the Global Manager. Operators with monitoring privileges can view groups in the Topology Browser and Map Console, but are not able to create or edit groups.

You should be aware of the following points when creating selective groups:

- Top-level selective groups cannot contain members, only child groups. As such, you need to create both a top-level group and one or more child groups to assign topology elements to a group.
- By default, each top-level group contains all the topology elements that match its target class and matching criteria.
- Each group is identified by a unique name. The name is displayed in group maps and in the Topology Browser. It is the value of the DisplayName attribute for the group instance.
- Hierarchical groups are visible in the Group Definition window, but they cannot be edited.

Layout of the Group Definition window

You create and edit groups through the Group Definition window. You open this window by selecting Groups from the Configure menu of the Global Console.

The Group Definition window is divided into two panels. The left panel displays the Global Manager, groups, and the group members. When you choose a selective group in the left panel, the Properties, Priorities, and Matching Criteria tabs display in the right panel of the Group Definition window.

NOTICE

When you select a top-level group, only the Properties tab is displayed. Top-level selective groups do not have priority or matching criteria.

Priority and matching criteria determine which topology elements are members of each group. If you have worked with configuration groups (Polling Groups and Threshold Groups) for EMC Smarts applications, the priority and matching criteria for topology groups function similarly.

"Properties of selective groups" on page 127 describe priority and matching criteria.

The toolbar of the Group Definition window contains four buttons:

- Delete removes the specified group from the topology of the Global Manager. If the deleted group contains child groups, the child groups are also deleted.

NOTICE

The Delete command does not remove hierarchical groups.

- Regroup tells the Global Manager to rebuild the selective groups from the selected group down to the bottom of the group hierarchy. If you select the Global Manager icon and invoke Regroup, the Global Manager rebuilds all of the selective groups. This, however, is not usually necessary. It is more efficient to regroup a section of the group hierarchy when there are large number of topology elements in each group.

  You need to invoke Regroup after you make changes to the priority or matching criteria of a group. The Global Manager automatically regroups the topology when it synchronizes its topology with the underlying domains.

NOTICE

The Regroup command does not affect hierarchical groups.

- Create Top Level Group displays the New Group dialog box where you specify the name, description, and target class for the group. This command is only available when the Global Manager is selected.
- Create New Group displays the New Group dialog box where you specify the name, description, and target class for a child group. This command is available when a group is selected in the group tree hierarchy in the left panel of the Group Definition window.

You can also find these commands under the Group menu. In addition, the Group menu also contains the Save Groups command. When you invoke Save Groups, the Global Manager saves its in-memory database to the repository file.

Method for creating selective groups

To create a selective group:

1. Select **Groups** from the Configure menu of the Global Console. The **Group Definition** window appears.

   NOTICE

   This requires administrator privileges. The Configure menu is not displayed for users with monitoring privileges.

2. To create a top-level group, select the **Global Manager** icon and click the **Create Top Level Group** toolbar button.

   To create a child group, select the parent group and click the **Create New Group** toolbar button. The **New Group** dialog box appears.

3. Specify a name, description, and target class for the group. After you finish specifying values for these three fields, click **OK**.

   The new group displays in the **Group Definition** window. You can also see the group as an instance of the SelectiveGroup class in the Topology Browser Console or as a group in the **Groups** tab of the Map Console.

4. Change the priority of the new group. If there are other groups at the same level of the group hierarchy, they are listed under the **Priorities** tab. By default, a new group is assigned the lowest priority.

   To change a group's priority, select the group whose priority you want to change. Under the **Priorities** tab, click the up arrow to give the group a higher priority or click the down arrow to give the group a lower priority.

5. Click the **Matching Criteria** tab and specify a matching pattern for the group. By default, a new group does not contain matching criteria, meaning it matches all possible elements.

   For example, if you want to create a group that includes systems from a certain geographical area, then you might match against the value of the Type and Location attributes.

   For example:

   - Specify a pattern that matches the type of systems you want to group.

     To create a group of routers you would add Type as a matching criteria attribute and specify a matching pattern of Router.

   - Specify a pattern that matches against the geographical locale specified in the system's Location attribute.

     To match against routers in New York, you could add Location as a matching criteria attribute and specify a pattern of "*NY*|*New York*". This pattern would match against "NY" and "New York" anywhere in the Location attribute. Table 21 on page 127 lists the classes whose values you can match against.

6. Select the parent for the newly created group, or the Global Manager, and click the **Regroup** toolbar button.

NOTICE

To create a catch-all group, create a group with no matching criteria and assign it the lowest priority.

Editing the properties of a selective group

You can edit the description, target class, priority, and matching criteria of a selective group. Similar to creating a group, you must attach to Global Manager with administrative privileges and select Groups from the Configure menu.

You can edit all the properties of a group before applying the changes. However, if you select another group before clicking Apply, your changes are not applied. In this case, the console displays a dialog box asking if you want to abandon the changes that have not been applied.

To edit the properties of a selective group:

1. Select the group whose properties you want to edit.

   The properties of the group appear in the right panel of the **Group Definition** window.

   The description and target class of the group appear under the **Properties** tab.

2. Do one of the following:
   - To change the target class, select a class from the pop-up menu.
   - To change the description, edit the text in the **Description** text box.
3. Select the **Priority** tab to change the priority of the group.
4. Select the **Matching Criteria** tab to change the matching pattern of the group.
5. Click **Apply**. If you change the target class, priority, or matching criteria, you also need to click **Regroup**.

Creating hierarchical groups

This section describes how to create hierarchical groups, including the syntax of the group datafile and how to load the group information into the Global Manager.

To create hierarchical groups:

1. Create one or more hierarchical group datafiles that specify the groups and their members. You can also add new groups and their members to existing hierarchical group datafiles, including the two example group datafiles (topology-group.data.template and service.data.template) that are provided with the system. To use one of the example group datafiles, make a copy of the example file and rename it appropriately. These data files must be located in the *BASEDIR*/smarts/local/conf/ics directory. "Syntax of hierarchical group datafiles" on page 133 describes the syntax of the group data files.
2. If necessary, edit the Business Templates options in the Global Manager Administration Console to add or delete the names of hierarchical group datafiles.
3. If hierarchical group files are added or deleted, reconfigure the Global Manager. "Reconfiguring the server" on page 106 describes how to reconfigure the Global Manager.
4. Import the group definitions by regrouping the group datafiles. "Regrouping hierarchical group data" on page 135 describes how to regroup the datafiles.

The Global Manager processes the group and service topology data specified in the BusinessSection. If the Global Manager does not encounter any errors, it does the following:

- Adds any new group and service information to its topology
- Deletes any group and service information that is in the current topology but which has been deleted in the new hierarchical group datafiles being imported

NOTICE

Both reconfiguring and regrouping update the service topology used by Business Impact Manager.

After the hierarchical group data is loaded into the Global Manager, you can view hierarchical groups through the Global Console.

Syntax of hierarchical group datafiles

The syntax for group datafiles provide a means for specifying groups, children (subgroups), and members. The following example illustrates the syntax of a hierarchical group datafile:

```
HierarchicalGroup NewYork children NY-Routers "NY Customers"
HierarchicalGroup NewYork children Queens Nassau HierarchicalGroup
   NewYork children Bronx
   HierarchicalGroup NewYork members Router::nyc1
HierarchicalGroup NY-Routers members Router::nyc1
HierarchicalGroup "NY Customers" members
   file:/opt/dev/incharge-sa/smarts/local/conf/ics/ny-customers.member
   s
```

This example illustrates the following syntactic rules of hierarchical group datafiles:

- Each line that specifies a hierarchical group must start with the keyword HierarchicalGroup.
- Elements of a hierarchical group must be separated by one or more spaces.
- The name of the group follows the keyword HierarchicalGroup. The name of the group must be unique for all existing groups. If the same group name is used in multiple lines, each line is referring to the same instance. In the example above, all four lines starting with “HierarchicalGroup NewYork” are defining either children or members of the NewYork group.
- If the name of a group contains a space, it must be enclosed in double quotes.
- The keyword children indicates that this line specifies subgroups of the named group. For example, NY-Routers and “NY Customers” are child groups, or subgroups, of the group named NewYork.
- The keyword members indicates that this line specifies members of the named group. For example, the router nyc1 is a member of the group named NY-Routers. You must specify the class name and the instance name of the topology element, separating them with a double colon (::).
- You cannot specify child groups and members in the same line. You can, however, use multiple lines to specify members or children for the same group.
- You can specify a list of members in a member file. For example, the file ny-customers.members lists members of the “NY Customers” group. However, you must specify the full path to the file, by using the correct syntax for the host operating system.

The following example shows the syntax of a file that lists the members of a group. Because the hierarchical group datafile specifies the name of the group, you only need to specify the ‹*class*›::‹*instance*› pairs, one per line, for each member, for example:

```
Router::nycbrd1
Router::nycbrd3
Switch::nycs2
```

Reconfiguring the Global Manager

“Reconfiguring the server” on page 106 provides additional information on reconfiguring the Global Manager.

Regrouping hierarchical group data

To regroup the hierarchical group and service topology datafiles, invoke the following command from the *BASEDIR*/smarts/bin directory:

```
▼dmctl -s <global_manager> invoke GA_DaemonDriver::
      ICS-Group-Driver start ▲
```

NOTICE

▼▲ The command must be typed as one line.

Depending on your security configuration, you might be prompted for your EMC Smarts username and password.

When the Global Manager regroups the topology information in the datafiles, it first checks for errors. If the Global Manager encounters an error, it writes output to the terminal or its log file and does not process the datafiles.

Purging of HierarchicalGroupManager instance

The HierarchicalGroupManager instance maintains relationships with HierarchicalGroups and SelectiveGroups. SelectiveGroups are created using SAM console and HierarchicalGroups are created from configuration files. Once HierarchicalGroupManager instance is purged, the group instances are not visible since the relationship information is lost even though group instances are present in the repository.

To recreate SelectiveGroups, run the following command:

```
./sm_adapter -s INCHARGE-linux ics/ics-restore-selectivegroup.asl
```

In case of HierarchicalGroup instances, perform the following steps to recreate them:

1. In the Global Manager Administration Console, expand **INCHARGE-SA** › **ICS Configuration** and click **Business Templates**.
2. Click **Modify List** on the **Configure "Business Templates"** screen.
3. On the **Edit Business Template** dialog-box, select a template from **Selected Template Files** and remove it.
4. From **Available Template Files**, add the template you just removed.
5. Click **OK** on the **Edit Business Template** dialog-box.
6. Click **Apply** on the **Configure "Business Templates"** screen.
7. Start the group driver by running the following command:

   dmctl -s INCHARGE-SA invoke GA_DaemonDriver::ICS-Group-Driver start

Custom icons for map nodes

An administrator with appropriate privileges can add or remove a custom icon to a class of elements or to individual elements. A custom icon can be used to customize maps for your organization and to meet the needs of your operators.

The Global Console includes a default set of icons for all ICIM elements. Icons can be associated with classes or instances.

If an icon is not associated specifically with a class, the icon for the parent class is used; this is the derived icon. In addition, the icon for a given class could be associated multiple levels up in the class hierarchy.

The set of icons shipped with the console does not include one icon for each class. The derived icon (from the parent class) is used for those classes that do not have a specific icon assigned to them.

You can use two methods to assign icons to class or instances:

- Use the Edit Map Icons operation in either the **Global Manager Administration Console** or the **Map Console**.
- Use the **sm_config** utility to specify the name of the map icon specification file. Appendix C, "XML Reference," describes how to assign map icons using **sm_config**.

NOTICE

Map icon text defaults to Helvetica 12 point. You can customize the map icon text. "Adjusting Java properties" in the *EMC Smarts Service Assurance Manager Operator Guide* provides detailed information.

Assigning icons to classes or instances

Before assigning images to map nodes, you must create the images and save them to the proper directory on the host where the Global Manager is running. Note that the icons included in the console are not loaded over the network.

The requirements for custom images are:

- Images must be in GIF format.
- Images must be 56 pixels square.

After you obtain the custom images, copy them to the *BASEDIR*/smarts/local/images/icons directory. Images located in this directory are automatically displayed in the icon configuration dialog box. The filename of the image, minus any file extension, is used to identify the image.

If an image file located on the Global Manager has the same name as an icon included with the console, the image located on the Global Manager takes precedence.

Assigning an icon to a class

To assign an icon to a class:

1. Select **Configure** › **Edit Map Icons** operation in the Global Manager Administration Console.

   The **Select Icons for:** ‹*Global Manager*› dialog box appears.

2. Select the class in the **Classes** list box.

   The **Classes** list box displays the classes of managed elements. By default, all classes are assigned an icon. However, it might be an icon associated with a parent class. Certain classes are assigned the same icon as their parent class. For example, the Redundancy Group icon is used for the System Redundancy Group, Network Connection Redundancy Group, and Network Adapter Redundancy Group. You can determine what icon is used for the System Redundancy Group by selecting this class in the Classes list and seeing what is displayed as the derived icon. The name of the class to which the derived icon is assigned is also displayed.

3. Click **Use the specific icon**.
4. Scroll through the list at the bottom of the dialog box to choose an icon.
5. Click **Apply**.
6. When you are finished, click **OK**.

Assigning an icon to an instance

You can assign an icon to an instance without changing the class icon.

To assign an icon to an instance:

1. Select **Configure** › **Edit Map Icons** operation in the **Global Manager Administration Console**.

   The **Select Icons for:** ‹*Global Manager*› dialog box appears.

2. Select the appropriate class in the **Classes** list box.
3. Select **Select Instances**.

   The **Instances** list box displays the instances of the selected class in the topology.

4. Select the instance in the **Instances** list box.
5. Click **Use the specific icon**.
6. Scroll through the list at the bottom of the dialog box to choose an icon.
7. Click **Apply**.
8. When you are finished, click **OK**.

Reassigning the derived icon to a class

To reassign the derived icon to a class:

1. Select **Configure** › **Edit Map Icons** operation in the **Global Manager Administration Console**.

   The **Select Icons for:** *‹Global Manager›* dialog box appears.
2. Select the class in the **Classes** list.
3. Click **Use the derived icon**.
4. Click **Apply**.
5. When you are finished, click **OK**.

Reassigning the derived icon to an instance

To reassign the derived icon to an instance:

1. Select **Configure** › **Edit Map Icons** operation in the **Global Manager Administration Console**.

   The **Select Icons for:** *‹Global Manager›* dialog box appears.
2. Select the class in the **Classes** list.
3. Click the **Select Instances** checkbox and then select the instance.
4. Click **Use the derived icon**.
5. Click **Apply**.
6. When you are finished, click **OK**.

CHAPTER 8
Tool Configuration for the Global Manager

This chapter contains the following information:

- Types of program tools 140
- How tools are invoked 141
- Security considerations for tools 142
- Sample tool scripts 143
- Creating tools 145
- Running automated tools with sm_adapter 155
- Running tools over X Windows 156
- Adding domain configurations 156

Types of program tools

A tool is a script that is typically invoked in response to a notification. Such a response might be to ping an affected device or to open a trouble ticket. Tools can also be invoked on topology elements. Service Assurance supports three types of tools:

- Server tools
- Client tools
- Automated tools

Server and client tools are associated with a user profile and invoked by an operator using the Global Console. Two tool submenus, Server Tools and Client Tools, appear in a pop-up menu when an operator right-clicks a notification or a device. Access to server and client tools can be controlled by associating or not associating tools to user profiles:

- The Server Tools menu in the Global Console displays a list of available server tools. Server tools are invoked by the Global Manager and available to any console user. For the tools to be available for a user, the user's profile must include the tools. "Security considerations for tools" on page 142 provides additional information.
- Client Tools are invoked on the host where the Global Console is running, which requires that the tool scripts are installed on the host where the console is running. The Client Tools menu displays a list of available client tools, provided the user's profile includes the tool.

  NOTICE

  Client tools can also be invoked from the web console. The *EMC Smarts Service Assurance Manager Dashboard Configuration Guide* describes how to configure client tools for the web console and the Business Dashboard.

- An automated tool is a type of server tool that is invoked by an escalation policy or an adapter. Automated tools are not visible to users of the Global Console.

Additional Service Assurance tools

In addition, Service Assurance provides the following tools that are described in separate documents. They include:

- Business Process Tools. The *EMC Smarts Business Impact Manager User Guide* provides additional information.
- Report Manager tools. The *EMC Smarts Report Manager User Guide* provides additional information.
- Remedy tools. The *EMC Smarts Adapter for Remedy ARS User Guide* provides additional information.
- Concord eHealth tools. The *EMC Smarts Adapter for Concord eHealth User Guide* provides additional information.

How tools are invoked

This section describes how to invoke tools and where the output of server and client tools appears.

Invoking server and client tools

Server and client tools are invoked through the Global Console. Such tools are always invoked on a particular target object by the console operator. The target object can be a notification or a topology element such as a router. When an operator right-clicks the target, the pop-up menu lists the available tools. The *EMC Smarts Service Assurance Manager Operator Guide* describes how to invoke tools from the Global Console.

Viewing the output of a server or client tool

Server and client tools, depending on their purpose, might produce output. For example, the output of a ping tool indicates whether the ping completed successfully.

For client and server tools, you can configure whether a tool displays any output. When a server tool is configured to display output, the output is echoed to stdout and collected by the Global Manager. The Global Manager sends the output back to the console where the tool was invoked.

The output of server and client tools is displayed in the Tool Output window. The Tool Output window opens when the tool is invoked. Output is written to the window while the tool is running. If a tool produces an error, the error message is also displayed in the Tool Output window. If an error occurs for a server tool that is not configured to display output, the Tool Output window displays the return code of the tool.

NOTICE

If the SAM server is started by attaching the server to a broker address that is different from the one specified during SAM installation, then the SM_BROKER variable for the non-default broker must be specified in the *local/conf/runcmd_env.sh* file. If the SM_BROKER variable for the non-default broker is not specified, the server tool will not work as expected.

NOTICE

The Tool Output dialog box includes a Stop button. Client tools, such as shell scripts that do not respond to a SIGTERM signal will not stop when an operator clicks Stop. Create a client tool script that responds to a SIGTERM signal or manually stop the process from the command line or from the Task Manager.

Invoking automated tools

An automated tool is run when it is triggered based on the settings of an escalation policy. An escalation policy is configured to invoke automatic tools to run on notifications that meet the escalation criteria for specified periods of time. Chapter 9, "Escalation Configuration for the Global Manager," includes additional information on configuring the Global Manager.

Alternatively, an automated tool runs with the automatic action adapter on the same host as the Global Manager. Start this adapter by invoking the **sm_adapter** command and supplying a list of arguments. After starting the automatic action adapter, no further intervention is required. The adapter uses a notification list to listen for particular notifications. When it receives a matching notification, it automatically responds by invoking the specified tools through the Global Manager.

Once started, the automatic action adapter runs in the background and responds to notifications that match its notification list. For example, the automatic action adapter, configured with a trouble-ticket system tool, could respond to a notification by opening a trouble ticket. "Modifying a tool" on page 154 includes additional information about creating an automated tool.

Information recorded to a notification's audit log

When a server or automated tool is invoked on a notification, information about the tool is recorded to the notification's audit log. This information includes:

- Date and time when the tool was invoked.
- Name of user that invoked the tool.
- Whether the tool completed successfully.
- Name of the tool.

Security considerations for tools

If a tool invokes an EMC Smarts utility, such as dmctl, or otherwise initiates a connection to the Global Manager, you must configure security to enable that connection. An essential piece of information is knowing the username under which the tool is invoked, for example:

- Client tools are invoked under the username of the operator who started the Global Console.
- Server tools are invoked under the username that the Global Manager is running under. Remember, server tools are invoked on the same host as the Global Manager.
- Automated tools are invoked under the username that started the **sm_adapter** command or by the name of the Global Manager.

If a client tool needs to connect to the Global Manager and perform an action that requires administrative privileges, you have two choices:

- Provide the operator with administrative privileges.
- Create a server tool that performs the action.

As the server tools run under the username that invoked the Global Manager process, this user must have an authentication record in the clientConnect.conf and serverConnect.conf files on the host where the tool is to be executed. You must configure the authentication records for this account so that prompting is not required. Prompting should not be used for automated tools.

Sample tool scripts

EMC Smarts provides a number of sample tool scripts that, with minor modifications, work on most systems. The main purpose of these scripts, however, is to provide examples that you can examine when developing your own tools.

NOTICE

The ping and Telnet tools are sample scripts. You need to customize them to meet the needs of your environment. For example, the sample scripts do not take firewalls into account.

Table 23 on page 143 lists the sample server and automated tools, the display name for the tool as seen by users in the Global Console, and provides a brief description. Server and automated tools are located in the *BASEDIR*/smarts/actions/server directory.

To modify one of the sample tool scripts, open it with the **sm_edit** utility to create a local copy of the script.

Table 23 Sample server tool scripts (page 1 of 2)

Server tool script	Display name for tool	Description
ics-closetkt	Sample - Close Trouble Ticket	This tool inserts a static value into the field of a notification. The context and Status Criteria specify that this tool is only active for notifications with the text OPEN in the TroubleTicketID field.
ics-opentkt	Sample - Open Trouble Ticket	This tool inserts a static value into the field of a notification. The context and Status Criteria specify that this tool is only active for notifications without the text OPEN in the TroubleTicketID field.
ics-ping-interface	Sample - Ping-Interface	This tool pings an IP interface. The Context Criteria specify that this tool is active for notifications with a value of “Interface” for the ClassName attribute.
ics-ping-IP	Sample - Ping-IP	This tool pings an IP interface. The Context Criteria specify that this tool is active for notifications with a value of “IP” for the ClassName attribute.

Table 23 Sample server tool scripts (page 2 of 2)

Server tool script	Display name for tool	Description
ics-ping-all	Sample - Ping-all	This tool pings all the IP interfaces associated with a device. The Context Criteria specify that this tool is active when the target is a UnitaryComputerSystem. As such, valid targets include both notifications and instances. When invoked, this tool retrieves all of the IP addresses associated with the target instance from one or more underlying domains. Two variables that control the behavior of the tool: • PING_ALL_DOMAINS controls whether the tool retrieves IP addresses from each underlying domain that manages the instance. When set to 0, the default, the tool retrieves IP addresses from a single domain. When set to 1, the tool retrieves IP addresses from all domains. • PING_ONCE controls whether the tool pings each retrieved IP address or whether it stops after the first successful ping. When set to 0, the tool stops after the first successful ping. When set to 1, the default, the tool pings all available IP addresses.
ics-ping-device	Sample - Ping-device	This tool pings the IP address of the SNMP agent for the affected element. The Context Criteria specify that this tool is active when the target is a UnitaryComputerSystem. As such, valid targets include both notifications and instances. When invoked, this tool retrieves the IP address of the SNMP agent associated with the target instance from one or more underlying domains. The tools script includes one variable that controls the behavior of the tool: • PING_ALL_DOMAINS controls whether the tool retrieves IP addresses from each underlying domain that manages the instance. When set to 0, the default, the tool retrieves IP addresses from a single domain. When set to 1, the tool retrieves IP addresses from all domains.
ics-telnet	Sample - Telnet	This tool opens a Telnet session with the affected device. The .sh version of this script pings an IP address of the affected device before attempting to connect. If the ping fails, the script pings another IP address on the device, and so on until a ping succeeds or all the pings fail. When a ping succeeds, the script connects to the IP address where the ping succeeded. If this tool is invoked from a console that is running over X Windows, the script opens a separate window on the user's display to invoke the Telnet session.

Table 24 on page 145 lists the sample client tools and provides a brief description of each. Client tools are located in the *BASEDIR*/smarts/actions/client directory.

Table 24 Sample client tools scripts

Client tool	Description
SmGetEnv	This tool parses the environment variables passed by the Global Manager to the tool script. It also prints all environment variables to the Tool Output window. A version of this tool script is also provided in Perl.
SmLaunchPerlScript	This tool launches the *SmGetEnv.pl* tool script to illustrate how you can launch Perl scripts from a tool. You must edit this script to specify the location where Service Assurance is installed.
browser	This tool opens a web browser. You must edit this script to specify the location where your browser is installed.
ciscoworks	This tool opens a web browser that loads the administration page for CiscoWorks. You must edit this script to specify the location of your browser and the host where CiscoWorks is running.
pinger	This tool pings the name of the specified element.
Reporting	This action accesses reports available through the Service Assurance Report Manager. It opens a web browser with the URL set to the location of the Crystal Enterprise EportFolio application. You must edit this script to specify both the web browser and the URL.

Remember that tool scripts are system-specific. On Windows systems, a tool is typically invoked by the Windows command interpreter (cmd.exe). You can also write a tool script in a language such as Perl, which runs on different platforms with little or no modification.

Creating tools

This section provides the information necessary to create and configure a tool script.

Follow these steps to develop and use a tool script:

1. Determine the action to be performed by the tool and write the script.
2. Save the script to the proper location.
3. Test the tool script to ensure that it executes properly.
4. Assign the proper read and execute permissions to the tool script.
5. Configure the tool to invoke the script. In addition, client and server tools must be associated with the appropriate user profile. For automated tools, you must create an adapter or escalation policy to invoke the tool.

You can configure tools by using one of the following tools:

- Global Manager Administration Console, as described in this section.
- The **sm_config** utility, "Importing and Exporting Configurations" on page 189 provides additional information.

How data is passed to a tool script

When a tool is invoked, the attributes of the tool target are automatically passed to the tool script. The attributes are passed to the tool script as environment variables in the form:

```
SM_OBJ_<NAME>=<value>
```

The ‹*NAME*› parameter identifies the attribute and the ‹*value*› parameter contains the attribute's value. For example, the ClassName attribute would be passed in the environment variable:

```
SM_OBJ_CLASSNAME=Router
```

Your tool script must parse these environment variables and extract the information required by the program invoked by the tool script. For example, if you want a tool that telnets to a device, you need to extract the environment variable that contains the name or IP address of the device, as shown in the example code (in Windows CMD script):

```
rem if SM_OBJ_Name isn't set in environment report an error
if not defined SM_OBJ_Name (
   echo Invalid SM_OBJ_NAME
   exit 0
)

rem Use the value of SM_OBJ_Name to connect to the device
start /wait cmd /c "telnet %SM_OBJ_Name% "
```

The tool target can either be a notification or an instance. For both types of targets, a tool script receives the attributes listed in Table 25 on page 146 as environment variables.

Table 25 Attributes passed to all tool scripts

Environment variable	Description
SM_REMOTE_USER	Specifies the EMC Smarts username of the person that invoked the tool.
SM_SERVER_NAME	Specifies the name of the Global Manager that executes the tool script.

Table 26 on page 146 lists the attributes that are passed to the tool script when the target is an instance.

Table 26 Attributes for element targets

Element target attributes	Description
SM_OBJ_CLASS_NAME	Class name of the system. For example, Router.
SM_OBJ_INSTANCE_NAME	Name of the system.
SM_OBJ_DOMAIN_NAME	Name of the Global Manager.

Table 27 on page 147 lists the attributes that are passed to the tool script when the target is a notification.

Table 27 Attributes for notification targets (page 1 of 2)

Notification target attributes	Description
SM_OBJ_Acknowledged	Specifies whether the notification is acknowledged. TRUE if the notification has been acknowledged, FALSE if not.
SM_OBJ_Active	Specifies whether the notification is active. TRUE if the notification is active, FALSE if not.
SM_OBJ_Category	Type of notification sent by the Global Manager. Possible values include: • BackplaneUtilization • Error • Performance • PowerSupply • Resource • SystemConnectivity • Temperature • VLANConnectivity
SM_OBJ_Certainty	Confidence that this notification is the correct diagnosis. Value ranges from 0 to 100.
SM_OBJ_ClassDisplayName	Name of the ClassName that is displayed to the user.
SM_OBJ_ClassName	Name of the class where the event occurred. Might not be the same as SM_OBJ_ClassDisplayName.
SM_OBJ_ElementClassName	Class name of the managed element most closely related to this event.
SM_OBJ_ElementName	Name of the managed element most closely related to this event.
SM_OBJ_EventDisplayName	Name of the EventName that is displayed to the user.
SM_OBJ_EventName	Name of the event. Might not be the same as SM_OBJ_EventDisplayName.
SM_OBJ_EventText	A description of the notification.
SM_OBJ_EventType	MOMENTARY when the notification has no duration. DURABLE if the notification has a period for which it is active, such as a link failure.
SM_EventState	State of the event. Possible values include: • ACTIVE • WAS_ACTIVE • SUSPENDED • INACTIVE The UNITIALIZED state is not relevant here.
SM_OBJ_FirstNotifiedAt	Time, in seconds, when the notification first became active.

Table 27 Attributes for notification targets (page 2 of 2)

Notification target attributes	Description
SM_OBJ_Impact	Numeric value that indicates the effect of this event on related elements.
SM_OBJ_InMaintenance	TRUE if the device is in maintenance mode, FALSE if not.
SM_OBJ_InstanceDisplayName	Name of the InstanceName that is displayed to the user.
SM_OBJ_InstanceName	Name of the instance where the event occurred. Might not be the same as SM_OBJ_InstanceDisplayName.
SM_OBJ_IsProblem	TRUE if (and only if) at least one of the types is “PROBLEM”, and none of the types are anything other than “PROBLEM” or “UNKNOWN”.
SM_OBJ_IsRoot	TRUE if the notification is a root cause, FALSE if not.
SM_OBJ_LastChangedAt	Time, in seconds, when the status of the notification last changed.
SM_OBJ_LastClearedAt	Time, in seconds, when the notification was last cleared.
SM_OBJ_LastNotifiedAt	Time, in seconds, when the notification was last notified.
SM_OBJ_Name	InternalEventHandle for the notification.
SM_OBJ_OccurrenceCount	Number of times the notification has occurred.
SM_OBJ_Owner	Name of the person responsible for this notification. Value is SYSTEM when acknowledged by the Global Manager.
SM_OBJ_Severity	Level of severity for this notification. 1 = CRITICAL 2 = MAJOR 3 = MINOR 4 = UNKNOWN 5 = NORMAL
SM_OBJ_SourceDomainName	Name of the underlying domain that sent this notification. If more than one domain is listed, the names are separated by commas.
SM_OBJ_SourceEventType	Indicates the type of event from an underlying source or sources. Value can be: PROBLEM, EVENT, AGGREGATE, or UNKNOWN. The default value is UNKNOWN.
SM_OBJ_TroubleTicketID	Trouble-ticket number associated with this notification.
SM_OBJ_UserDefined1-20	Twenty user-defined fields.You can populate these fields with data with an ASL hook script.

Table 28 on page 149 lists additional attributes that are passed to the tool script when the tool is invoked through an escalation policy.

Table 28 Attributes for escalation targets

Element target attributes	Description
SM_POBJ_POLICY	Name of the policy that executed the tool.
SM_POBJ_PATH	Name of the path that executed the tool.
SM_POBJ_LEVEL	Name of the level that executed the tool.

In addition to the attributes listed above, a server tool script receives the DISPLAY environment variable. The value of this variable is used to determine the location of the user's X Window System display. "Running tools over X Windows" on page 156 includes detailed information for invoking server tools through an X Windows System.

Where to save tool scripts

After you write a script, you need to save it to the proper location and ensure that it is executable. A server tool, which includes automated tools, must be located in the *BASEDIR*/smarts/local/actions/server directory on the host where the Global Manager is running. The Global Manager must be able to execute the script.

A client script must be located on the host where the Global Console is installed. The proper location is the *BASEDIR*/smarts/local/actions/client directory. The console user must be able to execute the script. Server and client tools are configured in the Global Manager Administration Console.

Configuring with the Global Manager Administration Console

New tools, or modifications to existing tools, are not available to console users until they restart the Global Console.

To configure a tool by using the Global Manager Administration Console:

1. Select the type of tool you need to configure from the **Edit** menu or click the appropriate toolbar button. This displays one of the following wizards:
   - Automated Tool Creation Wizard
   - Client Tool Creation Wizard
   - Server Tool Creation Wizard
2. Type a name for the tool. This name is displayed in the Global Manager Administration Console, and, for server and client tools, in the Global Console.
3. Do one of the following:
   - Click **Create New**.
   - Click **Copy Existing**.

     When you copy an existing tool, the configuration parameters for the tool are the same as the copied tool.
4. Click **Next**.

5. Define the following tool parameters:
   - Program to run
   - Time-out Interval
   - Trace
   - Display (client and server tools only)
   - User Profiles (client and server tools only)

   "Tool parameters" on page 150 describes these parameters.
6. Click **Next**.
7. Use the **Filter Builder** to specify the Context Criteria for a server or client tool and click **Next**.

   "Using the Filter Builder" on page 182, includes detailed information on the Filter Builder. "Context and Status Criteria for client and server tools" on page 151 includes additional information on Context and Status Criteria.
8. Use the **Filter Builder** to specify the Status Criteria for a server or client and click **Next** to advance to the confirmation screen.
9. Click **Finish** to create the tool. You will receive an error message or a success message based on the result.

Tool parameters

Table 29 on page 150 describes the configuration parameters for tools. Only a subset of these parameters are available for automated tools.

Table 29 **Server tool configuration parameters**

Parameter	Description
Name	Name of the tool that is displayed in the console.
Program	Specifies the name of the program that is invoked when the tool runs. Type the program name or select a program from the pop-up menu: • For client tools, this is a list of tools in the BASEDIR/smarts/actions/client and BASEDIR/smarts/local/actions/client directories on the host where the Global Manager Administration Console is running. • For server or automatic tools, this is a list of tools in the BASEDIR/smarts/actions/server and BASEDIR/smarts/local/actions/server directories. This parameter cannot contain path separators. This is a list box of existing server or client scripts. The name cannot contain path separators.
Timeout Interval	Specifies the number of seconds to wait for a tool script to complete. If the tool script does not complete within the specified time, the tool process is terminated. The default is 30 seconds.
Trace	Specifies whether trace output, used for debugging, should appear in the tool output window when a tool is run.

Table 29 Server tool configuration parameters (continued)

Parameter	Description
Display Output	Specifies whether a tool script should display output to the console. If this parameter is set to true, the console opens the Tool Output window immediately after invoking the tool. If false, the Tool Output window opens only if the tool returns an error. If this parameter is omitted, it defaults to true. This parameter is not applicable to automated tools.
User Profiles	Specifies the user profiles that include this tool. This parameter is not applicable to automated tools.
Context	Determines whether a tool displays in the console's pop-up menu when the user right-clicks on a tool target. A tool always displays when this parameter is omitted. This parameter is not applicable to automated tools.
Status	Determines whether a tool is enabled or disabled in the console's pop-up menu. A tool is always enabled when this parameter is omitted. This parameter is not applicable to automated tools.

Configuration of server tools prompts

Server tools include the capability of prompting for input when a user invokes the tool. The input provided by the user is passed to the server tool when it runs.

Administrators, using the Server Tool Wizard in the Global Manager Administration Console, can specify the parameters that are displayed to the user in the prompt. Each parameter includes three fields:

- Name, which is required, is the internal name of the parameter
- Text, which is the description displayed to the user
- Width, which specifies the visible size of the input field in characters

Parameters are passed from the Global Console to the script, by way of the Global Manager, as environment variables in the form SM_POBJ_‹*name*›, where ‹*name*› is the internal name of the parameter.

Context and Status Criteria for client and server tools

Context Criteria and Status Criteria determine whether a tool is applicable to the selected target. If an operator selects an instance, then the target is an instance. If an operator selects a notification, the target can be either the notification or the instance where the notification occurred, as identified by the ElementClass and ElementInstance attributes.

Context Criteria determine whether a tool is displayed in the menu when a target is selected. The console checks the Context Criteria when the notification is received and does not re-check until the notification is archived and re-notified. Hence, you must apply context checking to attributes whose values do not change:

- Category
- ClassDisplayName
- ClassName
- ElementClassName
- ElementName

- EventDisplayName
- EventName
- EventType
- InstanceDisplayName
- InstanceName

Although you can apply context checking to other attributes, the results are based on the initial value of the attribute.

Status Criteria determines whether a tool is active. The name of a disabled tool is dimmed in the menu and cannot be invoked. You should apply Status Criteria to attributes whose value might change, including:

- Acknowledged
- Active
- Certainty
- ClearOnAcknowledge
- EventState
- EventText
- Impact
- InMaintenance
- IsProblem
- IsRoot
- OccurrenceCount
- Owner
- Priority
- Severity
- SourceDomainName
- SourceEventType
- TroubleTicketID

The tool target determines what attributes are passed to the tool to be matched against the Context Criteria. For example:

- When the target is an element, you can match against the attributes specified in Table 26 on page 146.
- When the target is a notification, the matching criteria determine which attributes are passed to the Context Criteria.

Results differ depending on the matching criteria:

- If the matching criteria specify a notification attribute, then the attributes of a notification, from Table 27 on page 147, are passed.
- If the matching criteria specify an element, then the attributes of an element are passed.

Creating Context and Status Criteria

Chapter 10, "Working with Filters," provides information on specifying Context and Status Criteria by using the Filter Builder. You can use an expression filter or an ASL filter to specify context and Status Criteria. Unlike the filters you can create for notification lists, you can use either an expression filter or an ASL filter, but not both.

The following example illustrates creating Context Criteria using Filter Builder:

1. Select a class from the **Context** object list box. The filter expression is matched against one or more attributes of the selected class.

   For example, select the UnitaryComputerSystem class.

2. Add an attribute to the filter sheet to specify a wildcard expression against an attribute of UnitaryComputerSystem. This enables you to narrow the range of notifications for which the tool is displayed.

   If you do not add an attribute, the Context Criteria matches all instances of UnitaryComputerSystem or its subclasses.

3. Click **Next**. The filter sheet for Status Criteria appears.
4. Select a class from the **Status** object list box. The filter expression is matched against one or more attributes of the selected class.

   For example, to match Routers, select the **Router** class.

5. Add an attribute to the filter sheet to specify a wildcard expression against an attribute of the Router class.

   If do not add an attribute, the Status Criteria matches all instances of the Router class.

6. For example, select the **Vendor** attribute and type a value of CISCO. The Status and Context Criteria for this tool specify that it displays in the Server Tools menu for any system element or any notification for which the failing element is a system. If that system is a router with a value of CISCO for its Vendor attribute, the tool is active and can be invoked.
7. Do the following:
   - Click **Next** to view the confirmation screen.
   - Click **Finish** to create the tool configuration.

Modifying a tool

To modify an existing server, client, or automated tool:

1. Select the tool in the tree of the Global Manager Administration Console.

   The **Configure Tool** panel appears.

2. Modify the appropriate parameters.
3. Click **Apply** to synchronize the changes in the Global Manager.

Disabling a tool

You can disable server, client, and automated tools. This allows you to prevent the tool from being used. Disabled tools are not displayed in the Global Console and do not execute when launched through an escalation policy or through **sm_adapter**.

To disable a tool:

1. Select the appropriate client, server, or automated tool in the **Global Manager Administration Console**.
2. On the configuration panel, clear the **Enable** checkbox.

   The name of disabled tool displays in gray in the navigation tree.

3. Click **Apply**.

Changing the order in which tools are displayed

You can determine the order in which tools are displayed in the Server Tools and Client Tools menus of the Global Console.

To change the display order:

1. In the **Global Manager Administration Console**, select **Automatic**, **Client**, or **Server** under **Tools** in the tree.

   The **Prioritize Tools** panel appears.

2. Use the up and down arrows to move a tool up or down in the menu list.
3. Click **Apply**. The changes become effective immediately.

Running automated tools with sm_adapter

The configuration options for an automated tool are specified as arguments to the **sm_adapter** program when you invoke it at the command line. Table 30 on page 155 lists the arguments you should specify when invoking an automated tool.

Table 30 Arguments to sm_adapter for automated action

Argument	Default value	Description
--server=	INCHARGE-SA	The name of the Global Manager from which the adapter receives its notification list and notifications.
--subscribe=	N/A	Name of the notification list received from the Global Manager.
-DNotifyAction=	N/A	Name of the tool as specified in the Global Manager Administration Console. This is the tool to invoke when a notification is active. This argument is optional.
-DClearAction=	N/A	Name of the tool as specified in the Global Manager Administration Console. This is the tool to invoke when the notification clears. This argument is optional.
-DLogActions=	FALSE	Specifies whether the adapter should log each action to *stdout*. This argument is optional.
--output	N/A	Redirects the output from the adapter to a log file in BASEDIR/smarts/local/logs. The --DLogActions argument must be true to enable output.
rules/ics/auto-action.asl	N/A	Script required to perform the automated action.

The following example shows the arguments for the automatic action adapter that invokes the “Open Trouble Ticket” script when a notification becomes active (is notified) and invokes the “Close Trouble Ticket” tool when the notification becomes inactive (clears):

```
▼ BASEDIR/smarts/bin/sm_adapter--server=INCHARGE-SA \
          --subscribe=TicketingNL/n  \
          -DNotifyAction="Open Trouble Ticket"  \
          -DClearAction="Close Trouble Ticket"  \
          -DLogActions=TRUE  \
          --output  \
          ../rules/ics/auto-action.asl ▲
```

NOTICE

▼▲ The command must be typed as one line.

Running tools over X Windows

If necessary, you can configure a server tool to invoke an X Windows application. The console user that invokes such an action must run an X server in order to display the X window. When the Global Console attaches to the Global Manager, it passes the user's display information to the Global Manager.

The Global Console determines a user's display by using the following rules:

- If the environment variable SM_DISPLAY is set, its value is passed to the Global Manager.
- If the environment variable DISPLAY is set, its value is passed to the Global Manager.
- The value is set to *‹host›*:0.0 where *‹host›* is the name of the user's system.

Because X Windows applications are typically long running, take special care when writing the tool script to avoid unwanted interactions with the tool timeout. You should write the tool script so that the X Window application runs in the background. This allows the tool script to exit and the Global Manager to stop the timeout. If the tool's script runs in the foreground, the Global Manager terminates the process when the timeout expires.

The **sm_xcmd** utility is provided by EMC Smarts for executing an X Windows application in the background. The utility performs a basic connectivity test of the user's X Windows display. The ics-telnet.sh server tool is an example of a tool script that uses sm_xcmd to run an application over X Windows.

Adding domain configurations

Domains are configured using the graphical interface accessed from the Global Manager Administration Console. The domain configuration information is saved to repository files. The repository files include information for the following predefined domains:

- INCHARGE-AM
- INCHARGE-PM
- INCHARGE-AM-PM
- INCHARGE-AM2
- INCHARGE-OI
- NCM
- INCHARGE-DISCMGR
- INCHARGE-MPLS-ANALYSIS
- INCHARGE-MPLS-MONITORING
- INCHARGE-MPLS-TOPOLOGY

NOTICE

Only the INCHARGE-AM domain is enabled. You must enable other domains or create new ones as necessary.

If your domain configurations are not predefined in the Service Assurance Manager server, you can use the following tools to add new configurations to the Service Assurance Manager server:

- Import XML Domain Configurations
- Import XML Configurations

Chapter 5, "Configuring Domains,"provides additional information on the ics-default.xml file.

Importing XML Domain Configurations

To add new configurations using the XML domain configuration tool:

1. Open the **Topology Browser Console**.
2. Right-click the instance of the local InChargeDomain representing the Service Assurance Manager server.
3. Select **Server Tools** › **Import XML Domain Configuration**.

   The **Import XML Domain Configuration** dialog box appears, as illustrated in Figure 11 on page 157.

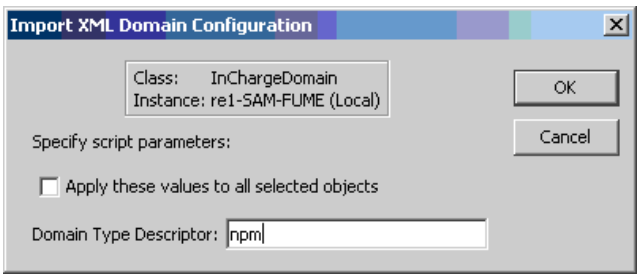

Figure 11 Import XML Domain Configuration dialog box

4. Type a domain type descriptor in the **Domain Type Descriptor** field.

   Table 31 on page 157 defines domain type descriptors for all available domain configuration files.

Table 31 Available domain type descriptors (page 1 of 2)

Domain type descriptor	Filename	Description
acm	ics-acm-default.xml	Application Connectivity Manager post 1.0
acm-1.0	ics-acm-1.0-default.xml	Application Connectivity Manager 1.0
ai	ics-ai-default.xml	Application Insight
app-poller	ics-app-poller-default.xml	Application Poller
asm	ics-asm-default.xml	Application Services Manager
atm	ics-atm-default.xml	ATM-Frame Relay Manager
bmc	cs-bmc-default.xml	--

Table 31 Available domain type descriptors (page 2 of 2)

Domain type descriptor	Filename	Description
ciena	ics-ciena-default.xml	Ciena TMF14 Adapter
ctm	ics-ctm-default.xml	Cisco TMF814 Adapter
dfm	ics-dfm-default.xml	Device Fault Manager
mcast	ics-mcast-default.xml	Multicast Manager
mpls	ics-mpls-default.xml	MPLS Manager
mpls1.3	ics-mpls1.3-default.xml	MPLS Manager 1.3 and earlier
npm	ics-npm-default.xml	Network Protocol Manager
oi	ics-oi-default.xml	Open Integration Server
sam	ics-sam-default.xml	Hierarchical SAM
sdi	ics-sdi-default.xml	SQL Data Interface
sia	ics-sia-default.xml	Storage Insight
sia-group	ics-sia-group-default.xml	Storage Insight using the new domain group mechanism
sim	ics-sim-default.xml	Security Infrastructure Manager
sonet	ics-sonet-default.xml	--
vhm	ics-vhm-default.xml	Voice Health Manager
voip	ics-voip-default.xml	Voice over IP Manager
wdm	ics-wdm-default.xml	--
xdatmfrip	ics-xdatmfrip-default.xml	Cross Domain ATM-FR to IP
xdotmip	ics-xdotmip-default.xml	Cross Domain OTM to IP

5. Click the **Apply these values to all selected objects** option to have the domain values applied to selected objects.
6. Click **OK**.

NOTICE

The domain configuration is imported in a disabled state. To begin synchronization with Service Assurance Manager, enable the domain and click the **Launch Server Reconfiguration** toolbar button located on the Global Manager Administration Console toolbar menu.

NOTICE

If the SAM server is started on a non-default broker, importing an XML Domain configuration throws an error. This issue can be avoided by updating the value of the non-default broker under **SM_BROKER_DEFAULT** in the **runcmd_env.sh** file.

Importing XML configurations

To add new configurations using the generic configuration tool:

1. Open the **Topology Browser Console**.
2. Right-click the instance of the local InChargeDomain representing the Service Assurance Manager server.
3. Select **Server Tools** › **Import XML Configuration**. The **Import XML Configuration** dialog box appears, as illustrated in Figure 12 on page 159.

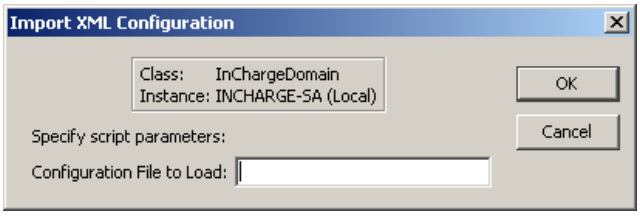

Figure 12 Import XML Configuration dialog box

4. Type the name of the XML file you want to load in the **Configuration File to Load** field.

   NOTICE

   The XML being added must reside in the SITEMOD/conf/ics directory.

5. Click **OK**.

CHAPTER 9
Escalation Configuration for the Global Manager

This chapter contains the following information:

- Overview of escalation 162
- How notifications are scheduled for escalation 165
- Developing escalation policies 166
- Modifying escalation policies and paths 169
- Removing or deleting escalation paths and policies 175
- Testing and troubleshooting escalation policies 176

Overview of escalation

Escalation provides a method for automatically responding to a notification:

- As soon as the notification is received by the Global Manager.
- When the notification satisfies certain conditions.

Automatic escalation can enhance your problem resolution process by:

- Dispatching maintenance staff with a description of hardware failures and the specific parts they will need.
- Restoring service by restarting applications, initiating failover processes, or rerouting services around failed components.
- Notifying customers of service outages and providing status updates.
- Alerting successive layers of management of unresolved problems to ensure that they receive the proper level of attention.

Escalation helps prevent important notifications from being overlooked or ignored and can immediately start the process of remedying the situation that caused the notification.

For example, you can create an escalation policy that is activated when a specific router goes down. When this notification occurs, the Global Manager matches it to the appropriate escalation policy and path. The path then begins invoking tools such as opening a trouble ticket and paging technicians.

A notification continues to escalate while it matches the filter criteria of the path. As time progresses, the notification may escalate to different levels in the path based on the duration, and the path may invoke additional tools to respond to the notification. If the status of the notification changes in such a way so that it no longer matches the path's filter criteria, the notification no longer escalates to the next path level.

For example, if a Router Down notification matches against the path filter for 15 minutes, the notification is escalated to the next level. At this level, the escalation path could invoke a tool that sends email to a manager. But, if a technician acknowledges the notification after 10 minutes, and this acknowledgement causes the notification to no longer match the path filter criteria, the notification does not escalate to the next level.

Escalation policy structure

An escalation policy is a set of related escalation paths. An escalation policy includes a policy filter and one or more escalation paths, as shown in Figure 13 on page 163. Notifications that match the policy filter are passed to the escalation paths. If a notification does not match the policy filter, it is ignored. If a policy filter is not defined, all notifications are passed to the escalation paths.

An Escalation Path also includes a filter and up to six escalation levels. A notification must match both the policy filter and the path filter to begin at the first escalation level, Level 0.

Each escalation level includes one or more automated tools to invoke and a time duration, after which a notification advances to the next level. An escalation level cannot have a duration of less than 1 minute. You can, if necessary, create an escalation level that does not invoke a tool.

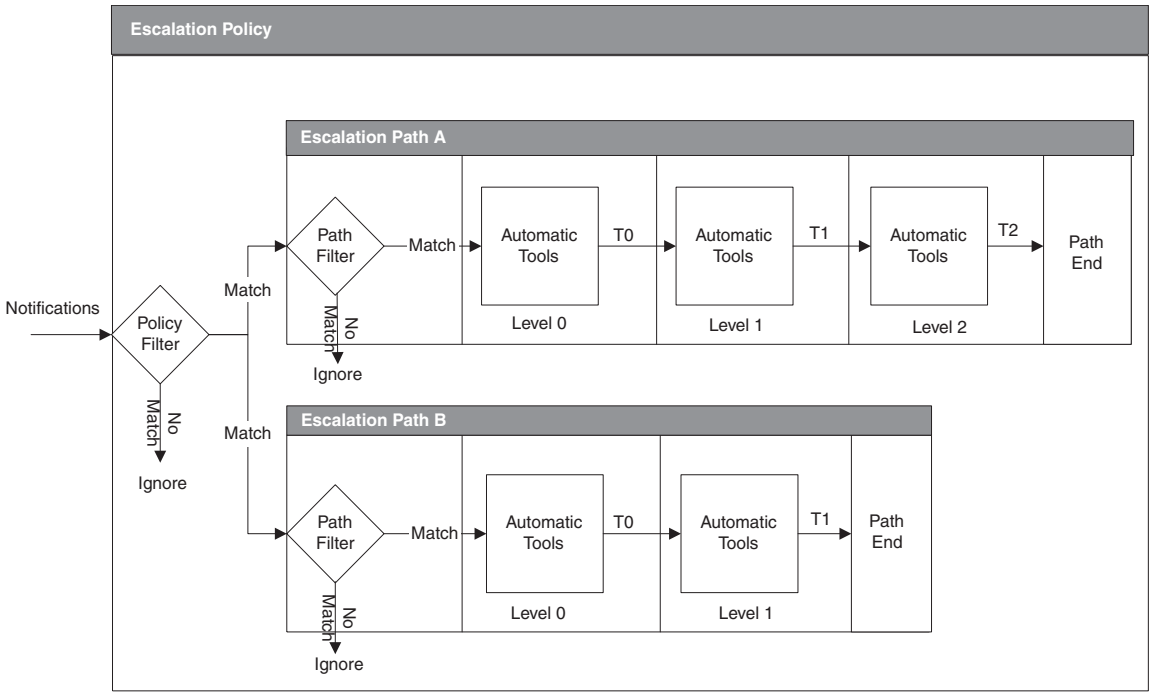

Figure 13 Escalation policy structure

When a notification matches both the policy and path filters, it starts at escalation level 0. When a notification reaches level 0, the tools associated with level 0 are invoked immediately. The notification remains at Level 0 for the specified duration, shown as T0 in Figure 13 on page 163, before escalating to Level 1. At this point, all Level 1 tools are invoked and, after the duration of T1, the notification escalates to Level 2. This process repeats through each level defined for the path. If the notification changes during escalation and no longer matches the path and policy filters, it leaves the escalation path without invoking any additional tools.

Viewing an escalation policy

You create, configure, and monitor escalation policies with the Global Manager Administration Console. Expanding Escalation Policies in the administration tree lists escalation policies. Selecting a policy in the administration tree displays the Escalation Policy view, as shown in Figure 14 on page 164.

Policy names in the administration tree are black when any escalation path in the policy is enabled. When all paths in a policy are disabled, the policy name is dimmed.

The Configure Escalation Policy view allows you to create, modify, and delete escalation policies, paths, and levels. In addition, you can monitor the status of an escalation policy by right-clicking on an escalation path and selecting Show Notifications from the menu.

Escalation paths are displayed as horizontal bars. The name on the left of the escalation path is the pathname. Within each bar are shaded blocks, each of which represents an escalation level within the path. The duration and automated tools are listed within each escalation level of the path.

An enabled escalation path is displayed with each level a different color. A disabled path is dimmed. All the escalation paths shown in Figure 14 on page 164 are enabled.

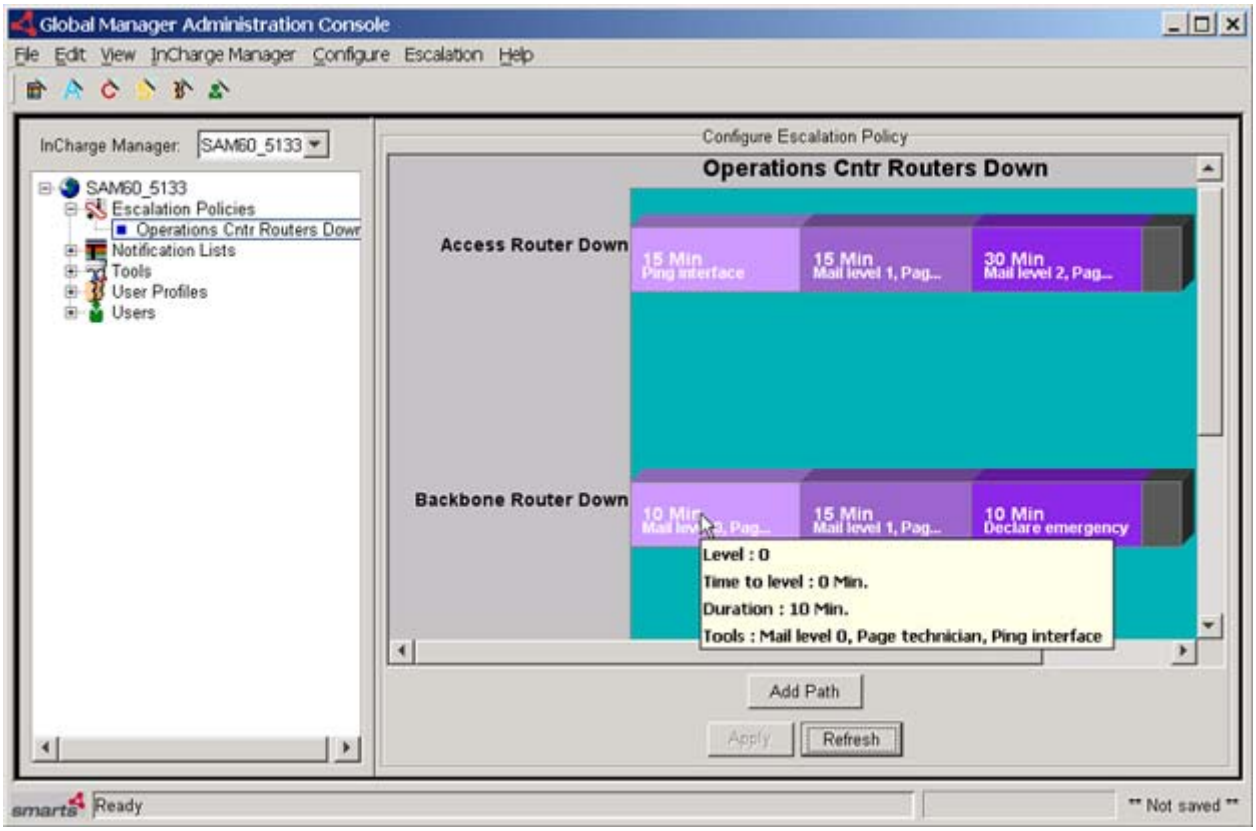

Figure 14 Escalation policy view

When you hold the cursor over an escalation level, a tooltip window displays:

- Level: Indicates the level (0–5) within the path.
- Time to level: Indicates the amount of time it takes a notification to reach this level within the path.
- Duration: Indicates the amount of time a notification spends at this level before advancing to the next level. However, Duration does not include the time it takes to invoke any tools. When a level is reached, actions are executed first. When all actions have finished executing, the notification is scheduled to enter the next level. The notification will enter the next level after the amount of time set by Duration.
- Tools: Lists the automated tools that are invoked at this level.

At the bottom of the Configure Escalation Policy window are three buttons:

- Add Path — Adds a new escalation path to the displayed policy. This is equivalent to selecting Escalation › Path › New.
- Apply — Saves any changes made to the escalation configuration to the Global Manager. If you have unsaved changes, you are prompted to save before leaving the view.

- Refresh — Retrieves the configuration of current escalation policy from the Global Manager. This is important if another administrator might be modifying the policy that you are currently viewing. The Global Manager prevents two administrators from overwriting each other's changes without first viewing the changes. For example, if you make changes to an escalation policy and apply them while another administrator is also making changes, the other administrator will not be able to apply the changes. The Global Administration Manager Console prompts the other administrator to refresh their policy view.

Refresh can also be used to cancel changes that have not been applied.

How notifications are scheduled for escalation

When the Global Manager receives or creates a notification, it compares the notification against the filter for each policy and path. If the notification matches a policy and path filter, the Global Manager schedules the notification for escalation. When a level is reached, actions are executed first. When all actions have finished executing, the notification is scheduled to enter the next level. The notification will enter the next level after the amount of time set by the duration of the level.

The Last Notify time, or the LastNotifiedAt notification attribute, also determines whether a notification enters into an escalation policy. The Last Notify Time must be later than the time at which the path was enabled in order for the notification to escalate.

If a Global Manager is stopped and restarted, the Global Manager restores the state of escalated notifications. Notifications which are scheduled to escalate to a new level before the Global Manager stops are rescheduled based on the current time when the Global Manager is restarted.

The rescheduled escalation time accounts for the time that the Global Manager was not running. A rescheduled notification, however, will never escalate more than one level as a result of the time the Global Manager was not running.

Effect of Global Manager restart on escalation

The following examples illustrate the effect of a Global Manager restart on escalating notifications.

Example 1

1. A notification enters escalation level 0 at 1 P.M. The duration of the level is 1 hour.
2. At 1:10 P.M. the Global Manager is shut down.
3. At 1:25 P.M. the Global Manager is restarted.

The notification was scheduled to escalate to level 1 at 2 P.M. When the Global Manager restarts, this notification is rescheduled to escalate at 2 P.M.

Example 2

1. A notification enters escalation level 1 at 2 P.M. The duration of level 1 is 10 minutes and the duration of level 2 is 10 minutes.
2. The Global Manager is shut down at 2:05 P.M. and restarted 20 minutes later at 2:25 P.M.

Prior to the Global Manager shutting down at 2:05 P.M., the notification would have escalated to level 2 at 2:10 P.M. and was scheduled to escalate to level 3 at 2:20 P.M. (the duration of level 2 is 10 minutes).

Since the notification was in level 1 when the Global Manager was shutdown, it needs to be rescheduled for the next level (level 2) when the Global Manager is restarted. Therefore, when the Global Manager is restarted, it reschedules the notification to escalate to level 2 immediately at 2:25 P.M. Upon entering level 2 and performing actions, the notification is scheduled for level 3 at 2:35 P.M.

Developing escalation policies

The first step in creating an escalation policy is to formalize your response plans to the notifications you are likely to encounter. You may need to assign responsibilities and determine how quickly your organization is required to respond to various types of system and network failures. These plans form the template for your escalation paths and levels.

You may also need to create automated tools to initiate your response to incoming notifications. Chapter 8, “Tool Configuration for the Global Manager,” describes how to create and configure automated tools.

The process of creating an escalation policy at the Global Manager Administration Console consists of the following steps:

1. Create an escalation policy by copying an existing policy or creating a new policy.
2. Add one or more escalation paths to the policy. Again, you can copy an existing path or create a new path.
3. Add escalation levels, up to six, to each path.

Creating an escalation policy

To create an escalation policy, complete the following steps by using the Global Manager Administration Console:

1. Select **Escalation** › **Policy** › **New**. The **Create a New Escalation Policy** dialog box appears.
2. Type the name for the escalation policy.

   Create the policy by copying an existing policy or by creating a new policy. When you create a new policy, the parameters of the policy, such as filters and paths, must be created. When you copy a policy, all the properties of the existing policy are copied to the new policy. However, the name of the newly created escalation policy always appears dimmed in the administration tree even if the original policy name appears bright.

3. Click **Edit Filter** to create or modify the filter that is matched against notifications to determine whether they qualify for the policy. The **Filtering New Policy** text dialog box appears.

   You can specify an expression filter, an ASL filter, or both. When you combine filters, by adding filter sheets, a notification must match at least one of the filter sheets to pass the filter. Chapter 10, "Working with Filters," includes additional information about the **Filtering New Policy** dialog box.

4. When you finish creating filters, click **OK** to close the **Filtering New Policy** dialog box.

5. Click **OK** to close the **Create a new Escalation policy** dialog box. The name of the escalation policy appears in the administration tree and at the top of the **Configure Escalation Policy** window.

When using ASL filter under **Edit Filter,** the escalation policy is created only if the relative path to the ASL file is provided as ics/‹asl file name›. By default, Global Manager searches for the ASL file under local/rules and the policy is not created even though the ASL file exists under BAEDIR/rules/ics. Also, the escalation policy is not created if the absolute path to the ASL file is specified, which is BAEDIR/rules/ics/‹asl file name›.

Creating an escalation path

You can create an unlimited number of paths within a policy, but it is usually easier to organize and manage smaller groups of paths in more policies.

To create an escalation path:

1. Select the escalation policy in the administration tree.

   The **Configure Escalation Policy** panel appears.

2. Create an escalation path by adding a new path or copying an existing path.

   - To create a new blank path, select **Escalation › Path › New** or click **Add Path** at the bottom of the **Escalation Policy View** panel.

     The **Create a new Escalation Path** dialog box appears.

   - To copy a path, select the path you wish to copy and select **Escalation › Path › Copy**.

     The **Copy Escalation Path** dialog box appears.

   NOTICE

   You can only copy a path from the current policy.

3. Type the name of the escalation path. If you are copying the path, modify it to suit your needs. "Modifying escalation policies and paths" on page 169 provides additional information.

4. Click **Edit Filter** to create or modify the path filter.

   The **Filtering New Path** dialog box appears.

The filter determines what notifications follow this path of the policy. Similar to the policy filter, you can specify an expression filter, an ASL filter, or both. Chapter 10, "Working with Filters," includes information on setting up a filter.

NOTICE

For most escalation paths, you should add a filter that matches active notifications to ensure that you only escalate active notifications.

5. When you finish creating filters, click **OK** to close the **Filtering New Path** dialog box.
6. Click **OK** to close the **Create a new Escalation Path** dialog box.

   Your path is displayed in the **Configure Escalation Policy** panel of the Global Manager Administration Console.
7. Repeat this procedure for each escalation path.

The next step is to add one or more levels to each escalation path.

Adding escalation levels

NOTICE

An escalation path requires at least one level. You can add up to six levels to each path.

To add escalation levels:

1. Right-click the path and select **Level** › **Insert** or select the path and choose **Escalation** › **Level** › **Insert**. The **Insert Level** dialog box appears.

   If the escalation path already contains a level, you can control where the new level is inserted:

   - To insert a new level at the end of the path, select the box at the end of the path to insert the level.
   - To insert a level before an existing level, select the level that you want to insert the new level in front of to insert the level.
2. Specify the duration of the level by typing a number and selecting the time unit from the list box.
3. Select automated tools from the **Available Tools** list and move them to the Invoked Tools list. Tools are invoked immediately after a notification reaches this level. Chapter 8, "Tool Configuration for the Global Manager," includes information on creating tools.
4. Click **OK** to close the **Insert Level** dialog box.

   The new level appears in the escalation path.
5. Repeat these steps for each level in the escalation path.
6. When the path is complete, click **Apply** to save the changes to the Global Manager.
7. Repeat this procedure for each escalation path in the escalation policy.

   After you finish an escalation policy, you can enable one or more of its paths.

Modifying escalation policies and paths

The types of modifications you can make to an escalation path are determined by whether the escalation path is enabled or disabled. For example, when:

- A path is enabled, you can edit the duration and tools for any escalation level in the path. You cannot add or remove escalation levels from the path or change the policy or path filters. However, you can change the script when asl script is used as a filter.
- A path is disabled, you can edit the escalation path filter and change the duration or invoked tools at any escalation level. In addition, you can add or remove escalation levels.
- All paths in a policy are disabled, you can edit the escalation policy filter.

Enabling and disabling escalation paths

Escalation policies cannot be enabled or disabled. Instead, you can enable or disable the escalation paths within a policy. When all of the escalation paths within a policy are disabled, the policy is effectively disabled.

Disabling an escalation path cancels the escalation of all notifications currently scheduled within the path. New notifications are not added to a disabled path. To avoid losing any notifications currently scheduled on the path, but to prevent new notifications from being scheduled, you can retire the path instead of disabling it. "Retiring an escalation path" on page 173 includes more information on retiring a path.

Disable an escalation path

To disable an escalation path:

1. Select the escalation policy in the administration tree of the **Global Manager Administration Console**. The **Configure Escalation Policy** panel appears.
2. Right-click the path and select **Path** › **Disable**. You are prompted to confirm this choice.
3. Click **Yes** to disable the path. The path is dimmed to indicate it is disabled.
4. Click **Apply**.

When all of the paths for an escalation policy are disabled, the name of the policy in the administration tree is dimmed to indicate that the policy has no active paths.

Enable an escalation path

When a path is enabled, the path begins to process any notifications with a Last Notify time that is later than when the path was enabled.

To enable an escalation path:

1. Select the policy in the administration tree.

   The **Configure Escalation Policy** panel appears in the **Global Manager Administration Console**.
2. Right-click the path and select **Path** › **Enable** from the menu.

   The path is enabled.
3. Click **Apply** to save the change to the **Global Manager**.

Modifying an enabled escalation path

When you change the tools or duration of a level in an enabled path, the following is true:

- Notifications escalated beyond this level are unaffected.
- Notifications not yet escalated to this level are subject to the new duration or tools.
- Notifications currently at the modified level are rescheduled according to the new duration or use the new tools.

The following example illustrates what happens when the duration of a level is increased:

1. A notification enters Level 1 of an escalation path at 1 P.M. The duration of the level is 20 minutes and the notification is scheduled to escalate to Level 2 at 1:20 P.M.
2. At 1:10 P.M., the duration of this level is increased to 30 minutes. As a result, the notification is rescheduled to escalate to Level 2 at 1:30 P.M.

The following example illustrates what happens when the duration of a level is decreased:

1. A notification enters Level 1 of an escalation path at 1 P.M. The duration of the level is 30 minutes and the notification is scheduled to escalate to Level 2 at 1:30 P.M.
2. The notification is scheduled to enter Level 2 at 1:30 P.M. The duration of Level 2 is 5 minutes and the notification is scheduled to enter Level 3 at 1:35 P.M.
3. At 1:20 P.M., the duration of Level 1 is decreased to 10 minutes. As a result, the notification escalates to Level 2 immediately. At Level 2, the notification is scheduled to escalate to level 3 at 1:25 P.M. because the duration of Level 2 is 5 minutes.

A change in duration will not cause a notification to escalate more than one level.

Changing the tools or duration of a level

To change the tools or duration of a level in an enabled escalation path:

1. Select the escalation policy name in the administration tree. The policy appears in the **Configure Escalation Policy** panel of the **Global Manager Administration Console**.
2. Right-click the level you want to modify and select **Level** › **Edit Properties**. The **Edit Level** dialog box appears.
3. Change the **Duration** or **Invoked Tools**:
   - Type a new value for **Duration** and select the units from the list box.
   - Change the **Invoked Tools** list:
     - Select a tool in the **Invoked Tools** list and click → (right arrow) to remove it.
     - Select a tool in the **Available Tools** list and click ← (the left arrow) to move the tool to the **Invoked Tools** list.

NOTICE

If a tool is not available, you can create the tool by clicking **Tool Wizard**. Chapter 8, "Tool Configuration for the Global Manager," includes information on creating automated tools.

4. Click **OK** to close the **Edit Level** dialog box. The changes to the level appear in the path.
5. Click **Apply** to send the change to the **Global Manager**.

Modifying disabled escalation paths or policies

When an escalation path is disabled, you can modify the path filter and add or remove levels. If you disable all of the paths in an escalation policy, you can edit the escalation policy filter.

Disabling a path cancels the escalation of any notifications along the path. If you want to modify a path or policy but do not want to cancel any scheduled notifications, you can retire the path or paths instead. This prevents new notifications from joining the path but allows scheduled notifications to continue to escalate. "Retiring an escalation path" on page 173 includes information on retiring a path.

Modifying an escalation policy filter

To modify the filter of an escalation policy, you must first disable all of the paths within the policy:

1. Select the escalation policy in the administration tree of the **Global Manager Administration Console**. The policy appears in the **Escalation Policy Configuration** panel.
2. Select **Escalation** › **Policy** › **Edit Properties**. The **Edit selected Policy** dialog box appears.
3. Click **Edit Filter**. The **Filtering Policy test** dialog box appears.
4. Change the filters as needed.

   NOTICE

   Chapter 10, "Working with Filters," contains information on changing a filter.

5. When you finish editing the filters, click **OK** to close the **Filtering Policy test** dialog box.
6. Click **OK** to close the **Edit selected Policy** dialog box.
7. Click **Apply** to send the changes to the **Global Manager**. You can now enable the paths in the policy.

Modifying an escalation path filter

To change a path filter, you must first disable the path:

1. Select the escalation policy in the administration tree of the **Global Manager Administration Console**. The policy appears in the **Escalation Policy Configuration** panel.
2. Right-click the path and select **Path** › **Edit Properties**. The **Edit selected Path** dialog box appears.
3. Click **Edit Filter**. The **Filtering Path test** dialog box appears.

4. Change the filters as needed.

   NOTICE

   Chapter 10, "Working with Filters," contains information on changing a filter.

5. When you finish editing the filters, click **OK** to close the **Filtering Path test** dialog box.
6. Click **OK** to close the **Edit selected Path** dialog box.
7. Click **Apply** to send the changes to the **Global Manager**. You can now enable the path.

Inserting escalation levels in a path

To insert escalation levels in a path:

1. For a disabled path, select the level (or Path End box) that will follow the new level. The selected level or box changes color to indicate that is selected.
2. Right-click to display the **Path** menu.
3. Select **Level** › **Insert**.

   The **Insert Level** dialog box appears.

4. Type the **Duration** and select the units from the list box.
5. Select any quantity or combination of tools from the **Available Tools** list.

   NOTICE

   If a tool is not available, create the tool by clicking **Tool Wizard**. Chapter 8, "Tool Configuration for the Global Manager," includes information on creating automated tool.

6. Click ← (left arrow) to move the selected tools to the **Invoked Tools** list.
7. Click **OK** to close the **Insert Level** dialog box. The level appears in the path.
8. Repeat steps 1 through 6 to add additional levels up to a total of six levels.

   NOTICE

   Remember that new levels are always inserted in front of the selected level.

9. When your path is complete, click **Apply** to send the changes to the **Global Manager**.

   You can now enable the path. "Enable an escalation path" on page 169 provides detailed information on enabling paths.

Removing escalation levels from a path

To remove escalation levels from a path:

1. For a disabled path, select the level to remove. The selected level changes color to indicate that it is selected.
2. Right-click to display the **Path** menu.
3. Select **Level** › **Remove**. The level disappears from the path.
4. Repeat step 1 through 3 to remove additional levels.

   NOTICE

   A valid path must have at least one level.

5. When your path is complete, click **Apply** to send the changes to the **Global Manager**.

   You can now enable the path. "Enable an escalation path" on page 169 provides detailed information on enabling paths.

Modifying tools or duration at an escalation level

This procedure is identical to the one performed when the path is enabled. "Changing the tools or duration of a level" on page 170 includes information on modifying tools or duration at an escalation level.

Retiring an escalation path

To avoid losing escalations, consider retiring the path. Retiring an escalation path allows all notifications with a Last Notify time earlier than the retirement time to enter or continue escalation through the path. The path will not escalate any notifications with later Last Notify times.

To retire an escalation path:

1. Select the escalation policy in the administration tree of the **Global Manager Administration Console**. The policy appears in the **Escalation Policy Configuration** panel.
2. Select the appropriate path by clicking a level in the path. The level changes color indicating it is selected.
3. From the **Escalation** menu, select **Path** › **Retire**. The **Confirm Disable** dialog box appears.
4. Click **Yes** to retire the path. All levels in the path change color (to one shade of blue) indicating the path is retired.
5. Click **Apply** to send the change to the **Global Manager**.
6. After checking the retired path to determine whether any notifications are being escalated, disable or remove the path.

Modifying escalation paths by using the Retire option

You may need to modify a path to add escalation levels or edit the filter. These changes require that the path is disabled. Disabling the path immediately terminates the escalation of any scheduled notifications.

The Retire option enables you to avoid this issue and provides a smooth transition to a new path. When a path is retired, notifications that are currently being escalated through the path continue to follow the escalation path. In addition, existing notifications that now meet the filtering criteria for the retired path will enter the path (from level 0). However, new notifications that meet the filtering criteria for the retired path are not added to the path.

To retire an existing path and replace it with a modified path:

1. Copy the enabled path you want to change.

   NOTICE

   The copied path will be disabled.

2. Modify the path you copied.
3. Enable the new path, but do not apply the changes.
4. Retire the existing path. "Retiring an escalation path" on page 173 includes information for retiring an escalation path.
5. Click **Apply** to simultaneously enable the new path and retire the original path. Notifications on the retired path continue to escalate while any new notifications are handled by the new path.
6. Remove the retired path when all notifications escalate through the entire path.

The new path only processes notifications that are notified after the enable time of the path. The existing path, because it is retired, processes any notifications with a notify time less than the time that the path was retired.

Removing or deleting escalation paths and policies

Enabled paths and policies can be removed or deleted. This terminates all scheduled escalations for the removed path or for all paths in a deleted policy.

Removing an escalation path

A path can be removed when it is enabled, disabled, or retired.

To remove an escalation path:

1. Select the policy name in the administration tree. The policy appears in the **Escalation Policy Configuration** panel.
2. Select the path.
3. From the **Escalation** menu, select **Path** › **Remove**. The path is removed from the **Escalation Policy Configuration** panel.
4. Click **Apply** to send the change to the Global Manager.

Deleting an escalation policy

A policy can be removed with enabled, disabled, or retired paths. When an escalation policy is deleted, all escalation paths within the policy are deleted. All currently scheduled escalations will be unscheduled, and the policy will be removed from the system and no longer process any notifications.

To delete an escalation policy:

1. Select the policy name in the administration tree. The policy appears in the **Escalation Policy Configuration** panel.
2. From the **Escalation** menu, select **Policy** › **Delete**. The **Confirm Delete** dialog box appears.
3. Click **Yes** to confirm the policy deletion. The policy name is removed from the administration tree in the Administration Console.

Testing and troubleshooting escalation policies

As escalation policies usually handle important notifications, test your escalation paths to ensure they function appropriately.

EMC Smarts software allows you to quickly test and verify escalation paths through the use of **sm_ems**, the Audit log, and the automated tools you intend to invoke.

After configuring and enabling escalation policy and paths, use **sm_ems** with the notify command option to issue a sample of the specific notification that you hope to escalate. The *EMC Smarts Service Assurance Manager Adapter Platform User Guide* describes the **sm_ems** utility.

Once your sample notification appears in Notification Log Console, double-click the notification and select the Audit Log tab. Check the log for escalation-related entries in Table 32 on page 176. Escalation-related Audit Log entries always have the userid of SYSTEM.

Finally, ensure your automated tools are invoked and function as intended.

Table 32 Escalation-related entries in the Audit Log

Type	Audit Log description	Description
Action Completed	‹tool name›	The named automated tool was invoked and completed.
Action Failed	‹tool name›	The named automated tool was invoked, but failed.
Action Invoked	‹tool name›	The named automated tool was invoked.
ESCALATION	MATCHED:‹policy name/path name›	Notification has characteristics that match the filter criteria for the specified escalation policy and path.
ESCALATION	REACHED:‹policy name›/‹path name›, ‹level number›	Notification has reached the indicated escalation level in the specified path.
ESCALATION	SCHEDULED:‹policy name›/‹path name› for ‹level number› at ‹date› ‹time›	Notification is scheduled to reach the next escalation level at the specified date and time.
ESCALATION	TERMINATED ESCALATION: Unmatched ‹policy name›/‹path name›	A change in the notification means it no longer matches the filter criteria for the escalation policy and path and is removed from the policy.
ESCALATION	TERMINATED ESCALATION: Disabled ‹policy name›/‹path name›	The escalation policy or path was disabled and notifications are no longer escalated within the policy or path.
ESCALATION	TERMINATED ESCALATION: Notification Archived, ‹policy name›/‹path name›	The notification was archived after it entered the escalation policy and was removed from the policy.
ESCALATION	RESCHEDULED: ‹policy name›/‹path name› from ‹level› due at ‹datestamp› ‹timestamp›	An action within a level was rescheduled to a new date or time.
ESCALATION	END OF PATH: ‹policy name›/‹path name›, ‹level number›	Notification has gone through all escalation levels and reached the end of the named escalation path.

Viewing notifications being escalated

Once an escalation path is enabled, you can monitor any notifications that escalate through the path. You can select an escalation level within a path and view the notifications which are currently escalated to the level. You can then select any of the notifications to see the notification properties.

To view notifications being escalated:

1. Select the policy name in the administration tree. The policy appears in the **Escalation Policy Configuration** panel.
2. Select the appropriate path by clicking a level in the path. The level changes color.
3. From the **Escalation** menu, select **Level › Show Notifications**. The **Escalated Notifications** dialog box displays any escalated notifications currently at the level.
4. Double-click an escalated notification to display its notification properties.

   NOTICE

   Notification data displayed is static. Changes are not automatically updated in the view. Click **Refresh** to update the view.

5. Click **Close** to close the **Escalated Notifications** dialog box.

CHAPTER 10
Working with Filters

This chapter contains the following information:

- Overview 180
- Building expression filters 180
- ASL filters 186

Overview

A filter screens incoming notifications by comparing a set of defined criteria against the attributes of a notification or ICIM class. Values that match the criteria pass through the filter. Filters are used to configure:

- Notification lists — "Notification lists" on page 52 provides additional information.
- Context and Status Criteria of client and server tools — "Context and Status Criteria for client and server tools" on page 151 provides additional information.
- Escalation paths and policies — "Developing escalation policies" on page 166 provides additional information.

You can define two types of filters: an expression filter or an ASL filter. You create an expression filter using the Filter Builder, which is described in the next section. An ASL filter, specified in the ASL Filter checkbox of the Filter Builder, is defined in a separate ASL file. "ASL filters" on page 186 provides more information about creating ASL filters.

Building expression filters

An expression filter uses matching criteria to filter against the value of an attribute of the specified class. Matching criteria comprises a series of characters, and optionally, wildcards:

- "Attributes for matching notification properties" on page 215 includes a complete list of notification attributes.
- "Wildcard Patterns" on page 219 provides a description of wildcards.

Table 33 on page 180 describes the parameters of an expression filter.

Table 33 Expression filter fields

Parameter	Description
Attribute	Name of the attribute. The expression filter matches against the value of this attribute.
matches	Matching pattern that is compared against the value of an attribute: • For Boolean attributes, a menu lists the values. • For attributes whose values are known, the Helper button lists potential values, taken from the notifications in the Global Manager. You can use wildcards within the value field.
AND (green plus sign)	Add more attributes to the filter sheet to match against, creating a more narrow filter. A notification must match all of the attributes on a sheet to pass the filter.
OR (Sheet Tab)	Add more filter sheets to filter. A notification must match at least one of the filter sheets to pass the filter. You can add sheets by clicking Add Sheet or selecting Edit › Add Sheet.

Layout of the Filter Builder

The Filter Builder interface differs for notification filters, context and status criteria, and escalation policies.

Filter Builder for notification lists

The Filter Builder for notification lists opens with a blank “Sheet:1” tab. The sheet contains a Use ASL Script checkbox and a green plus sign, which is used to add notification attributes to the sheet. For notification lists, you can specify an ASL filter, an expression filter, or both. If you use a sheet to specify an ASL filter, you cannot specify an expression filter on that same sheet.

Menus for each filter criterion enable you to select a notification attribute and one of the following conditions, if applicable:

- A match condition such as a specific value or expression. You can also specify wildcard characters in a match condition. Values are case-sensitive.

  NOTICE

  Values entered into the notification list filter from the Global Console are not case-sensitive.

- The is child of condition (the IsA filter option) for notification attributes that have a class name for their value. “IsA Filter option for notification lists and logs” on page 182 provides more information on this option.

To treat a wildcard character as a text character, insert a backslash (\) before the wildcard character. For example, to filter on all event names that include a bracket, specify “*\]*”. (Do not include the quotation marks in the expression.)

Filter Builder for Context and Status Criteria

The Filter Builder for specifying Context Criteria and Status Criteria opens with a blank “Sheet:1” tab. At the top, the sheet contains a Use ASL Script checkbox and a list box that lists Context objects or Status objects. For expression filters, you select a class from the object menu and then click the plus sign to add attributes that your expression matches against. When using the Filter Builder for Context and Status Criteria, you can use either an expression filter or an ASL filter, but not both. In addition, you cannot add more sheets.

Filter Builder for escalation policies

The Filter Builder is used to specify policy filters and path filters. In both cases, you are matching against notification attributes. The Filter Builder dialog box provides an Edit menu from which you can use to add, copy, or delete sheets from your filter. Select the Use ASL Script checkbox to specify an ASL filter or click the plus sign (+) to create an expression filter.

At the top of the dialog box, the Edit menu enables you to add, copy, or delete a Sheet tab. On the left, the plus sign (+) button enables you to add one filter criterion at a time. Menus for each filter criterion enable you to select a notification attribute and a match condition such as a specific value or expression. The IsA Filter option is available when supported on the server. You can also specify wildcard characters in a match condition. Values are case-sensitive.

IsA Filter option for notification lists and logs

NOTICE

The IsA Filter option is available for filtering the notification log and summary views on clients whether or not the Global Manager supports it, because the filtering is performed by the client. The option is only available for Notification List and Escalation filtering on the server if the server supports it; this is because the filtering is on the server.

The IsA filter option, displayed as is child of, only appears for notification attributes that have a class name for their value, including:

- Class
- ClassName
- ElementClass

To create an IsA filter for one of these notification attributes:

1. Select the class and the is child of option from the pull-down menus.
2. Click **OK** to apply your filter and exit the dialog box.

   The **Notification Log Console** then displays only those notifications for instances of the specified class and its subclasses.

NOTICE

Only a single class name, without any wildcard characters of any kind, may be specified for an IsA filter.

For example, if IsA filter is used and based on the RedundancyGroup class, the notification log would display only all notifications related to instances of the following:

- WebServerRedundancyGroup
- DatabaseRedundancyGroup
- OracleRAC
- ApplicationServerRedundancyGroup
- WebSphereCluster

Using the Filter Builder

An expression filter comprises one or more sheets. Each sheet specifies an ASL filter or is composed of one or more attribute/value pairs. Each pair includes the attribute name and an expression to match against that attribute's value. To match a given sheet, a notification or topological element must match every attribute/value pair specified in the sheet.

You can specify multiple sheets. A notification or topological element passes the filter if it matches at least one sheet.

Using the Value Helper

When defining the expression for a filter, you can use the Helper button to open a dialog box that enables you to add values to an expression. The type of help provided depends on the value type of the attribute for which you are defining an expression. For example, the Value Expression Helper for the Certainty notification attribute allows you to specify a range. An expression for a string field can include wildcard patterns as well as characters. For example, the Element Class helper displays a list of known Element Classes. You can select multiple names and the helper dialog box automatically creates an OR expression with the selected values.

You can also use wildcards in the Value field. Type one or more characters in the Value field and specify a wildcard. After you type the Value field, the Value Helper is disabled. "Wildcard Patterns" on page 219 provides additional information on wildcards.

Figure 15 on page 183 illustrates the Value Helper dialog box with two Element Class elements selected.

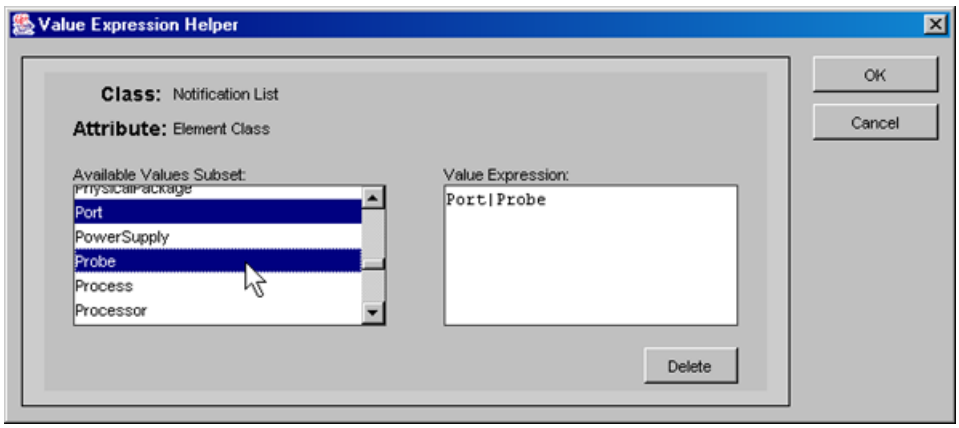

Figure 15 Filter Value Helper dialog box

After clicking OK, the dialog box is closed and the expression is listed to the right of the attribute in the filter builder where you can manually edit it.

Filter examples

Figure 16 on page 184 illustrates different expression filters. The filter in the first example matches all notifications with a severity of 1 or 2.

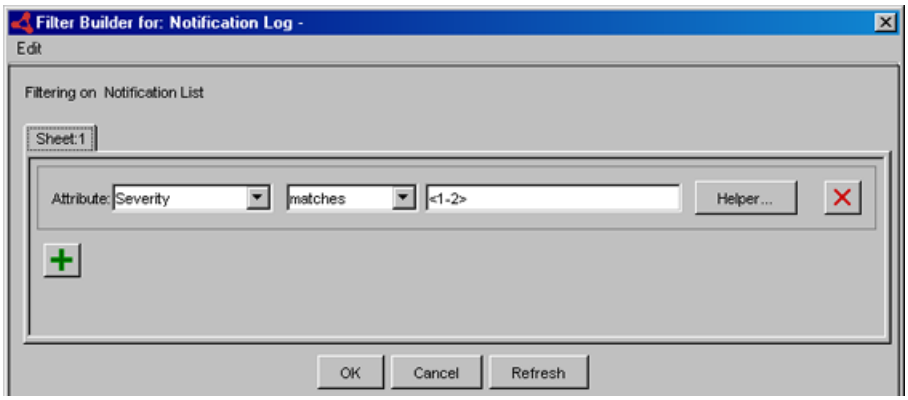

Figure 16 Filter notifications with severity of 1or 2

The filter in Figure 17 on page 184 matches all notifications with a value of Router or Switch in the Element Class attribute.

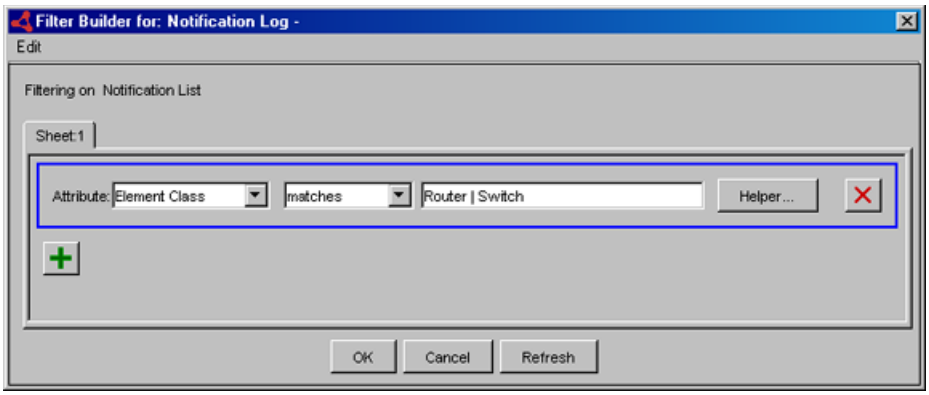

Figure 17 Filter notifications of element class router

The filter inFigure 18 on page 185 uses an implicit “or” condition by adding a second sheet. When you specify multiple sets of criteria as shown in the example, a notification passes the filter if it matches all of the conditions in one sheet. In this example, a notification passes the filter if it has a severity of one or two OR an impact between 5 and 100.

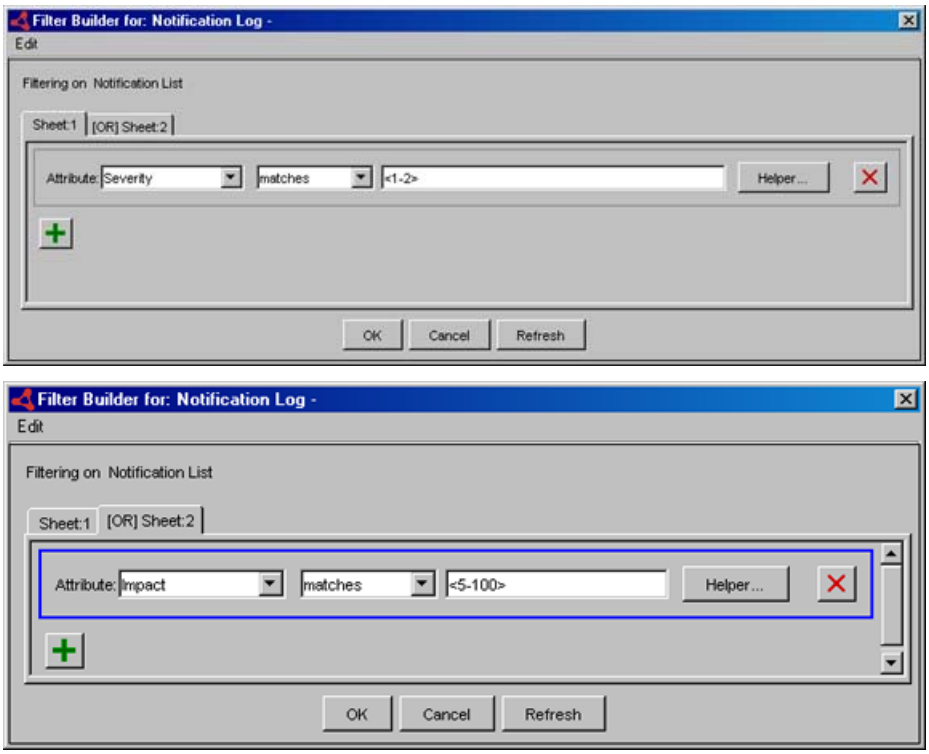

Figure 18 Filter notifications with severity of 1-2 or an impact of 5 or more

The filter example in Figure 19 on page 185 matches all notifications that have an owner assigned to them unless that owner is SYSTEM or maint.

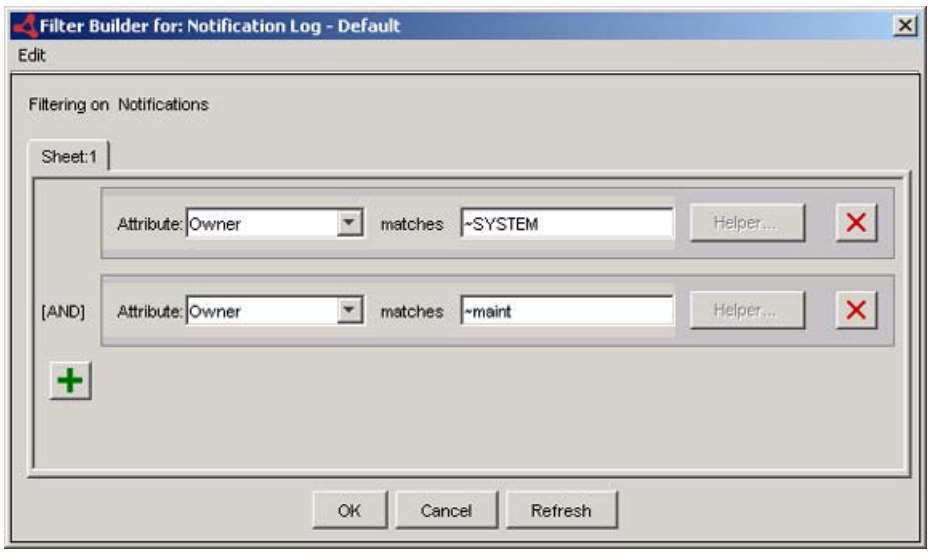

Figure 19 Filter showing notifications owned by all users other than “SYSTEM” or “maint”

The filter example in Figure 19 on page 172 matches all notifications for routers with a value of 1 in the severity field.

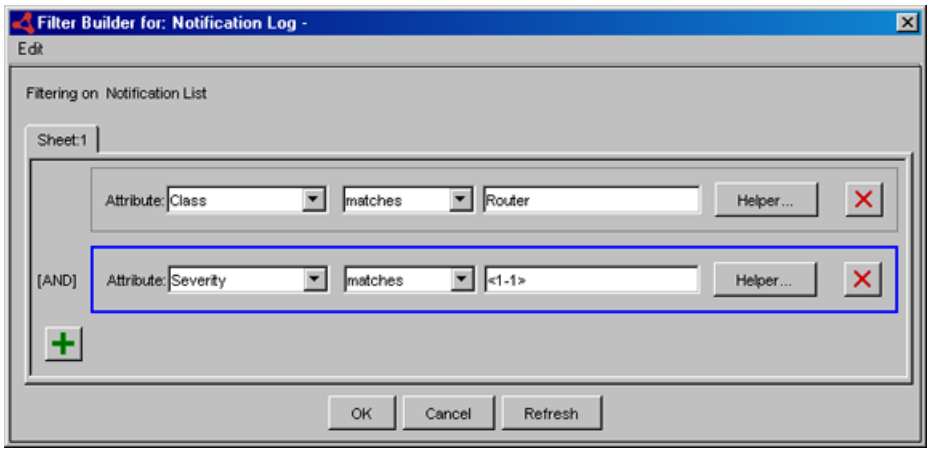

Figure 20 Filter showing routers with severity 1

ASL filters

If an expression filter does not provide enough flexibility to define filtering criteria, you can use an ASL filter. An ASL filter is an ASL program that processes each notification or other type of object.

The *EMC Smarts ASL Reference Guide* provides detailed information on ASL expressions.

An example ASL filter script, nl-sample-filter.asl, is included in the *BASEDIR*/smarts/rules/ics directory.

Using ASL filter to load Notif remotely

To reload Notif remotely using ASL, invoke the clearConfigCache method on singletonNotifProcessorInterface instance of notifProcessorInterface(class) on the appropriate SAM server.

Here is an example of the command:

```
$./dmctl -b localhost:8822 -s INCHARGE-OI invoke
   notifProcessorInterface::singletonNotifProcessorInterface
   clearConfigCache
```

where localhost:8822 is the port on which the Broker is running and INCHARGE-OI is the name of the server.

This command, when successfully executed, will reload the Notif processor with all the .ncf files available in all the SITEMOD locations, and display **TRUE** as output on the CLI.

Notification lists and escalation policies

The following example illustrates the format of an ASL filter used to filter Notification List and Escalation Policies. It filters for notifications with an EventName attribute “Failure.”

The name of the notification object is assigned to the variable notification. If the value of EventName is Failure, the value of Result is set to TRUE and the notification passes the filter, as illustrated in the following code sample:

```
START do {
      notification = object(NotificationName);
      if (notification->EventName == "Failure")
      {
         Result = TRUE;
      }
      else
      {
         Result = FALSE;
      }
}
```

To apply the ASL filter, access the filter builder. Select the Use ASL Script checkbox and type the name of the ASL script.

Context and Status Criteria for client and server tools

The following example illustrates the format of an ASL filter used to filter context and status criteria for client and server tools. It filters for notifications with an EventName attribute Failure.

The name of the object is assigned to the variable ObjectName. If the value of ObjectName is Failure, the value of Result is set to TRUE and the notification passes the filter, as illustrated in the following code sample:

```
START do {
   timeNow = object(ObjectName)->nowString
   if (glob("*1<0-1>:*:* AM*|*<1-5>:*:* PM*|*12:*:* PM*", timeNow))
   {
      Result = "TRUE";
   }
   else
   {
      Result = "FALSE";
   }
}
```

To apply the ASL filter:

1. Open the filter builder.
2. Click **Use ASL Script**.
3. Type the name of the ASL script.

Saving ASL filters

The ASL filter files must be located in the appropriate directory under the *BASEDIR*/smarts/local/rules directory. The files are located in different subdirectories for different products.

For example, when saving files:

- Global Manager ASL files should be saved in the *BASEDIR*/smarts/local/rules/ics directory.
- When changed, the Global Manager's nl-sample-filter.asl file must be saved as *BASEDIR*/smarts/local/rules/ics/nl-sample-filter.asl.
- Adapter Platform ASL files should be saved in the *BASEDIR*/smarts/local/rules/icoi directory.

NOTICE

When you save a file, specify the path to a specific *BASEDIR*/smarts/local/rules directory.

CHAPTER 11
Importing and Exporting Configurations

This chapter contains the following information:

- Overview 190
- About the default XML files 190
- Running sm_config 193
- Importing configurations 195
- Exporting configurations 198
- Modifying individual repository objects 199
- Importing tools implemented by EMC Smarts adapters 200

Overview

This chapter describes how to use **sm_config** to import and export repository objects such as notification lists, users, user profiles, tools, and escalation policies. It also describes how to use **sm_config** for configuring map icons.

The **sm_config** utility imports configurations, written in XML, for the Global Manager and the Adapter Platform. The **sm_config** utility is used for the following purposes:

- As an alternative to creating notification lists, users and user profiles in the Global Manager Administration Console.
- To batch process multiple configurations.
- To duplicate configurations between Global Managers.
- To import tools for third-party applications.

A strong understanding of XML is necessary if you intend to create and manage repository objects using XML. Appendix C, "XML Reference," provides a guide for creating notification lists, user profiles, users, tools and escalation policies according to the ics-config.dtd XML specification.

About the default XML files

Default configuration settings for the Global Manager are shipped in *BASEDIR*/smarts/conf/ics in a file called ics-default.xml. This file requires the Document Type Definition (DTD), which must be in the same directory. The name of the DTD is ics-config.dtd.

The ics-default.xml file is only used when the Global Manager is started without a repository (*.rps) file. It creates the default repository objects if none are defined in the repository. For example, the ics-default.xml file creates the default notification lists if there are no notification lists found in the repository.

The ics-default.xml file also reads:

- The console_operations.xml file

  The console_operations.xml file resides in the *BASEDIR*/smarts/conf/ics directory. This file requires a DTD, consoleoper-config.dtd. The console_operations.xml file and the consoleoper-config.dtd must be stored in the same directory.

- The map_gif.xml file

  The map_gif.xml file resides in the *BASEDIR*/smarts/conf/maps directory. This file also requires a DTD, map-config.dtd. The map_gif.xml file and the map-config.dtd must be stored in the same directory.

If you start the Global Manager without a repository, the ics-default.xml file is read to instantiate the Global Manager with default configurations. These default configurations include objects which are added to the Global Manager. Objects included in each of the default XML files are:

- ics-default.xml
  - Business Templates:
    - Topology-group.data.template
    - service.data.template
  - Domains:
    - INCHARGE-AM
    - INCHARGE-PM
    - INCHARGE-AM-PM
    - INCHARGE-AM2
    - INCHARGE-OI
    - NCM
    - INCHARGE-DISCMGR
    - INCHARGE-MPLS-ANALYSIS
    - INCHARGE-MPLS-MONITORING
    - INCHARGE-MPLS-TOPOLOGY
  - Notification Lists:
    - Default
    - Maintenance
  - Server Tools:
    - Clear Business Process Event
    - Disable BypassPropogateNotifications
    - Disable Maintenance Tracking
    - Enable BypassPropogateNotifications
    - Enable Maintenance Tracking
    - Import XML Configuration
    - Import XML Domain Configuration
    - Notify Business Process Event
    - Sample - Close Trouble Ticket
    - Sample - Open Trouble Ticket
    - Sample - Ping-all
    - Sample - Ping-device

  - Sample - Ping-Interface
  - Sample - Ping-IP
  - Sample - Telnet
  - Show Storage Impacts
  - Update Business Process Event
- Client Tools:
  - Report Manager
  - Launch Web Browser
  - Dump Properties
  - Ping
- User Profiles:
  - admin-profile
  - default-profile
  - maint-profile
  - oper-profile
- Users:
  - admin
  - default
  - maint
  - oper

◆ console_operations.xml

The console_operations.xml file contains configuration information related to console operations. This file cannot be modified.

◆ map_gif.xml

The map_gif.xml file sets the default image file that is used as a map icon for ICIM classes. This file cannot be modified.

Running sm_config

Using the **sm_config** command you can import objects into and export objects from the repository. In addition, the command allows you to enable, disable, and delete objects.

The options for importing, exporting, and enabling configurations are specified as commands to **sm_config** when you invoke it at the command line. Table 34 on page 193 lists the arguments you should specify when invoking **sm_config**, which is run from *BASEDIR*/smarts/bin.

NOTICE

The information included in Table 34 on page 193 can be retrieved by entering the --help argument when invoking **sm_config** from the command line.

Table 34 Arguments to sm_config for managing configurations (page 1 of 3)

Command option / argument	Description
Commands	
import [command-options] ‹input-xmlfile›	Imports the specified XML file into the Global Manager.
export --server= [‹output-xmlfile›]	Exports the current object configurations to XML. If no output XML filename is specified, the XML is exported as standard output.
delete --classname=‹class› --instancename=‹instance› [--type=‹action_type›]	Removes the specified instance from the Global Manager repository.
enable --classname=‹class› --instancename=‹instance› [--type=‹action_type›]	Enables the specified instance in the Global Manager repository.
disable --classname=‹class› --instancename=‹instance› [--type=‹action_type›]	Disables, but does not delete, the specified instance in the Global Manager repository.
--help	Displays the sm_config command option / argument help.
Command options	
--force	To be used with the import command. Imports repository objects even if existing objects were modified since the XML was last exported. This overwrites existing repository objects that may have been changed. Also -f can be used.

Table 34 Arguments to sm_config for managing configurations (page 2 of 3)

Command option / argument	Description
--replace	To be used with the import command. Deletes all the XML-configured repository objects that currently exist in the Global Manager (specifically notification lists, user profiles, users, tools, escalation policies, and actions). New repository objects, as specified in the XML file, are created in their place. Also -r can be used.
--classname=‹class›	To be used with the delete, enable, or disable command. Specifies the class name of the object you want to modify. Also -c ‹class› can be used.
--instancename=‹instance›	To be used with the delete, enable, or disable command. Specifies the instance name of the object you want to modify. Also -i ‹instance› can be used.
--type=‹action_type›	To be used with the delete, enable, or disable command when you are modifying an action or tool. Specifies the action type. The possible values are server, client, or auto. Also -t ‹action_type›. **Notice:** This option is required if --classname=Action can be used.
Arguments	
‹input-xmlfile›	Name of the Input XML file. Input XML files for Service Assurance Manager must be located in BASEDIR/smarts/local/conf/ics. **Notice:** Only the filename (including the file extension) must be specified, not the path to the file.
‹output-xmlfile›	Name of the file that is generated when exporting configurations. Output XML files for Service Assurance Manager are created in BASEDIR/smarts/local/conf/ics. Output XML files for the Adapter Platform are created in BASEDIR/smarts/local/conf/icoi. If an output XML File is not specified, the XML is written to standard output.
‹class›	Supported Values: • Action (To be used for client, server, and automatic tools.) • AutoActionPolicy (To be used for Escalation Policies.) • NotificationList • UserProfile • User **Notice:** There is no map icon support for ‹class›. You cannot import an individual instance of a map icon.

Table 34 Arguments to sm_config for managing configurations (page 3 of 3)

Command option / argument	Description
‹instance›	Instance value is the logical name of the repository object.
‹action_type›	Used with the --type command option to specify the type of action to delete, enable, or disable. Possible values are: • client • server • auto
Standard options	
--broker=‹broker_location›	Alternate location of the broker as ‹host›:‹port›. This value can be verified in the SM_BROKER environment variable. **Notice:** This argument is optional. Also -b ‹broker_location› can be used.
--server=‹server_name›	Name of Global Manager or Adapter Platform. **Notice:** This argument is required. Also -s ‹server_name› can be used.

Importing configurations

Adding new objects and modifying existing objects requires the following steps:

1. Define the objects in an XML file.
2. Import the XML file.

Defining objects in XML

You have several options for creating XML files for import with **sm_config**, including:

- Export the current configurations to a file and modify the XML as needed. "Exporting configurations" on page 198 provides additional information on exporting configurations.
- Copy and modify an existing object from one of the XML sample files located in *BASEDIR*/smarts/conf/ics.
- Write the XML for the new object.

When making any modifications to the XML, be sure to adhere to the standards of the following DTDs:

- ics-config.dtd located in *BASEDIR*/smarts/conf/ics
- map_config.dtd located in *BASEDIR*/smarts/conf/maps.

"XML Reference" on page 223 provides additional information on Service Assurance XML standards.

Sample XML configuration files

Sample XML documents are provided for use as templates and can be edited to suit your needs. The following sample files are located in *BASEDIR*/smarts/conf/ics:

- Tools: actionconfig-sample.xml
- Notification lists: nlconfig-sample.xml
- Escalation policy: policyconfig-sample.xml
- User profile: profileconfig-sample.xml
- User: userconfig-sample.xml
- Multiple types of repository objects: ics-config-sample.xml
- Map icons: mapgif-sample.xml

*Work*ing with XML sample files

The simplest way to create new repository objects in XML is to use a template. Sample XML files located in *BASEDIR*/smarts/conf/ics are provided for this purpose. Use **sm_edit** to open and save a copy of this file in your *BASEDIR*/smarts/local directory. For example, to create a new notification list, open the nlconfig-sample.xml file using **sm_edit**:

```
BASEDIR/smarts/bin>sm_edit ../conf/ics/nlconfig-sample.xml
```

Modify the current notification list example to create a new one that suits your needs.

For example, change the name="Default" attribute to name="NotificationList_1". Remove everything up to the ColumnHeading declaration and then change one of the column headings to display the "Certainty" percentage of the event.

The revised portion of the sample files looks like:

```
<ics_config>

   <nlconfig name="NotificationList_1" enable="True" timestamp="0">
      <filterconfig type="Expression">
         <criterion attribute="Owner">~maint</criterion>
         <criterion attribute="Owner">~SYSTEM</criterion>
       </filterconfig>

      <columnheading
         column="InstanceDisplayName">"Name"
      </columnheading>
      <columnheading
         column="EventDisplayName">"Event"
      </columnheading>
      <columnheading
         column="Certainty">"Certainty"
      </columnheading>

   </nlconfig>

</ics_config>
```

You can subscribe to multiple notification lists by adding multiple nlconfig elements to the nlconfig XML file. For example, you can add:

```
   <nlconfig name="NotificationList_2" enable="True" timestamp="0">
      <filterconfig type="Expression">
         <criterion attribute="Owner">~maint</criterion>
         <criterion attribute="Owner">~SYSTEM</criterion>
```

```
    </filterconfig>

    <columnheading
       column="InstanceDisplayName">"Name"
    </columnheading>
    <columnheading
       column="EventDisplayName">"Event"
    </columnheading>
    <columnheading
       column="Certainty">"Certainty"
    </columnheading>

</nlconfig>
```

NOTICE

When creating new users, the serverConnect.conf file and the clientConnect.conf file must be modified to authenticate them with usernames and passwords. The *EMC Smarts System Administration Guide* provides additional information on modifying these files.

Importing XML files to the Global Manager

When you have prepared an XML file with the appropriate additions or changes, you can import it into the Global Manager. By default, the **sm_config** import command will add new objects but will not allow modification to objects that exist in the repository. For example, to import a file called NewUsers.xml located in *BASEDIR*/smarts/local/conf/ics, use the following procedure:

1. Change directory to *BASEDIR*/smarts/bin.
2. Run the command:

   sm_config -s INCHARGE-SA import NewUsers.xml

3. Verify that the new configurations were imported by viewing them in the **Global Manager Administration Console**.

Importing options

There are two options for the **sm_config** import command:

- --force: Use this option to override the default and allow modification of existing XML-configurable objects with those in your XML import file.
- --replace: Use this option when you want to remove all the XML-configurable objects that exist in the repository and replace them with the objects in your XML import file. This option is typically used when copying configurations from one Global Manager to another Global Manager in order to duplicate the repository objects.

As an example, use the following command to import configuration objects from a file called ics-conf.xml into a Global Manager called INCHARGE-SA and overwrite any existing configurations:

```
sm_config -s INCHARGE-SA import --force ics-conf.xml
```

Exporting configurations

Exporting the repository objects to XML enables you to view the current configurations of your Global Manager. Exporting configuration-related objects from one Global Manager in XML and then importing them to a second Global Manager allows you to duplicate configurations between Global Managers.

Before exporting, you need to know the name of the appropriate Manager and the location (host and port) of the broker. Exporting produces an XML representation of the current configurations for your notification lists, users, user profiles, tools, escalation policies, and map icons. By default, the XML file is created in the *BASEDIR*/smarts/local/conf/ics directory.

To export the repository object configurations, use the following command syntax:

```
▼sm_config -s <Global_Manager_Name>
   --broker=<host:port> export <file_name>.xml ▲
```

NOTICE

The command must be typed as one line.

For example, to create an XML file called *Current_Configuration.xml* containing the repository objects in the INCHARGE-SA Global Manager on the host and port MyHost:426, the following command is used:

```
▼ sm_config -s INCHARGE-SA
   --broker=MyHost:426 export Current_Configuration.xml ▲
```

NOTICE

The command must be typed as one line.

"Importing XML files to the Global Manager" on page 197 provides detailed information on importing the XML file.

Modifying individual repository objects

Using **sm_config** you can also enable, disable, and delete individual repository objects. To modify these configurations, you need to know:

- Class name
- Instance name
- Action type (if the class name is Action)

Enabling or disabling repository objects

Disabling an object deactivates it and prevents it from being used. To disable or enable an object from the repository, use the following command syntax:

```
▼sm_config [options...] disable --classname=<class>
   --instancename=<instance> ▲
```

NOTICE

The command must be typed as one line.

To enable the object, replace the disable command with the enable command. For example, to enable a user profile called UserProfile1 in a Global Manager named INCHARGE-SA, use the following command:

```
▼ sm_config -s INCHARGE-SA enable --classname=ICS_UserProfile
   --instancename=ICS-Profile1 ▲
```

NOTICE

The command must be typed as one line.

Deleting repository objects

Deleting a repository object completely removes it from the Global Manager. If you are not completely sure you want to remove it, consider disabling the object instead. "Enabling or disabling repository objects" on page 199 provides additional information.

NOTICE

Map objects cannot be deleted.

To delete an object from the repository, use the following command syntax:

```
▼ sm_config [options...] delete --classname=<class>
   --instancename=<instance> ▲
```

NOTICE

The command must be typed as one line.

For example, to delete a server action named Telnet from a Global Manager named INCHARGE-SA, use the following command:

```
▼ sm_config -s INCHARGE-SA delete --classname=ICS_Action
  --instancename=Telnet --type=server ▲
```

NOTICE

The command must be typed as one line.

Importing tools implemented by EMC Smarts adapters

If you are deploying an EMC Smarts adapter that integrates with a third-party application, you may need to import an XML definition file for the tools implemented by that adapter.

The XML definition files for these adapters are imported into the Global Manager using **sm_config**. "Importing XML files to the Global Manager" on page 197 provides detailed information on importing XML files. Refer to the user guide for that adapter for more information regarding the specific tool.

For example, to import the tools definition for the Concord eHealth tool from a file called configureEHealthTools.xml into a Global Manager called INCHARGE-SA, type the following command:

```
sm_config -s INCHARGE-SA import configureEHealthTools.xml
```

CHAPTER 12
SAM Configuration Parameters

This chapter contains the following information:

- System default parameters 202
- ics-event-processing parameters 205
- ics-nl-processing parameters 206
- Console parameters 207
- trap.d conf parameters 210
- Environment variables 210

System default parameters

System defaults are system-wide settings that affect the Global Manager and its clients. Table 36 on page 205 describes the System Defaults parameters.

Table 35 System Defaults parameters (page 1 of 3)

Parameter	Description
Auto Acknowledgement Interval	Defines interval, in seconds, after which an inactive and unowned notification is acknowledged. Notifications that are acknowledged by the Global Manager are owned by the user SYSTEM. **Default**: 300 seconds
Inactive Auto Archive Interval	Defines interval, in seconds, after which an inactive and acknowledged notification is archived. If this value is set to zero, archiving is disabled and notifications will not be deleted, causing Global Manager to use more memory. **Default**: 14400 seconds (4 hours)
Audit Trail Size Limit	Defines the number of audit log entries for each notification that are saved and visible in the Global Console before the log contents are archived. When this limit is reached, half of the entries are written to the notification archive. **Default**: 100 entries
SMTP Server	Defines the name of the SMTP mail server through which mail messages sent by the Global Console's Mail tool are sent. This mail server must be reachable from the host where the Global Manager is running. This value can be overridden in the Global Console. **Default**: "localhost"
Fetch Local Notification Properties	Defines how the properties of a particular notification are retrieved when requested by a client application. Client applications include the Global Console, the dmctl utility, and adapters. If set to "true", properties are retrieved from the Global Manager. If set to "false", properties are retrieved from the Global Manager and from the underlying domain. **Default**: False
RMI Time Out	Defines the maximum amount of time, in seconds, the Global Manager is allotted to retrieve Notification Properties, Find System, or get System Containment information from an underlying domain. If a negative value is specified, no time-out value is set. **Default**: 60 seconds
Number Of Worker Threads	Defines the number of worker threads used by the escalation mechanism in executing tools associated with escalation levels in escalation paths. Increase the number of threads if the escalation mechanism falls behind in executing tools for escalation levels. Valid values are 1 - 10. **Default**: 2

Table 35 System Defaults parameters (page 2 of 3)

Parameter	Description
Broadcast Event State on Restart	Defines whether notifications that result from internal Global Manager events are propagated to clients that subscribe to them. Internal Global Manager events include system disconnects and system reconfigurations. If set to True, notifications resulting from internal Global Manager events are propagated. If set to False, the events are not propagated. **Example:** In the case of the value being TRUE in a hierarchical SAM configuration, notifications at presentation level SAM will go to 'Suspended' state if there is a disconnect between aggregation level SAM and a source domain. This will happen because aggregation SAM will broadcast the notification state change from Active to WAS_ACTIVE. In the case of the value being False, the notifications in the aggregation SAM will change to WAS_ACTIVE, but no change will be sent to presentation level SAM. When aggregation SAM reconnects to the source domain events it will send clears for notifications that are no longer active. When configuring this parameter, the value must be identical for all Service Assurance Manager servers within your Service Assurance Manager deployment. For changes to this parameter configuration to take effect, the Service Assurance Manager servers must be restarted. **Notice:** For optimum benefit, configure this parameter as early in your Service Assurance Manager deployment as possible, before all source domains are added to the Service Assurance Manager server. The quickest way to apply this parameter configuration is to restart the Service Assurance Manager server. However, if you are configuring this parameter in a production environment, apply the parameter configuration to the Service Assurance Manager server by making a minimal change to the configuration of each of the underlying source domain types, and clicking the **Reconfigure** button. One example of a minimal configuration change is to modify the value for the Smoothing Interval parameter. A minimal configuration change is required for each domain type. This results in the Service Assurance Manager server disconnecting from each underlying domain, rereading all the settings including the new modification, and then reconnecting to the underlying domains. **Default**: True
Hierarchical Acknowledge	Defines whether acknowledging or unacknowledging a notification in the server sends the acknowledge (or unacknowledged) information to the underlying servers. If set to True, the notification in the underlying servers are also acknowledged or unacknowledged. If set to False, the notification in the underlying servers are not changed. **Default**: True
Companion UI Hostname	Defines the location of the related Companion UI vApp. The accepted values are the Domain Name System (DNS) host or the IP address of the vApp (for example, companionUI or companionUI.mycompany.com).
Use DNS Lookup in This Domain	Defines whether the system performs Domain Name System (DNS) searches when searching for computer systems. If set to True, the system queries the DNS in this domain. **Notice:** When performing a DNS search, perform the search from the Adapter Platform (OI). **Default**: True

Table 35 System Defaults parameters (page 3 of 3)

Parameter	Description
Use DNS Lookup in Underlying Domains	Defines whether the system performs Domain Name when searching for computer systems. If set to True, the system queries the DNS in underlying domains. **Default**: True
Use DNS Domains in DNS Lookup	Defines whether DNS domains are used when the system performs a DNS search. If set to false, only original domain names are used during the DNS search. **Default**: False
DNS Domain Names	Defines the list of domain names the system uses when performing a DNS search. **Notice:** When entering multiple domain names, separate each domain name with a comma.
Underlying Domains Order	Defines the order in which the underlying domains are searched. **Notice:** When entering multiple domain names, separate each domain name with a comma.
Fully Qualified Domain Name Suffixes	Defines which domain names returned as a result of performing a DNS search are considered to be fully qualified. Domain names matching suffixes entered in this parameter are considered fully qualified.

ics-event-processing parameters

Table 36 on page 205 describes the ics-event-processing parameters.

Table 36 ics-event-processing parameters

Parameter	Description
default debug	This parameter helps track the event flow. If set to true, it prints all the messages under debug flag. **Default**: FALSE
default timeoutTopoValue	This prevents topology synchronization operations from waiting indefinitely. Depending on the value you specify for this parameter in the configuration file, if the child (underlying domain) has gone down, then the parent (domain) waits for the time interval specified here before starting off topology synchronization operations. The default value of 600 seconds can be increased on the basis of size and complexity of the topology. **Default**: 600 seconds
default broadcastEventState	This variable allows you to customize and bypass internal event storms. It displays information about domain-related events, such as system disconnects and system reconfigurations. This parameter is set using the Global Console. **Default**: TRUE

ics-nl-processing parameters

Table 37 on page 206 describes the ics-nl-processing parameters.

Table 37 ics-nl-processing parameters

Parameter	Description
default debug	If set to true, it prints all the messages under debug flag. **Default**: FALSE
default broadcastEventState	This variable allows you to customize and bypass internal event storms. **Default**: True
default timeoutTopoValue	This prevents topology synchronization operations waiting indefinitely. **Default**: 600 seconds

Console parameters

Table 38 on page 207 describes the Console parameters.

Table 38 Console parameters (page 1 of 3)

Parameter	Description
com.smarts.connectionTimeout	This property indicates the maximum amount of time that the console will wait for the Domain Manager pulse check (the upTime attribute) to respond. If there is no response within the specified time, the console disconnects from the Domain Manager. **Default**: 300 seconds (5 minutes)
com.smarts.highLatency	This will be set to true in high latency environment. **Default**: FALSE
com.smarts.securityMessage	Defines the name of the ASCII file containing the security warning message. **Default**: String
com.smarts.server.logMessages	If set to true, causes the server to write messages to the server log file. **Default**:TRUE
com.smarts.mapNodeText.font	Allows the user to define the map icon font.. **Default**: Helvetica
com.smarts.mapNodeText.fontSize	Allows the user to define the map icon font size. **Default**: 12 point
com.smarts.java.log	If specified, the Java log contents are saved to the specified file. **Default**: No file specified
com.smarts.map.showAllBusiness	If set to true, it shows all nodes and edges in a Business Services map, including those that are not part of paths that originate from or terminate at the map central node. If set to false, it shows the nodes and edges in a Business Services map that either originate from or terminate at the map central node. **Default**: TRUE
com.smarts.map.showNormalBanner	If set to True, it shows map nodes with the status of Normal with a green color bar underneath the icon. If false, shows map nodes with the status of Normal without a green color bar underneath the icon. **Default**: FALSE
com.smarts.mapBackgroundColor	Enables you to specify a single background color for all maps. This parameter overrides the default color for all map types. Note that if the setting you enter is invalid, the background color for all maps will be white. **Default**: 600 seconds

Table 38 Console parameters (page 2 of 3)

Parameter	Description
com.smarts.showDomainConsistsOf	Defines whether the InChargeDomainConsistsOf relationship displays in the topology view. If configured to false, the InChargeDomainConsistsOf relationship does not display in the topology view reducing the possibility of the console not responding. If configured to true, all the information for the ConsistsOf relationship displays in the topology view. **Default**: FALSE
com.smarts.fasttopologybrowse	Defines whether the topology browser performance is enhanced for displaying related instance information. Options include: • Off — Default value • On — Speed up enhancement is activated with instance information sorted • Nosort — Speed up enhancement is activated with instance information not sorted **Default**: OFF
com.smarts.clientTool.setEnv	Defines the method of passing parameters to client scripts. If configured to true, the console directly loads parameters passed to the client script into the environment. Configuring this property to true, enables parameters with values that cause Windows parsing of the command to fail to be passed to the client script. If configured to false (or when a value is not specified), the console passes parameters to the client scripts as command line arguments and requires the client script to parse the parameters to load them into the environment. **Default**: FALSE
com.smarts.getMap.timeout	Defines the time, in seconds, that the Get Map function tries to retrieve map data before the function times out. Once the Get Map function times out, the Get map timed out message appears. **Default**: 120
com.smarts.timestampFormat	Overrides the default formatting of timestamps in the console, for example, the First Notify, Last Notify, and Last Change fields in the notification view.
com.smarts.user.timezone	Causes the console to format all times and timestamps according to the specified time zone. Examples of acceptable values are: PST for U.S. Pacific Standard Time and GMT for Greenwich Mean Time. The default is the time zone where the console is running. Class TimeZone at the Java website includes additional information.
com.smarts.writeable	Defines the location of console logs.

Table 38 Console parameters (page 3 of 3)

Parameter	Description
smarts.remote.broker	Defines the default broker for domain attach dialog boxes. **Default**: host:port
com.smarts.securityEnabled	If set to true, security is enabled. This is the default value when the sm_gui.exe is executed. If set to false, security is disabled.
com.smarts.log.level	Defines the level of messages written to the console java log. Options include: • FATAL • SEVERE • SERIOUS • DEFAULT (default option) • WARNING • INFO • CONFIG • DEBUG • TRACE
smarts.allowBGjpegs	If set to true, it allows map background files to be saved in jpeg (.jpg) format.If set to false, map background files will not be saved in jpeg format.
com.smarts.viewer.map.disableMaps	If set to true, it disables the display of maps. If set to false, it does not diable maps. **Default**: TRUE
com.smarts.viewer.hideToolbarText	If set to true, it hides the display text for the toolbar buttons. If set to false, shows the text for the toolbar buttons. **Default**: TRUE
com.smarts.maxTopoInstances	It determines the number of topology instances that will be displayed on the browser. **Default**: 10000
smarts.maps.maxGroupMembers	A positive integer. This is the maximum number of members of a leaf group that will display for a Group Membership map without a prompt dialog box. The dialog box prompts the operator when there is a large number of members to display. **Default**: 100
com.smarts.maxFilterInstances	Enables you to change the maximum number of instances that will automatically display in the Topological Filter tab. **Default**: 50

trap.d conf parameters

The *EMC Smarts Service Assurance Manager Adapter Platform User Guide* provides details on the trap.d conf parameters.

Environment variables

Table 39 on page 210 describes the environment variables.

Table 39 Environment variables

Parameter	Description
SM_NOTIF_DISPLAY_PACKET	If set to true, it displays packet information regarding notification. **Default**: FALSE
SM_SAM_TO_SAM_HYPER_NOTIF_SLEEP	Sets the polling interval to fetch notifications. **Default**: 10
SM_ENABLE_SERIAL_SYNCS	If set to true, enables serial mode for topology synchronization. **Default**: FALSE
SM_DELAY_BETWEEN_TOPO_SYNCS	Sets the minimum delay between two successive synchronization to a domain. **Default**: 120 seconds
SM_JDK_HOME	The SM_JDK_HOME environment variable is used to indicate the location of the JDK, if present. If you provide the location of the JDK while running the appropriate setup script, then the SM_JDK_HOME variable will be defined in the runcmd script.
SM_JAVA	The SM_JAVA environment variable specifies a Java Runtime Environment (JRE) other than that in BASEDIR/smarts/jre. The contents are the paths to the libraries of the JRE. On UNIX, the value of SM_JAVA is added to the LD_LIBRARY_PATH (Solaris and Linux). On Windows, the value of SM_JAVA is added to the PATH. This variable is optional. If not set, runcmd will assume standard paths relative to the value of SM_JAVAHOME.
SM_JAVAHOME	The SM_JAVAHOME environment variable is used in conjunction with SM_JAVA to specify a non-standard JRE. It specifies the location of the JRE, and defines the value of the JVM's ""java.home"" system property. • If SM_JAVAHOME is left unset and SM_JDK_HOME was specified, then SM_JAVAHOME will effectively be set to SM_JDK_HOME/jre. • If left is not set and SM_JDK_HOME is also unset, and if the BASEDIR/jre directory exists, then SM_JAVAHOME will effectively be set to BASEDIR/jre.
SM_JAVA_ENABLED	Set this environment variable to Yes in the BASEDIR directory path to ensure that Java can be successfully started for NOTIF.

APPENDIX A
ICIM Classes Used with Matching Criteria

This appendix contains the following information:

- Classes and attributes for matching managed systems 212
- Attributes for matching notification properties 215

Classes and attributes for matching managed systems

To configure the Global Manager, you may need to specify a matching pattern or matching criteria. The matching pattern or criteria are compared to the name or value of system and notification attributes. Tasks where you might specify a matching pattern include:

- Creating notification lists
- Specifying tagging patterns
- Determining the status or criteria of tools
- Creating selective groups with the Global Console

Matching criteria are strings that are matched against the value of the specified attribute. The criteria can contain any combination of text, integers, and wildcards.

Table 40 on page 212 lists the attributes of the system class hierarchy. The ICIM_System class is at the top of the hierarchy. As you move down the table, the classes become increasingly more specialized until you reach classes that represent specific systems such as routers and switches.

NOTICE

Classes lower in the hierarchy inherit the attributes of classes higher in the hierarchy.

Table 40 System classes and their attributes (page 1 of 3)

System class	Attributes	Type	Description
ICIM_System	CreationClassName	String	Name of the class from which the element was instantiated.
	DisplayClassName	String	Name of the system's class that is displayed in the console.
	DisplayName	String	Name of the system that is displayed in the console.
	Description	String	Description of the system.
	IsManaged	Boolean	Determines if the system is managed by the underlying domain. **Notice:** Unmanaged elements do not appear in the Global Manager topology.
	Name	String	Name of the system.
	PrimaryOwnerContact	String	Contact information for the primary owner of this system.
	PrimaryOwnerName	String	Name of the primary owner of this system.
	SystemName	String	Name of the system of which this component is a part of or this service is hosted by.
ICIM_ComputerSystem			ICIM_ComputerSystem does not add attributes to the system class hierarchy. A computer system is a collection of managed system elements such as file systems, processors, and memory.

Table 40 System classes and their attributes (page 2 of 3)

System class	Attributes	Type	Description
UnitaryComputerSystem	Certification	String	Level of certification assigned to this device during discovery. Possible values are: • CERTIFIED • VALIDATED • TEMPLATE • GENERIC • UNCERTIFIED • UNDISCOVERED • UNSUPPORTED
	Location	String	Description of the physical location of this system.
	Model	String	Vendor name for the system.
	Type	String	Type of the computer system. Possible values are: • BRIDGE • FIREWALL • HOST • HUB • LOADBALANCER • MSFC • NODE • PROBE • ROUTER • RSFC • RSM • SWITCH • TERMINALSERVER • UNCERTIFIED
	Vendor	String	Name of the system's manufacturer.
Bridge	Type	String	Unitary computer system that bridges packets between separate segments. Value is BRIDGE.
Firewall	Type	String	Unitary computer system that controls the flow of traffic between networks. Value is FIREWALL.
Host	Type	String	Unitary computer system that represents a workstation or server. Value is HOST.
Hub	Type	String	Unitary computer system that connects multiple segments. Value is HUB.
LoadBalancer	Type	String	Unitary computer system that distributes client traffic across multiple web servers. Value is LOADBALANCER.
MSFC	Type	String	Unitary computer system that represents a Multi Layer Switch Feature Card. An MSFC is a card installed into a switch to perform routing between VLANs. Value is MSFC.
Node	Type	String	Unitary computer system that has not yet been certified by discovery.

Table 40 System classes and their attributes (page 3 of 3)

System class	Attributes	Type	Description
Probe	Type	String	Unitary computer system that monitors networks or systems. Value is PROBE.
Router	Type	String	Unitary computer system that routes packets between computer networks. Value is ROUTER.
RSFC	Type	String	Unitary computer system that represents a Router Switch Feature Card. An RSFC runs Cisco IOS router software and directly interfaces with Catalyst switches to provide inter-VLAN routing. Value is RSFC.
RSM	Type	String	Unitary computer system that represents a Router Switch Module. Often installed in switches to route packets between VLANs. Value is RSM.
Switch	Type	String	Unitary computer system that switches packets between separate segments. Value is SWITCH.
TerminalServer	Type	String	Unitary computer system that represents a terminal server or similar access device. Value is TERMINALSERVER.
Uncertified	Type	String	[Deprecated, replaced by Node] Unitary computer system that represents a system that has not yet been certified by discovery. Value is HOST.

Attributes for matching notification properties

Table 41 on page 215 lists the notification attributes that are contained in notifications generated by the Global Manager. When you create a matching pattern, use the name of the attribute, not the column heading from the Notification Log.

Table 41 Notification attributes (page 1 of 3)

Attribute	Default column name	Value type	Description
Acknowledged	Acknowledged	Boolean	Indicates if a notification has been acknowledged. Possible values include: • TRUE if the notification has been acknowledged. • FALSE if not.
Active	Active	Boolean	Indicates if a notification is active. Possible values include: • TRUE if the notification is active. • FALSE if not.
AuditTrail	n/a	String	Audit trail includes five fields separated by spaces. The fields include: • Serial number — A unique number that identifies the audit trail entry. • Timestamp — Identifies when this audit trail entry was written. • User — The InCharge username associated with this audit trial entry. • Action type — Identifies the reason for the audit trail entry. • Text — A brief description of the audit trail entry.
Category	Category	String	Type of notification sent by the Global Manager. Possible values include: • Availability • Discovery • Error • IMPACT • Operational • Performance • PowerSupply • Resource • Temperature
Certainty	Certainty	Float	Confidence that this notification is the correct diagnosis. Value ranges from 0 to 100.
ClassDisplayName	Class	String	Name of the class that appears to the user.
ClassName	ClassName	String	Class name of the instance where this event occurred. This attribute, with InstanceName and EventName uniquely identifies this notification.
ClearOnAcknowledge	n/a	Boolean	Indicates that this event should be cleared when it is acknowledged. Default is FALSE.
ElementClassName	Element Class	String	Class name of the managed element most closely related to this event.

Table 41 Notification attributes (page 2 of 3)

Attribute	Default column name	Value type	Description
ElementName	Element Name	String	Name of the managed element most closely related to this event.
EventDisplayName	Event	String	Name of the notification that appears to the user.
EventName	EventName	String	Name of this notification. This attribute, with ClassName and InstanceName, uniquely identify this notification.
EventState	Event State	String	Describes the state of the event. Value can be one of: • ACTIVE • WAS_ACTIVE • SUSPENDED • INACTIVE • UNITIALIZED
EventText	Event Text	String	A description of the notification.
EventType	Event Type	String	MOMENTARY when the notification has no duration. DURABLE if the notification has a period for which it is active, such as a link failure.
FirstNotifiedAt	First Notify	Integer	Time, in seconds, when the notification first became active. Events diagnosed by an underlying domain include timestamps that accurately reflect when the event occurred. The Global Manager uses this value from the event for the FirstNotifiedAt/LastNotifiedAt times when it creates the notification.
Impact	Impact	Integer	Numeric value that indicates the effect of this event on related elements.
InMaintenance	In Maintenance	Boolean	Indicates if the device is in maintenance mode. Possible values include: • TRUE if the device is in maintenance mode. • FALSE if not.
InstanceDisplayName	Name	String	Name of the instance that is displayed to the user.
InstanceName	InstanceName	String	Name of the object where this notification occurred. This attribute, with ClassName and EventName, uniquely identify this notification.
IsProblem	IsProblem	Boolean	IsProblem is defined as TRUE if (and only if) at least one of the types is “PROBLEM”, and none of the types are anything other than “PROBLEM” or “UNKNOWN”. **Notice:** Since filtering on IsRoot can result in transient behavior, it is recommended to filter on IsProblem instead.

Table 41 Notification attributes (page 3 of 3)

Attribute	Default column name	Value type	Description
IsRoot	IsRoot	Boolean	Indicates if the notification is an authentic problem. Possible values include: • TRUE if the notification is an authentic problem (root cause). • FALSE if not. If a given notification has no CausedBy relationship, it is declared to be a root event, and IsRoot for that notification is set to TRUE. It is often useful to filter on IsRoot to find those notifications which require immediate attention.
LastChangedAt	Last Change	Integer	Time, in seconds, when the status of the notification last changed.
LastClearedAt	Last Clear	Integer	Time, in seconds, when the notification was last cleared.
LastNotifiedAt	Last Notify	Integer	Time, in seconds, when the notification was last notified.
OccurrenceCount	Count	Integer	Number of times the notification has occurred.
Owner	Owner	String	Name of the person responsible for this notification. Value is SYSTEM when acknowledged by the Global Manager.
Severity	Severity	Integer	Level of severity for this notification. 1 = CRITICAL 2 = MAJOR 3 = MINOR 4 = UNKNOWN 5 = NORMAL **Notice:** Only the numbers, not the text descriptions, are passed by the Global Manager.
SourceDomainName	Source	String	Name of the underlying domain that sent this notification. If more than one domain is listed, the names are separated by commas.
SourceEventType	SourceEventType	String	Indicates the type of event from an underlying source or sources. Value can be one of: • PROBLEM • EVENT • AGGREGATE • UNKNOWN. The default value is UNKNOWN. (Not to be confused with Event Type.)
TroubleTicketID	Ticket ID	String	Trouble-ticket number associated with this notification.
UserDefined1-20	User Defined 1-20	String	Twenty notification attributes can be defined by the user. You can set a value with a hook script specified in DomainType section of ics-default.xml file.

NOTICE

For attributes that contain a time value, time is counted from Midnight, January 1st, 1970 (GMT). In the Global Console, these values are converted to a date and time.

APPENDIX B
Wildcard Patterns

This appendix contains the following information:

- Types of wildcard patterns 220

Types of wildcard patterns

A wildcard pattern is a series of characters that are matched against incoming character strings. You can use these patterns when you define pattern matching criteria.

Matching is done strictly from left to right, one character or basic wildcard pattern at a time. Basic wildcard patterns are defined in Table 42 on page 220. Characters that are not part of match constructs match themselves. The pattern and the incoming string must match completely. For example, the pattern abcd does not match the input abcde or abc.

A compound wildcard pattern consists of one or more basic wildcard patterns separated by ampersand (&) or tilde (~) characters. A compound wildcard pattern is matched by attempting to match each of its component basic wildcard patterns against the entire input string. Compound wildcard patterns are listed in Table 43 on page 221.

If the first character of a compound wildcard pattern is an ampersand (&) or tilde (~) character, the compound is interpreted as if an asterisk (*) appeared at the beginning of the pattern. For example, the pattern ~*[0-9]* matches any string not containing any digits. A trailing instance of an ampersand character (&) can only match the empty string. A trailing instance of a tilde character (~) can be read as “except for the empty string.”

NOTICE

Spaces are interpreted as characters and are subject to matching even if they are adjacent to operators like “&”.

Table 42 Basic wildcard patterns (page 1 of 2)

Character	Description
?	Matches any single character. For example, server?.example.com matches server3.example.com and serverB.example.com, but not server10.example.com.
*	Matches an arbitrary string of characters. The string can be empty. For example, server*.example.com matches server-ny.example.com and server.example.com (an empty match).
[set]	Matches any single character that appears within [set]; or, if the first character of [set] is (^), any single character that is *not* in the set. A hyphen (-) within [set] indicates a range, so that [a-d] is equivalent to [abcd]. The character before the hyphen (-) must precede the character after it or the range will be empty. The character (^) in any position except the first, or a hyphen (-) at the first or last position, has no special meaning. For example, server[789-].example.com matches server7.example.com through server9.example.com, but not server6.example.com. It also matches server-.example.com. For example, server[^12].example.com does not match server1.example.com or server2.example.com, but will match server8.example.com.

Table 42 Basic wildcard patterns (page 2 of 2)

Character	Description
‹n1-n2›	Matches numbers in a given range. Both *n1* and *n2* must be strings of digits, which represent nonnegative integer values. The matching characters are a non-empty string of digits whose value, as a nonnegative integer, is greater than or equal to *n1* and less than or equal to *n2*. If either end of the range is omitted, no limitation is placed on the accepted number. For example, 98.49.‹1-100›.10 matches a range of IPv4 addresses from 98.49.1.10 through 98.49.100.10. Example of an omitted high end of the range: ‹50-› matches any string of digits with a value greater than or equal to 50. Example of an omitted low end of the range: ‹-150› matches any value between zero and 150. For a more subtle example: The pattern ‹1-10›* matches 1, 2, up through 10, with * matching no characters. Similarly, it matches strings like 9x, with * matching the trailing x. However, it does not match 11, because ‹1-10› always extracts the longest possible string of digits (11) and then matches only if the number it represents is in range.
|	Matches alternatives. For example,"ab|bc|cd" without spaces matches exactly the three following strings: "ab", "bc", and "cd". A | as the first or last character of a pattern accepts an empty string as a match. Example with spaces "ab | bc" matches the strings "ab" and " bc".
\	Removes the special status, if any, of the following character. Backslash (\) has no special meaning within a set ([set]) or range (‹n1-n2›) construct.

Special characters for compound wildcard patterns are summarized in Table 43 on page 221.

Table 43 Compound wildcard patterns

Character	Description
&	"And Also" for a compound wildcard pattern. If a component basic wildcard pattern is preceded by & (or is the first basic wildcard pattern in the compound wildcard pattern), it *must* successfully match. Example: *NY*&*Router* matches all strings which contain NY and also contain Router. Example: ‹1-100›&*[02468] matches even numbers between 1 and 100 inclusive. The ‹1-100› component only passes numbers in the correct range and the *[02468] component only passes numbers that end in an even digit. Example: *A*|*B*&*C* matches strings that contain either an A or a B, and also contain a C.
~	"Except" for a compound wildcard pattern (opposite function of &).If a component basic wildcard pattern is preceded by ~, it *must not* match. Example: 10.20.30.*~10.20.30.50 matches all devices on network 10.20.30 except 10.20.30.50. Example: *Router*~*Cisco*&*10.20.30.*~10.20.30.‹10-20›* matches a Router, except a Cisco router, with an address on network 10.20.30, except not 10.20.30.10 through 10.20.30.20.

APPENDIX C
XML Reference

This chapter contains the following information:

- Sample XML files 224
- Structure of an XML document 224
- Configuration elements 226
- Global Manager configuration DTDs 250

Sample XML files

This chapter describes the XML needed to create and modify repository objects such as notification lists, users, user profiles, tools, escalation policies, console operations, and map icons. The XML files are imported into the repository using the **sm_config** utility. Chapter 11, "Importing and Exporting Configurations," provides additional information. You must have a strong understanding of XML if you intend to create and manage repository objects using XML.

Sample XML documents are provided to use as a template and can be edited to suit your needs. The following sample files are located in the *BASEDIR*/smarts/conf/ics directory:

- Tools: actionconfig-sample.xml
- Notification Lists: nlconfig-sample.xml
- Escalation Policy: policyconfig-sample.xml
- User Profile: profileconfig-sample.xml
- User: userconfig-sample.xml
- Multiple types of repository objects: ics-config-sample.xml
- Map Icons: mapgif-sample.xml

NOTICE

The instance_maptype-sample.xml file is reserved for future use.

A useful way of viewing valid, complex XML is to export the configurations you make in the Global Manager Administration Console. "Exporting configurations" on page 198 describes how to export the configuration of a Global Manager.

Structure of an XML document

The basic building blocks of an XML document are elements, attributes, and values. Tag is a general term that can be used for elements and attributes. A relevant sample of the Global Manager configuration XML DTD is provided with each of the structural descriptions.

Element

An element is a tag that is a container for subelements and attributes. For example, the nlconfig element contains defining information for the notification list you are adding or editing in the repository. If an element comprises any subelements, these are defined in parentheses after the element declaration in the DTD. For example, the notification list element, nlconfig, comprises a filter, column heading, and user profile as seen in this sample of the DTD:

```
<!ELEMENT  nlconfig  (filterconfig | columnheading | userprofile)*>
```

Attribute declaration (attribute list)

Attribute declarations, as defined in the ATTList tag of the DTD, provide a detailed definition for an element. Many elements use name declarations to identify the element as well as enable declarations that instruct the Global Manager to create and enable the object in repository.

An attribute is a source of additional information about an element, used to define the element:

- The value of an attribute must match one of the values listed. Values must appear in parentheses and be separated by OR (|) symbols.
- Attribute may be required (#REQUIRED), optional (#IMPLIED), or have a default value in quotes.
- All attribute values must be quoted.

The following is an example of an attribute declaration in the DTD for the notification (nlconfig) element:

```
<!ATTLIST nlconfig
   name CDATA #REQUIRED
   enable (TRUE | FALSE | true | false | True | False) "true"
   timestamp CDATA "0">
```

This indicates that the notification list element has three attributes:

- Name — A character string that is required.
- Enable — Determines whether the repository object is enabled when it is imported. This attribute can take one of the values in parentheses. The default value is true.
- Timestamp — Character string that identifies the date and time the object was last modified. The 0 indicates that, by default, there is no timestamp.

Value

The value setting of the attribute declaration is paired with the declaration in name=value groupings. The values define the properties of the element you are adding, modifying, or deleting in the repository. In the following example, the notification list element is made up of a notification list element tag (“nlconfig”), attributes (name and enable), values (Hosts_and_Routers and false), and a close tag for the element (/›):

```
<nlconfig name="Hosts_and_Routers" enable="false"> </nlconfig>
```

An attribute value is either required or implied. If the value is an attribute that you must define such as the name of the object, this is indicated by the phrase #REQUIRED after the attribute in the DTD. If the value is implied, then a default value is indicated in quotes. For example, TRUE is the default value for the enable attribute of the actionconfig element:

```
<!ATTLIST actionconfig
   name CDATA #REQUIRED
   enable (TRUE | FALSE | true | false | True | False) "TRUE"
   type ( server | client | auto ) #REQUIRED >
```

Value indicators

Table 44 on page 226 lists the terms and symbols used to build the XML expressions with proper syntax structure.

Table 44 XML value indicators

Value indicator	Description
CDATA	Character data (CDATA) type attributes may contain only character data.
#PCDATA	Parsed character data is text that will be examined by the parser for entities and proper syntax.
‹element›*	The asterisk (*) indicates that the content is optional (may occur zero or more times).
‹element›?	The question mark (?) after an element makes it optional, but only one may appear.

Configuration elements

This section describes the elements of the ics-config.dtd. For each element, an explanation of the attributes and values is provided as defined by the DTD. An XML example for most elements is provided. Additional requirements and guidelines for the XML representations are defined in the Semantics section of each element.

ics_config

This is the root element of any Service Assurance configuration XML document. In any XML document that you create for import into the Global Manager, this root element must be the opening and closing tag of the document. There are no attributes for the ics_config element. All configuration elements are contained within the ics_config element tags.

ics_config DTD description

Within the DTD, ics_config is defined as the root element. The nlconfig, actionconfig, userprofileconfig, userconfig, and policyconfig elements are listed as subelements in parentheses:

```
<!ELEMENT ics_config (nlconfig | actionconfig | userprofileconfig |
   userconfig | policyconfig | consoleoperationconfig | map_gifconfig
   )*>
```

ics_config XML example

For example, to define a new user profile, it must be nested within the root ics_config element:

```
<ics_config>
   <userprofileconfig name="default" enable="True" timestamp="0">
      <nl>Default</nl>
      <user>user-default</user>
      <console>NotificationLog</console>
   </userprofileconfig>
</ics_config>
```

Notification list (nlconfig)

This declares the notification list object and indicates that the subelements that can be nested are filters, column headings, and user profiles.

nlconfig DTD description

The nlconfig element defines the notification list repository object. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT nlconfig (filterconfig | columnheading | userprofile)>

 <!ATTLIST nlconfig
   name      CDATA #REQUIRED
   enable    (TRUE | FALSE | true | false | True | False)"TRUE"
   timestamp     CDATA "0" >
```

Table 45 on page 227 identifies the attribute declarations and possible values for the nlconfig element.

Table 45 Attribute-list declarations for the nlconfig element

XML attribute	Description	Possible value	Value required/ default value
name	A unique name for the notification list.	CDATA	Required.
enable	Indicates whether the notification list should be enabled when imported into the Global Manager or whether it was enabled at the time of export.	TRUE True true FALSE False false	Not Required. Default is “True” so the notification list is enabled when imported.
timestamp	When exporting, the value is the last time the object was modified by any user through the Administration Console or through sm_config. When importing, the timestamp defined in the XML file is compared with the timestamp of the object in the repository. Modifications to an existing object will fail when the import timestamp is later than the existing repository timestamp (unless the force option is used).	CDATA	Not Required. Default is “0”.

nlconfig XML example

For example, a notification list named "NL1" that is enabled without a specific timestamp would be defined as:

```
<nlconfig name="NL1" enable="True" timestamp="0"> </nlconfig>
```

NOTICE

To enable multiple notification lists, add multiple nlconfig elements to the Service Assurance configuration XML document.

Filter (filterconfig)

The filterconfig element defines filter criteria that determines which notifications or topology elements may pass through it. The filterconfig element is a subelement to the following elements:

- nlconfig
- policyconfig
- pathconfig
- status_criteria (for actions)
- context_criteria (for actions)

filterconfig DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  filterconfig (isa?, (criterion* | filename))>
```

This declares the filter object and indicates that the subelements that can be nested are isa, criterion, and filenames.

```
<!ATTLIST filterconfig
   type(ASL | EXPRESSION | asl | expression | Expression)  #REQUIRED
   enable (TRUE | FALSE | true | false | True | False) "TRUE">
```

Table 46 on page 228 identifies the attribute declarations for the filterconfig element.

Table 46 Attribute-list declarations for the filterconfig element

XML attribute	Description	Possible value	Value required/ default value
type	One of two types of notification filters. Chapter 10, "Working with Filters," describes expression and ASL filters.	EXPRESSION Expression expression ASL asl	Required.
enable	Indicates whether the filter should be enabled when imported into the Global Manager or whether it was enabled at the time of export.	TRUE True true FALSE False false	Not required Default is "True" so the filter is enabled when imported.

filterconfig XML example

For example, to indicate that an ASL filter called nl_notify.asl is to be utilized for a notification list called NL1, use the following syntax:

```
<nlconfig name="NL1" enable="True" timestamp="0">
   <filterconfig type="ASL">
      <filename>nl_notify.asl</filename>
   </filterconfig>
</nlconfig>
```

In another example, the ics-default.xml file defines a filter for the default notification list that allows all notifications not owned by maint or system. The XML representation of this is:

```
 <filterconfig type="EXPRESSION" enable="TRUE">
   <criterion attribute="Owner">~maint</criterion>
   <criterion attribute="Owner">~SYSTEM</criterion>
</filterconfig>
```

filterconfig semantics

You can have multiple criteria defined in a filter; these criteria are grouped by AND. If you have multiple filterconfig definitions, these are grouped by OR.

filename

This subelement is used when you are using an ASL filter. The filename refers to the ASL file to be used as the filter for the notification list. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT filename (#PCDATA)>
```

filename XML example

For example, to define an ASL filter using a file called nl_notify.asl, use the following syntax:

```
<filterconfig type="ASL" enable="TRUE">
   <filename>nl_notify.asl</filename>
</filterconfig>
```

isa

This isa element is used to define a filter for a particular ICIM class. There are no attributes associated with the isa element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT isa  (#PCDATA)>
```

Any ICIM class name is a valid #PCDATA value for this element.

isa XML example

For example, to define an expression filter that allows all instances of the UnitaryComputerSystem class to pass through, use the following syntax:

```
<filterconfig type="Expression" enable="TRUE">
   <isa>UnitaryComputerSystem</isa>
</filterconfig>
```

criterion

The criterion element is used to populate an expression filter with the notification attributes against which the notifications or topology elements are matched.

NOTICE

The criterion element utilizes wildcards to build expressions. “Wildcard Patterns” on page 219 describes wildcards.

criterion DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT criterion (#PCDATA)>
<!ATTLIST criterion
   attribute CDATA  #REQUIRED>
```

Table 47 on page 230 identifies the attribute declarations for the criterion element.

Table 47 Attribute-list declarations for the criterion element

XML attribute	Description	Possible value	Value required/ default value
attribute	Name of the notification attribute. Must be one of the attributes listed in "Attributes for matching notification properties" on page 215.	#PCDATA	Required

criterion XML example

For example, to define an expression filter that will only pass on notifications that do not have an open trouble ticket ID (~*OPEN*), use the following syntax:

```
<filterconfig type="Expression" enable="TRUE">
   <criterion attribute="TroubleTicketID">~*OPEN*</criterion>
</filterconfig>
```

Column heading (columnheading)

The columnheading element defines the names of the columns in the notification log in the Global Console.

columnheading DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT columnheading (#PCDATA)>

<!ATTLIST columnheading
   column CDATA   #REQUIRED
   column (String | Boolean | Time | Integer | Float | Percentage )
   "String">
```

Table 48 on page 231 identifies the attribute declarations for the columnheading element.

Table 48 Attribute-list declarations for the columnheading element

XML attribute	Description	Possible value	Value required/ default value
column	The name of the notification attribute.	CDATA (string?)	Required.
column	The text you want to display as the column heading in the Global Console for users of this notification list.	String Boolean Time Integer Float Percentage	Not required Default is "String" so the column heading text must be defined.

columnheading XML example

For example, this sample represents a notification list called NL1 with three columns. The ClassDisplayName attribute will be displayed in the NL1 Class Name column. The InstanceDisplayName attribute will be displayed in the NL1 Display Name column. Finally, the EventDisplayName attribute will be displayed in the NL1 Event Name column:

```
<nlconfig name="NL1" enable="True" timestamp="0">
   <columnheading column="ClassDisplayName">NL1 Class Name
      </columnheading>
   <columnheading column="InstanceDisplayName">NL1 Instance Name
      </columnheading>
   <columnheading column="EventDisplayName">NL1 Event Name
      </columnheading>
</nlconfig>
```

Column heading values

The column headings for a notification list correspond to notification attributes. Table 49 on page 231 describes the notification attributes, data types, and default display names of the column headings.

Table 49 Column heading attributes and default display names

Notification attributes	Data type	Default column name
Name	STRING	InternalEventHandle
ClassName	STRING	ClassName
InstanceName	STRING	InstanceName
EventName	STRING	EventName
ClassDisplayName	STRING	Class
InstanceDisplayName	STRING	Name
EventDisplayName	STRING	Event
ElementClassName	STRING	Element Class
ElementName	STRING	Element Name
SourceDomainName	STRING	Source

Table 49 Column heading attributes and default display names (continued)

Notification attributes	Data type	Default column name
Active	BOOLEAN	Active
OccurrenceCount	INTEGER	Count
FirstNotifiedAt	TIME	First Notify
LastNotifiedAt	TIME	Last Notify
LastClearedAt	TIME	Last Clear
LastChangedAt	TIME	Last Change
IsRoot	BOOLEAN	IsRoot
Acknowledged	BOOLEAN	Acknowledged
Owner	STRING	Owner
EventType	STRING	Event Type
EventState	STRING	Event State
Category	STRING	Category
EventText	STRING	Event Text
Severity	INTEGER	Severity
Impact	INTEGER	Impact
Certainty	PERCENTAGE	Certainty
InMaintenance	BOOLEAN	In Maintenance
TroubleTicketID	STRING	Ticket ID
UserDefinedN	STRING	User Defined N

Tool/Action (actionconfig)

The actionconfig element identifies the tool repository object. Chapter 8, “Tool Configuration for the Global Manager,” describes tool configuration.

actionconfig DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  actionconfig (program_name | timeout? | trace | display |
   status_criteria? | context_criteria | userprofile )*>
<!ATTLIST actionconfig
    name   CDATA  #REQUIRED
    enable (TRUE | FALSE | true | false | True | False)"TRUE"
    type  (SERVER | CLIENT | AUTO | server | client | auto | Server |
          Client | Auto)   #REQUIRED
    timestamp   CDATA  "0">
```

NOTICE

The userprofile subelement is used by the Global Console to reference the user profile in the display. This is not used for configuration. "User profile (userprofileconfig)" on page 239 describes how to configure user profiles using XML.

This declares the tool object and indicates that the allowable subelements to be nested are:

- Program Name ("program_name DTD description" on page 234 provides additional information)
- Timeout ("timeout DTD description (for actionconfig)" on page 235 provides additional information)
- Trace ("trace DTD description" on page 236 provides additional information)
- Display ("display DTD description" on page 235 provides additional information)
- Status Criteria ("status_criteria DTD description" on page 236 provides additional information)
- Context Criteria ("context_criteria DTD description" on page 237 provides additional information)

Table 50 on page 233 identifies the attribute declarations for the actionconfig element.

Table 50 Attribute-list declarations for the actionconfig element (page 1 of 2)

XML attribute	Description	Possible value	Value required/ default value
name	A unique name for the tool.	CDATA (string?)	Required.
enable	Indicates whether the tool should be enabled when imported into the Global Manager or whether it was enabled at the time of export.	TRUE True true FALSE False false	Not required Default is True so the tool is enabled when imported.
type	Indicates the classification of the tool: • Server — Tools executed on the same host as the Global Manager. These are displayed in the Tools menu on the Global Console. • Client — Tools executed locally by the console. These are displayed in the Tools menu on the Global Console. • Auto — Tools executed automatically through escalation policies or the automatic action adapter (sm_adapter). These are not visible in the tools menu on the Global Console.	SERVER Server server CLIENT Client client AUTO Auto auto	Required.

Table 50 Attribute-list declarations for the actionconfig element (page 2 of 2)

XML attribute	Description	Possible value	Value required/ default value
timestamp	When exporting, the value is the last time the object was modified by any user through the Global Manager Administration Console or through sm_config. When importing, the timestamp defined in the XML file is compared with the timestamp of the object in the repository. Modifications to an existing object will fail when the import timestamp is later than the existing repository timestamp (unless the force option is used).	CDATA	Not required Default is "0".

actionconfig XML example

For example, a server tool named Action1 that is enabled, running a server type program called server-action.sh without a specific timestamp would be defined as:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <program_name>server-action.sh</program_name>
</actionconfig>
```

Program name (program_name)

This is the name of the program that the Global Manager (for server and automatic tools) or the host running the Global Console (for client tools) executes when the tool is invoked. This subelement is implicitly required when defining an action in XML.

program_name DTD description

There are no attributes associated with the program_name element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT program_name (#PCDATA)>
```

program_name XML example

For example, if the server tool called Action1 is designed to execute a program called server-action.sh, use the following syntax:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <program_name>server-action.sh</program_name>
</actionconfig>
```

Timeout (actionconfig)

This specifies the maximum number of seconds to wait for the tool script to complete. If the tool script does not complete within the specified timeout interval, the Global Manager terminates the tool.

timeout DTD description (for actionconfig)

There are no attributes associated with the timeout element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT timeout (#PCDATA)>
```

timeout XML example (for actionconfig)

For example, if the server tool called Action1 is designed to execute a program called server-action.sh with a timeout value of 60 seconds, then the syntax to define that would look like this:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <timeout>60</timeout>
   <program_name>server-action.sh</program_name>
</actionconfig>
```

Display (display)

This specifies whether the tool should display output to the console. If this parameter is set to true, the console will open the Tool Output window immediately after running the tool. If false, the console only opens the Tool Output window if there is an error. The default is true.

display DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT display EMPTY>
   <!ATTLIST display
    display_value (FALSE | TRUE | true | false | True | False)"TRUE">
```

Table 51 on page 235 identifies the attribute declarations for the display element.

Table 51 Attribute-list declarations for the display element

XML attribute	Description	Possible value	Value required/ default value
display_value	Indicates whether or not to open the Tool Output window when running the tool. If false, the Tool Output window will only open if an error occurs when the tool runs.	TRUE True true FALSE False false	Not required. Default is "TRUE".

display XML example

For example, if the server tool called "Action1" is designed to execute a program called "server-action.sh" with a timeout value of 60 seconds, and you want to see the display output, then the syntax to define that would look like this:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <timeout>60</timeout>
   <program_name>server-action.sh</program_name>
   <display display_value="True" />
</actionconfig>
```

Trace (trace)

Trace indicates that additional debugging information will be included in the display output of the tool. This element is only valid if the display element is also declared.

trace DTD description

There are no attributes associated with the trace element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT trace EMPTY>
<!ATTLIST trace
    trace_value (FALSE | TRUE | true | false | True | False)"FALSE">
```

The EMPTY declaration indicates that the element has no sub-elements or character data associated with it.

trace XML example

For example, to turn on the trace utility when defining an action, use the following syntax:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <timeout>60</timeout>
   <program_name>server-action.sh</program_name>
   <display display_value="True" /diplay>
   <trace trace_value="True"/>
</actionconfig>
```

Status criteria (status_criteria)

This element determines whether a client or server tool is accessible from the popup menu in the Global Console. The tool is always accessible when the status_criteria element is omitted. This parameter is not available for automated tools.

status_criteria DTD description

There are no attributes associated with the status_criteria element. But, there is a filterconfig subelement that determines which notifications utilize the tool. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT status_criteria (filterconfig?)>
```

status_criteria XML example

For example, there is a server tool called Action1 designed to execute a program called server-action.sh with a timeout value of 60 seconds, with a display output. To set the status criteria to ensure that the tool only appears in the popup menu if there is an open trouble ticket for that notification, use the following syntax:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <timeout>60</timeout>
   <program_name>server-action.sh</program_name>
   <display display_value="True" /diplay>
   <status_criteria>
      <filterconfig type="Expression">
         <criterion attribute="TroubleTicketID">~*OPEN*</criterion>
      </filterconfig>
   </status_criteria>
</actionconfig>
```

Context criteria (context_criteria)

This determines whether the tool appears on the tool menu in the Global Console when the user right-clicks on a notification. The tool automatically appears in the tool menu when this parameter is omitted. This parameter is not available for automated tools.

context_criteria DTD description

There are no attributes associated with the context_criteria element. But, there is a filterconfig subelement that determines which notifications utilize the tool. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT context_criteria (filterconfig?)>
```

For example, there is a server tool called Action1 that executes a program called server-action.sh with a timeout value of 60 seconds, with a display output. To make sure that the tool is enabled in the popup menu for Router notifications, the syntax is:

```
<actionconfig name="Action1" type="Server" enable="True"
   timestamp="0">
   <timeout>60</timeout>
   <program_name>server-action.sh</program_name>
   <display display_value="True" /diplay>
   <context_criteria>
      <filterconfig type="Expression">
         <isa>ICIM_Notification</isa>
         <criterion attribute="ClassName">Router</criterion>
      </filterconfig>
   </context_criteria>
</actionconfig>
```

User (userconfig)

A user is an individual member of a user profile. If no user profile is specified, then the users are assigned to the Default user profile.

NOTICE

The userconfig element contains a subelement called userprofile. This subelement is used by the Global Console to reference the user profile in the display. This is not used for configuration. "User profile (userprofileconfig)" on page 239 describes the userprofileconfig element which is used to configure user profiles.

userconfig DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  userconfig (userprofile)*>
<!ATTLIST userconfig
   name   CDATA    #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"TRUE"
      timestamp CDATA  "0">
```

Table 52 on page 238 identifies the attribute declarations and possible values for the userconfig element.

Table 52 Attribute-list declarations for the userconfig element

XML attribute	Description	Possible value	Value required/ default value
name	A unique name for the user.	CDATA	Required.
enable	Indicates whether the user should be enabled when imported into the Global Manager or whether it was enabled at the time of export.	TRUE True true FALSE False false	Not required. Default is "True" so the notification list is enabled when imported.
timestamp	When exporting, the value is the last time the object was modified by any user through the Administration Console or through sm_config. When importing, the timestamp defined in the XML file is compared with the timestamp of the object in the repository. Modifications to an existing object will fail when the import timestamp is later than the existing repository timestamp (unless the force option is used).	CDATA	Not required. Default is "0".

userconfig XML example

For example, if two users called User1 and User2 are designed to be enabled with no specific timestamp, then the syntax to define that would look like this:

```
<userconfig name="User1" enable="True" timestamp="0"> </userconfig>
<userconfig name="User2" enable="True" timestamp="0"> </userconfig>
```

NOTICE

If you are creating new users, be sure they are authenticated with a username and password in the serverConnect.conf and clientConnect.conf files.

User profile (userprofileconfig)

A user profile defines what a set of users can see and do when they log in to the Global Console. The user profile determines what notification lists, consoles, and actions a set of users can access.

userprofileconfig DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  userprofileconfig (nl | console | user | action |
   consoleoperation)*> <!ATTLIST userprofileconfig
    name CDATA  #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"TRUE"
    timestamp   CDATA          "0">
```

This declaration states that the userprofileconfig element has four required subelements: nl, console, user, and action.

Table 53 on page 240 identifies the attribute declarations and possible values for the userprofileconfig element.

Table 53 Attribute-list declarations for the userprofileconfig element

XML attribute	Description	Possible value	Value required/ default value
name	A unique name for the user profile.	CDATA (string?)	Required.
enable	Indicates whether the user profile should be enabled when imported into the Global Manager or whether it was enabled at the time of export.	TRUE True true FALSE False false	Not required. Default is "True" so the notification list is enabled when imported.
timestamp	When exporting, the value is the last time the object was modified by any user through the Administration Console or through sm_config. When importing, the timestamp defined in the XML file is compared with the timestamp of the object in the repository. Modifications to an existing object will fail when the import timestamp is later than the existing repository timestamp (unless the force option is used).	CDATA	Not required. Default is "0".

userprofileconfig XML example

For example, in the ics-default.xml file, the "maint-profile" user profile is defined. It has the following attributes:

- One notification list (Maintenance).
- One console (NotificationLog).
- One user associated with it (maint).

The maint-profile user profile also has several actions and console operations defined:

```
<userprofileconfig name="maint-profile" enable="True" timestamp="0">
   <nl>Maintenance</nl>
   <console>NotificationLog</console>
   <user>maint</user>
   <action type="server">Sample - Telnet</action>
   <action type="server">Sample - Ping-Interface</action>
   <action type="server">Sample - Ping-IP</action>
   .
   .
   .
   <consoleoperation>Browse</consoleoperation>
   <consoleoperation>BrowseDetail</consoleoperation>
   <consoleoperation>ExpandMapNode</consoleoperation>
   <consoleoperation>ShowContainment</consoleoperation>
   .
   .
</userprofileconfig>
```

Notification list (nl)

The nl is the name of notification list defined by the nlconfig element. "Notification list (nlconfig)" on page 227 provides additional information.

nl DTD description

There are no attributes associated with the timeout element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT nl (#PCDATA)>
```

nl XML example

For example, in this user profile called UserProfile1 there is a notification list called NL1:

```
<userprofileconfig name="UserProfile1" enable="True" timestamp="0">
   <nl>NL1</nl>
   <console>Console1</console>
   <console>Console2</console>
   <user>User1</user>
   <user>User2</user>
   <user>User3</user>
   <action type="server">Action1</action>
   <action type="client">Action1</action>
</userprofileconfig>
```

nl semantics

When defining a user profile, you are required to provide a notification list. The nl subelement is required and only one is allowed.

Console (console)

The console element is the name of an InCharge console file (by default, the file extension is *.iccon and does not need to be typed in the declaration). The console files are located in *BASEDIR*/smarts/consoles or *BASEDIR*/smarts/local/consoles.

console DTD description

There are no attributes associated with the console element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT console (#PCDATA)>
```

console XML example

For example, in this user profile called UserProfile1 there are two consoles called Console1 and Console2:

```
<userprofileconfig name="UserProfile1" enable="True" timestamp="0">
   <nl>NL1</nl>
   <console>Console1</console>
   <console>Console2</console>
   <user>User1</user>
   <user>User2</user>
   <user>User3</user>
   <action type="server">Action1</action>
   <action type="client">Action1</action>
</userprofileconfig>
```

console semantics

The console subelement is required and multiple consoles are allowed.

User (user)

The user element is the name of user defined by the userconfig element. "User (userconfig)" on page 238 provides additional information.

user DTD description

There are no attributes associated with the user element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT user (#PCDATA)>
```

user XML example

For example, in this user profile called UserProfile1 there are three users called User1, User2, and User3:

```
<userprofileconfig name="UserProfile1" enable="True" timestamp="0">
   <nl>NL1</nl>
   <console>Console1</console>
   <console>Console2</console>
   <user>User1</user>
   <user>User2</user>
   <user>User3</user>
   <action type="server">Action1</action>
   <action type="client">Action1</action>
</userprofileconfig>
```

user semantics

This subelement is not required but if user(s) are not defined no user will access the profile.

Tool/Action (action)

The action subelement refers to the name of a tool defined by the actionconfig element. "Tool/Action (actionconfig)" on page 232 provides additional information. In addition to the name of the tool, you are required to define the type of tool (server, client, or automatic).

action DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT action (#PCDATA)>
<!ATTLIST action
    type (SERVER | CLIENT | AUTO | server | client | auto |
          Server | Client | Auto) #REQUIRED>
```

Table 54 on page 243 identifies the attribute declarations for the actionconfig element.

Table 54 Attribute-list declarations for the actionconfig element

XML attribute	Description	Possible value	Value required/ default value
type	Indicates the classification of the tool: • Server: tools executed on the same host as the Global Manager. These are displayed in the Tools menu on the Global Console. • Client: tools executed locally by the console. These are displayed in the Tools menu on the Global Console. • Auto: tools executed automatically through escalation policies or the automatic action adapter (sm_adapter). These are not visible in the tools menu on the Global Console.	SERVER Server server CLIENT Client client AUTO Auto auto	Required

action XML example

For example, in this user profile called UserProfile there are two tools defined. The first is a server tool called Action1 and the second is a client tool called Action1:

```
<userprofileconfig name="UserProfile1" enable="True" timestamp="0">
   <nl>NL1</nl>
   <console>Console1</console>
   <console>Console2</console>
   <user>User1</user>
   <user>User2</user>
   <user>User3</user>
   <action type="server">Action1</action>
   <action type="client">Action1</action>
</userprofileconfig>
```

action semantics

If actions(s) are not defined, the user will have no tools available in the tools menu of the Global Console. Multiple actions may be listed.

Console operation (consoleoperation)

The consoleoperation element is the list of console operations associated with a user profile.

consoleoperation DTD description

There are no attributes associated with the consoleoperation element. In the consoleoper-config.dtd, the element is defined as:

```
<!ELEMENT consoleoperation (#PCDATA)>
```

consoleoperation XML example

The following example shows the console operations associated with the training-profile user profile:

```
<userprofileconfig name="training-profile" enable="True">
  <nl>Maintenance</nl>
   <console>NotificationLog</console>
   <user>User1</user>

  <consoleoperation>Browse</consoleoperation>
  <consoleoperation>BrowseDetail</consoleoperation>
  <consoleoperation>ExpandMapNode</consoleoperation>
  <consoleoperation>ShowContainment</consoleoperation>
```

</userprofileconfig>

Escalation policy (policyconfig)

An Escalation Policy is composed of a policy filter and one or more escalation paths. If a policy filter is not defined, all notifications will be passed on to the paths for escalation.

policyconfig DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  policyconfig  (filterconfig | pathconfig)*> <!ATTLIST
   policyconfig
   name   CDATA     #REQUIRED
    timestamp   CDATA          "0"
```

Table 55 on page 244 identifies the attribute declarations and possible values for the policyconfig element.

Table 55 Attribute-list declarations for the policyconfig element

XML attribute	Description	Possible value	Value required/ default value
name	A unique name for the escalation policy.	CDATA	Required.
timestamp	When exporting, the value is the last time the object was modified by any user through the Administration Console or through sm_config. When importing, the timestamp defined in the XML file is compared with the timestamp of the object in the repository. Modifications to an existing object will fail when the import timestamp is later than the existing repository timestamp (unless the force option is used).	CDATA	Not required. Default is "0".

policyconfig XML example

This XML sample describes a policy called Policy-Host The expression filter indicates that all notifications pertaining to the ICIM class of Host will be entered into the policy.

There is one escalation path defined called path1. An expression filter at the path level determines that only notifications with a Trouble Ticket ID are escalated.

The escalation actions open a trouble ticket for the notification after 60 minutes, then follows up with an email after another 30 minutes:

```
<policyconfig name="Policy-Host" timestamp="0">

   <filterconfig type="EXPRESSION">
      <isa>ICIM_Notification</isa>
      <criterion attribute="ClassName">Host</criterion>
   </filterconfig>

   <pathconfig enable="TRUE" name="path1" retire="FALSE">
      <retireTime>0</retireTime> <enableTime>0</enableTime>
      <filterconfig type="EXPRESSION">
         <isa>ICIM_Notification</isa>
         <criterion attribute="TroubleTicketID"> </criterion>
      </filterconfig> <escalationlevel interval="1800" level="0">
      </escalationlevel> <escalationlevel interval="900" level="1">
      <action type="auto">Open Trouble Ticket</action>
      </escalationlevel> <escalationlevel interval="900" level="2">
      <action type="auto">Email Manager</action> </escalationlevel>
   </pathconfig>

   </policyconfig>
```

Filter (filterconfig)

The filterconfig element within the policyconfig element is the filter on the entire policy. This filter determines which notifications enter into the policy.

The filterconfig element within the pathconfig element is a filter on the escalation path. Notifications that enter the policy are filtered again to determine the path on which they will escalate.

“Filter (filterconfig)” on page 228 provides filter element details.

Escalation path (pathconfig)

You may have multiple escalation paths within an escalation policy.

pathconfig DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  pathconfig  (filterconfig | retireTime | enableTime |
   escalationlevel)*> <!ATTLIST pathconfig
   name  CDATA #REQUIRED
    enable (TRUE | FALSE | true | false | True | False)"False"
    retire    (TRUE | FALSE | true | false | True | False)"False" >
```

Table 56 on page 246 identifies the attribute declarations and possible values for the pathconfig element.

Table 56 Attribute-list declarations for the pathconfig element

XML attribute	Description	Possible value	Value required/ default value
name	A unique name for the escalation path.	CDATA	Required.
enable	Indicates whether the escalation path should be enabled when imported into the Global Manager or whether it was enabled at the time of export.	TRUE True true FALSE False false	Not Required. Default is “False”.
retire	Indicates whether the escalation path should be retired when imported into the Global Manager or whether it was retired at the time of export	TRUE True true FALSE False false	Not Required. Default is “False”.

pathconfig XML example

For example, this XML sample defines an escalation path called “Path1:”

```
<pathconfig enable="TRUE" name="path1" retire="FALSE">
   <retireTime>0</retireTime>
   <enableTime>0</enableTime>
   <filterconfig type="EXPRESSION">
      <isa>ICIM_Notification</isa>
      <criterion attribute="TroubleTicketID"> </criterion>
   </filterconfig> <escalationlevel interval="1800"
   level="0">
   </pathconfig>
```

Retire time (retireTime)

This is the time interval, in seconds, for which an escalation path is retired.

retireTime DTD description

There are no attributes associated with the user element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT retireTime (#PCDATA)>
```

retireTime XML example

For example, this XML sample defines an escalation path called Path1 with a retireTime of 5 minutes (300 seconds):

```
<pathconfig enable="TRUE" name="path1" retire="FALSE">
   <retireTime>300</retireTime>
   <enableTime>0</enableTime>
   <filterconfig type="EXPRESSION">
      <isa>ICIM_Notification</isa>
      <criterion attribute="TroubleTicketID"> </criterion>
   </filterconfig>
<escalationlevel interval="1800" level="0">
```

Enable time (enableTime)

This is the duration for an escalation path that is retired.

enableTime DTD description

There are no attributes associated with the user element. In the ics-config.dtd, the element is defined as:

```
<!ELEMENT enableTime (#PCDATA)>
```

enabletime XML example

For example, this XML sample defines an escalation path called “Path1” with an enable interval of 15 minutes (900 seconds):

```
<pathconfig enable="TRUE" name="path1" retire="FALSE">
   <retireTime>0</retireTime>
   <enableTime>900</enableTime>
   <filterconfig type="EXPRESSION">
      <isa>ICIM_Notification</isa>
      <criterion attribute="TroubleTicketID"> </criterion>
   </filterconfig>
<escalationlevel interval="1800" level="0">
```

Escalation level (escalationlevel)

Each Escalation Level includes a set of automatic tools to invoke and a time duration before a notification advances to the next level. As time progresses, a notification escalates to different levels in the path based on the level duration, and invokes tools to handle the notification.

escalationlevel DTD description

In the ics-config.dtd, the element is defined as:

```
<!ELEMENT  escalationlevel (action)*>
<!ATTLIST escalationlevel
   level      CDATA    #REQUIRED
    interval   CDATA    #REQUIRED >
```

Table 57 on page 248 identifies the attribute declarations and possible values for the escalationlevel element.

Table 57 Attribute-list declarations for the escalationlevel element

XML attribute	Description	Possible value	Value required/ default value
level	Level indicates the order of the level. Levels must be consecutive numbers beginning with “0”.	CDATA (0-5)	Required.
interval	The duration of time the notification spends in a level before going on to next level. This includes the time it takes to invoke the action at this level.	CDATA (in seconds)	Required.

escalationlevel XML example

For example, this sample defines two escalation levels:

- Level 0 will open a trouble ticket after a duration of 15 minutes (900 seconds)
- Level 1 will email a manager after a second duration of 15 minutes (900 seconds)

```
</escalationlevel> <escalationlevel interval="900" level="0">
   <action type="auto">Open Trouble Ticket</action>
</escalationlevel> <escalationlevel interval="900" level="1">
   <action type="auto">Email Manager</action>
</escalationlevel>
```

escalationlevel semantics

You can specify that no tools are invoked at a level, but the duration cannot be less than 1 minute.

Map icon configuration

Custom map icons can also be configured using the XML interface. The specification for map icons has its own DTD, map-config.dtd, which is located in the *BASEDIR*/smarts/conf/maps directory.

The map-config.dtd defines the mapgifconfig element.

mapgifconfig element

The mapgifconfig element is the root element of the map-config.dtd. It defines the subelements that specify the configuration for a map icon.

mapgifconfig DTD description

In the map-config.dtd, the element is defined as:

```
<!ELEMENT  map_gifconfig  (map | repos | image)*>
```

The map element is not currently used.

repos element

The repos element specifies a map configuration:

```
<!ELEMENT repos (#PCDATA)>
<!ATTLIST repos
  class  CDATA   #REQUIRED
  class (String | Boolean | Time | Integer | Float | Percentage )
   "String"
>
```

Table 58 on page 249 identifies the attribute declarations for the map element.

Table 58 Attribute-list declarations for the map element

XML attribute	Description	Possible value	Value required/ default value
class	The name of the ICIM class that the icon will represent.	CDATA (string)	Required.
class	The instance name of the ICIM class. If instance name is specified, the image/gif will be applied for that particular instance, otherwise "*" indicates that the image/gif is applied to all instances of that class.	String Boolean Time Integer Float Percentage	Not required. Default is "String" so the column heading text must be defined.

repos XML example

The following example specifies a map icon for the DatabaseServer ICIM class:

```
<map_gifconfig>
    <repos class="DatabaseServer">*</repos>
    <image>DatabaseServer</image>
</map_gifconfig>
```

image element

The image element specifies the name of the icon file, it has no attributes. The icon file must be of GIF format. It is not necessary to specify a filename extension such as .gif. The icon file must be located in the *BASEDIR*/smarts/local/conf/icons directory.

image XML example

The following example specifies the DatabaseServer image for the specified class:

```
<map_gifconfig>
    <repos class="DatabaseServer">*</repos>
    <image>DatabaseServer</image>
</map_gifconfig>
```

Global Manager configuration DTDs

The following DTDs identify the XML elements, attributes, and values for Global Manager configuration.

ics-config.dtd

The ics-config.dtd is located in the *BASEDIR*/smarts/conf/ics directory:

```
<?xml encoding="ISO-8859-1"?>

<!ELEMENT ics_config (nlconfig | actionconfig | userprofileconfig |
   userconfig | policyconfig | consoleoperationconfig |
   consoleoperationgroupconfig | map_gifconfig)*>

<!ELEMENT  nlconfig  (filterconfig | columnheading | userprofile)*>
<!ATTLIST nlconfig
   name   CDATA     #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"TRUE"
    timestamp   CDATA          "0"
>

<!ELEMENT columnheading (#PCDATA)>
<!ATTLIST columnheading
   column CDATA   #REQUIRED
   column (String | Boolean | Time | Integer | Float | Percentage )
   "String"
>
<!ELEMENT userprofile (#PCDATA)>

<!ELEMENT  userprofileconfig (nl | console | user | action |
   consoleoperation)*>
<!ATTLIST userprofileconfig
   name   CDATA     #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"TRUE"
    timestamp   CDATA          "0"
>

<!ELEMENT nl (#PCDATA)>
<!ELEMENT user (#PCDATA)>
<!ELEMENT action (#PCDATA)>
<!ATTLIST action
    type (SERVER | CLIENT | AUTO | server | client | auto | Server |
   Client | Auto) #REQUIRED
>
<!ELEMENT console (#PCDATA)>
```

```
<!ELEMENT  userconfig (userprofile)*>
<!ATTLIST userconfig
   name   CDATA     #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"TRUE"
    timestamp   CDATA          "0"
>

<!ELEMENT  actionconfig (program_name | timeout? | trace | display |
   status_criteria? | context_criteria | user_prompts? | userprofile)*>
<!ATTLIST actionconfig
   name   CDATA      #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"TRUE"
    type(SERVER | CLIENT | AUTO | server | client | auto | Server |
   Client | Auto)#REQUIRED
    timestamp   CDATA          "0"
>

<!ELEMENT trace EMPTY>
<!ATTLIST trace
    trace_value (FALSE | TRUE | true | false | True | False)"FALSE"
>
<!ELEMENT display EMPTY>
<!ATTLIST display
    display_value (FALSE | TRUE | true | false | True | False)"TRUE"
>
<!ELEMENT program_name (#PCDATA)>
<!ELEMENT timeout (#PCDATA)>

<!-- Note: user prompts are only allowed on server-side actions -->
<!ELEMENT user_prompts (one_prompt*)>

<!ELEMENT one_prompt (#PCDATA)>
<!ATTLIST one_prompt name CDATA #REQUIRED>

<!ELEMENT status_criteria (filterconfig?)>
<!ELEMENT context_criteria (filterconfig?)>

<!ELEMENT  filterconfig (isa?, (criterion* | filename))>
<!ATTLIST filterconfig
   type   (ASL | EXPRESSION | asl | expression | Expression)  #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)     "TRUE"
>

<!ELEMENT isa  (#PCDATA)>
<!ELEMENT filename (#PCDATA)>
<!ELEMENT criterion (#PCDATA)>
<!ATTLIST criterion
   attributeCDATA   #REQUIRED
>

<!ELEMENT  policyconfig  (filterconfig | pathconfig)*>
<!ATTLIST policyconfig
   name   CDATA     #REQUIRED
    timestamp   CDATA          "0"
>

<!ELEMENT  pathconfig  (filterconfig | retireTime | enableTime |
   escalationlevel)*>
<!ATTLIST pathconfig
   name   CDATA     #REQUIRED
    enable(TRUE | FALSE | true | false | True | False)"False"
    retire    (TRUE | FALSE | true | false | True | False)"False"
>

<!ELEMENT retireTime (#PCDATA)>
<!ELEMENT enableTime (#PCDATA)>
```

```
<!ELEMENT  escalationlevel (action)*>
<!ATTLIST escalationlevel
   level      CDATA    #REQUIRED
    interval   CDATA    #REQUIRED
>
<!ENTITY % consoleoper-config SYSTEM "consoleoper-config.dtd">
&consoleoper-config;
<!ENTITY % map-config SYSTEM "../maps/map-config.dtd">
%map-config;
```

consoleoper-config.dtd

The consoleoper-config.dtd is located in the *BASEDIR*/smarts/conf/ics directory:

```
<?xml encoding="ISO-8859-1"?>

<!ELEMENT consoleoperation (#PCDATA)>

<!ELEMENT consoleoperationconfig (#PCDATA)>
<!ATTLIST consoleoperationconfig
    name CDATA #REQUIRED
    enable (TRUE | FALSE | true | false | True | False) "TRUE"
    timestamp CDATA "0"
>

<!ELEMENT consoleoperationgroupconfig (consoleoperation)*>
<!ATTLIST consoleoperationgroupconfig
    name CDATA #REQUIRED
    displayname CDATA ""
    template (TRUE | FALSE | true | false | True | False) "FALSE"
    enable (TRUE | FALSE | true | false | True | False) "TRUE"
   timestamp CDATA "0" >
```

map-config.dtd

The map-config.dtd is located in the *BASEDIR*/smarts/conf/maps directory:

```
<?xml encoding="ISO-8859-1"?>

<!ELEMENT  map_gifconfig  (map | repos | image)*>

<!ELEMENT map (#PCDATA)>
<!ATTLIST map
   class  CDATA   #REQUIRED
   class  (String | Boolean | Time | Integer | Float | Percentage )
   "String"
>

<!ELEMENT repos (#PCDATA)>
<!ATTLIST repos
   class  CDATA   #REQUIRED
   class  (String | Boolean | Time | Integer | Float | Percentage )
   "String"
>

<!ELEMENT image (#PCDATA)>
```

INDEX

A

Acknowledged attribute 73, 113
Action Completed 176
action element 242
Action Failed 176
Action Invoked 176
actionconfig 232, 234
actionconfig-sample.xml 32
Adapter Platform
  default settings
    notification lists 191
    server tools 191
    user profile 192
Administration Console
  escalation policy 167
archiving notifications 73, 113
ASL scripts
  auto-action.asl 155
  dxa-user-def.asl 115
  nl-sample-filter.asl 186
attachTime 111
Attribute
  System
    Certification 213
    CreationClassName 212
    Description 212
    DisplayClassName 212
    DisplayName 212
    IsManaged 212
    Location 213
    Model 213
    Name 212
    PrimaryOwnerContact 212
    PrimaryOwnerName 212
    SystemName 212
    Type 213
attribute
  notification 215
    Acknowledged 73, 113, 215
    Active 215
    AuditTrail 215
    Category 215
    Certainty 215
    ClassDisplayName 215
    ClassName 215
    ClearOnAcknowledge 215
    ElementClassName 215
    ElementName 216
    EventDisplayName 216
    EventName 216
    EventState 216
    EventText 216
    EventType 216
    FirstNotifiedAt 216
    Impact 216
    InMaintenance 216
    InstanceDisplayName 216
    InstanceName 216
    IsProblem 216
    IsRoot 217
    LastChangedAt 217
    LastClearedAt 217
    LastNotifiedAt 217
    OccurrenceCount 217
    Owner 217
    Severity 217
    SourceDomainName 217
    SourceEventType 217
    TroubleTicketID 217
    UserDefined1-20 217
Audit Log 73
AutoAcknowledgementInterval 113, 114
auto-action.asl 155
Automated tool 142
  sm_adapter 155
Automatic Tools
  Create 170, 172
automatic tools 162
  creation 166
  invoke 163

B

Broadcast Event State on Restart 71, 203
Broker
  brokerConnect.conf 32
browser tool 145
Business Dashboard viewlet
  configuring security message 40
Business Impact Manager 21

C

Certification attribute 213
child group 126
ciscoworks tool 145
class

    reassign derived icon icon 138
  assign custom map icon 137
  ICIM_ComputerSystem 212
  ICIM_System 212
  instance
    assign custom map icon 137
    reassign derived icon icon 138
  Uncertified 124
  Undiscovered 123
  UnitaryComputerSystem 213
  Unsupported 123

client tools
    stopping 141
clientConnect.conf 32, 197
Column Heading 53
columnheading 230, 231
ColumnName 53
Companion UI vApp 71, 203
console 241
console_operations.xml 33, 192
consoleoper_config.dtd 33
consoleoper-ext.xml 33
Context Criteria 151
context_criteria
    XML 237
create
    automatic tools 170, 172
Create New Group 130
Create Top Level Group 130
CreationClassName attribute 212
criterion 229

D

default configurations
    ics-default.xml 190
Default notification list 56
Default service options
    Global Manager 36
    SAM Adapter Platform Server 36
delete
    escalation policy 175
deleting configurations 199
derived icon 136
    reassign to a Class 138
    reassign to an Instance 138
Description attribute 212
detachTime 111
disable escalation path 169
disabling
    configurations 199
    notification list 55
display 235
DISPLAY environment variable 156
DisplayClassName attribute 212
DisplayName attribute 212
domainconfig 66
DomainSection
    HookScript 67
    MinimumCertainty 66
    SmoothingInterval 67
domaintype
    description 66
    domainname 67
    dxaConfFile 66
    name 66
DTD 250
durable 114
duration
    modify
        escalation path 170
dxa-oi.conf 52
dxa-user-def.asl 115

E

elements
    column heading
        allowable values 231
    context_criteria 237
enable
    escalation path 163, 169, 175
    escalation policy 163
enabletime 247
enabling
    configurations 199
    notification list 55
environment variables
    DISPLAY 156
    SM_DISPLAY 156
    tools 146
environment variables input to tools 146
ESCALATION 176
escalation levels
    insert 172
    modify
        duration 173
        tools 173
    remove 173
    structure 162
    tools 162
escalation path
    disable 169
    disabled
        modify 171
    duration 170
    enable 163, 169
    filtering 163
    level 164
    modify 170
        duration 170
        filter 171
        Retire option 174
    Remove 175
    Retire 173
    Retire option 174
    tools 170
escalation policy
    Administration Console 167
    delete 175
    disabled
        modify 171
    enable 163
    filtering 163
    scheduling 176
    structure 162
escalationlevel 248
Event State
    ACTIVE 110
    INACTIVE 110
    SUSPENDED 110
    UNITIALIZED 110
    WAS_ACTIVE 110

exporting
configurations 198
escalation policies 198
notification lists 198
tools 198
user profiles 198
users 198

F

filename 229
filterconfig 228, 245
filtering
escalation path 163
escalation policy 163

G

Global Console 20
configuring 40
Create New Group 130
Create Top Level Group 130
customizing column headings 55
Delete group 130
displaying groups 126
Group Definition window 130
invoking tools 141
Regroup 130
Save Groups 130
saving remote console 50
Server Tools 140
Tool Output window 141
Tools menu 140
Global Manager 20
auto acknowledge 73
default service options 36
default settings 190
Business Templates 191
Client Tools 192
Domains 191
importing topology 123
reconfigure 134
repository objects 191
restarting 176
tagging 101
underlying domains 20
Global Manager Administration Console 30
accessing 31
layout 31
notification list
Name field 53
notification list configuration 54
Tools
Context 151
DisplayOutput 151
Name 150
Program 150
Status 151
tools 28
Global Manager Configuration DTD 250
Group
Hierarchical group 133
HierarchicalGroup 134
Selective group
Create Top Level Group 130
group 126
child group 126
displaying 126
hierarchical group 127
group data file 133
loading to Global Manager 134
regroup 135
member 126
selective group 127
Create New Group 130
deleting 130
matching criteria 127
priority 128
Regroup option 130
Save Groups command 130
target class 128
top-level 126
Group Definition window 130

H

Hierarchical group 127, 133
hierarchical group
group data file 133
regroup 135
HierarchicalGroup class 134
hook script 67, 115
Host class 124

I

ICIM 20, 108
ICIM_ComputerSystem class 212
ICIM_Notification 215
ICIM_System 212
CreationClassName 212
Description 212
DisplayClassName 212
IsManaged 212
Name 212
PrimaryOwnerContact 212
PrimaryOwnerName 212
SystemName 212
icim_xml.dtd 33
ics_config 226
ics-closetkt 143
ics-config.dtd 33
Service Assurance
XML configuration DTD 190
ics-config-sample.xml 32
ics-default.xml 33, 190
AutoAcknowledgementInterval 114
AutoAcknowlegementInterval 113
DomainSection 66
hook script 67, 115
MinimumCertainty parameter 66
name parameter 66

SmoothingInterval parameter 67
InactiveAutoArchiveInterval 114
ics-opentkt 143
ics-ping-all 144
ics-ping-device 144
ics-ping-interface 143
ics-ping-IP 143
ics-telnet 144
importing
configurations 195
escalation policies 195
notification lists 195
tools 195
user profiles 195
users 195
InactiveAutoArchiveInterval 114
InCharge Manager
defining parameters 66
insert
escalation levels 172
instance
assign a custom map icon 137
reassign derived icon 138
instance_maptype-sample.xml 33
isa 229
IsManaged attribute 212

L

level
escalation path 164
Location attribute 213
log file
starting a new log file 80

M

Maintenance notification list 56
Map icon
assigning to a class 137
assigning to an Instance 137
map_gif.xml 33, 192
map-config.dtd 33
mapgif-sample.xml 32
matching
pattern 220
matching pattern 212
member 126
MinimumCertainty 66
Model attribute 213
Modify
Escalation Path
Filter 171
modify
escalation levels 173
escalation path 170, 171
escalation policy 171
momentary notifications 115

N

Name attribute 212
nl 241
nlconfig 227
nlconfig-sample.xml 32
nl-sample-filter.asl 186
Node class 124
notification 109
acknowledging 73
archiving 73, 113
Audit Log 73, 142
auto acknowledge 73
concepts
life cycle 111
durable 114
momentary 115
naming convention 111
ownership 113
unique name 111
notification adapter
notification list 52
Notification attribute 108
Acknowledged 73, 113
Event State 109
ACTIVE 110
INACTIVE 110
SUSPENDED 110
UNITIALIZED 110
WAS_ACTIVE 110
First Notify 109
Last Change 109
Last Cleared 109
Last Notify 109
OccurrenceCount 114
Owner 73, 113
user defined 115
notification attribute
Count 109
notification list 28, 53
ASL filter 53
creating 54
default 47
default notification list 56
disabling 55
display heading 53
enabling 55
filter 52, 53
nl-sample-filter.asl 186
Maintenance notification list 56
Name parameter 53
parameters 53
purpose of 52
Notification Log
customizing column headings 55
number of worker threads 70, 202

O

open integration
sm_ems 26
SNMP Trap Adapter 25
Syslog Adapter 25
Owner attribute 73, 113

Owner notification attribute 73

P

pathconfig 246
pattern 220
pattern matching 220
pinger tool 145
policyconfig 244
policyconfig-sample.xml 32
PrimaryOwnerContact attribute 212
PrimaryOwnerName attribute 212
priority 128
profileconfig-sample.xml 32
program_name 234

R

Remove
    Escalation Levels 173
remove
    escalation path 175
Report Manager
    reporting tool 145
reporting tool 145
retireTime 247
roll_sam_auditlog command 80
rulesets
    auto-action.asl 155
runcmd_env.sh 32

S

SAM Adapter Platform Server
    default service options 36
sample.xml 33
scheduling
    escalation policy 176
Selective group 127
    deleting 130
selective group
    Create New Group 130
    Create Top Level Group 130
    Group Definition window 130
    matching criteria 127
    priority 128
    Regroup button 130
    save groups 130
    target class 128
serverConnect.conf 32, 197
Service Assurance
    components 20
    configuration files 32
service.data.template 32
sm_adapter
    running automated tools 155
sm_cloneDir.conf 32
sm_config 193, 195, 197, 224
    delete objects 199
    disable objects 199
    enable objects 199
    options 197
SM_DISPLAY 156
sm_ems 26
SM_IP_VERSIONS 77
SM_OBJ_Acknowledged 147
SM_OBJ_Active 147
SM_OBJ_Category 147
SM_OBJ_Certainty 147
SM_OBJ_CLASS_NAME 146
SM_OBJ_ClassDisplayName 147
SM_OBJ_ClassName 147
SM_OBJ_DOMAIN_NAME 146
SM_OBJ_ElementClassName 147
SM_OBJ_ElementName 147
SM_OBJ_EventDisplayName 147
SM_OBJ_EventName 147
SM_OBJ_EventState 147
SM_OBJ_EventText 147
SM_OBJ_EventType 147
SM_OBJ_FirstNotifiedAt 147
SM_OBJ_Impact 148
SM_OBJ_InMaintenance 148
SM_OBJ_INSTANCE_NAME 146
SM_OBJ_InstanceDisplayName 148
SM_OBJ_InstanceName 148
SM_OBJ_IsRoot 148
SM_OBJ_LastChangedAt 148
SM_OBJ_LastClearedAt 148
SM_OBJ_LastNotifiedAt 148
SM_OBJ_Name 148
SM_OBJ_OccurrenceCount 148
SM_OBJ_Owner 148
SM_OBJ_Severity 148
SM_OBJ_SourceDomainName 148
SM_OBJ_TroubleTicketID 148
SM_OBJ_UserDefined 148
SM_POBJ_LEVEL 149
SM_POBJ_PATH 149
SM_POBJ_POLICY 149
SM_REMOTE_USER 146
SM_SERVER_NAME 146
sm_xcmd 156
SmGetEnv 145
SmLaunchPerlScript 145
SmoothingInterval 67
SQL Data Interface 73
Status criteria 152
status_criteria 236
structure
    escalation policy 162
Syslog Adapter 25
system defaults 70, 202
SystemName attribute 212

T

tagging
    DisplayName 102
    Name 102
    notifications 102
    topology elements 102
target class 128

tool configuration 28
Tools
passing data to 146
Server
ics-ping-IP 143
tools 28
automated 140, 142
sm_adapter 155
Client 140, 149
browser 145
ciscoworks 145
pinger 145
Reporting 145
SmGetEnv 145
SmLaunchPerlScript 145
stopping 141
configuring 28
Context parameter 151
Display Output parameter 151
element target
SM_OBJ_CLASS_NAME 146
SM_OBJ_DOMAIN_NAME 146
SM_OBJ_INSTANCE_NAME 146
environment variables 146
escalation levels 162
escalation policy target
SM_POBJ_LEVEL 149
SM_POBJ_PATH 149
SM_POBJ_POLICY 149
input 146
invoking through console 141
Name parameter 150
notification target
SM_OBJ_Acknowledged 147
SM_OBJ_Active 147
SM_OBJ_Category 147
SM_OBJ_Certainty 147
SM_OBJ_ClassDisplayName 147
SM_OBJ_ClassName 147
SM_OBJ_ElementClassName 147
SM_OBJ_ElementName 147
SM_OBJ_EventDisplayName 147
SM_OBJ_EventName 147
SM_OBJ_EventState 147
SM_OBJ_EventText 147
SM_OBJ_EventType 147
SM_OBJ_FirstNotifiedAt 147
SM_OBJ_Impact 148
SM_OBJ_InMaintenance 148
SM_OBJ_InstanceDisplayName 148
SM_OBJ_InstanceName 148
SM_OBJ_IsRoot 148
SM_OBJ_LastChangedAt 148
SM_OBJ_LastClearedAt 148
SM_OBJ_LastNotifiedAt 148
SM_OBJ_Name 148
SM_OBJ_OccurrenceCount 148
SM_OBJ_Owner 148
SM_OBJ_Severity 148
SM_OBJ_SourceDomainName 148
SM_OBJ_TroubleTicketID 148
SM_OBJ_UserDefined 148
Program parameter 150
Server 140
Context criteria 151
ics-closektk 143
ics-opentkt 143
ics-ping-all 144
ics-ping-device 144
ics-ping-interface 143
ics-telnet 144
SM_REMOTE_USER 146
SM_SERVER_NAME 146
Status criteria 152
sm_xcmd 156
Status parameter 151
where to save scripts 149
X Windows 144
topology
group
see group
child group 126
group 126
groups 126
top level 126
Host class 124
importing 32, 123
Node class 124
overlapping IP addresses 101
tagging 101
DisplayName 102
Name 102
notifications 102
topology elements 102
Uncertified class 124
Undiscovered class 123
Unsupported class 123
topology-group.data.template 32
trace 236
traps
SNMP Trap Adapter 25
Type attribute 213

U

Uncertified class 124
underlying domain 20
attachTime 111
detachTime 111
Undiscovered class 123
UnitaryComputerSystem class 213
Certification 213
Location 213
Model 213
Type 213
Vendor 213
Unsupported class 123
user 242
user profile
admin 47
default 47

maint 47
notification list 28
oper 47, 48
saved console 28
userconfig 238
userconfig-sample.xml 32
userprofileconfig 239
users
default 192
utility
sm_ems 26
sm_xcmd 156

W

Web Console
configuring security message 40
weights.conf 32
wildcard 220

X

X Windows
tools 144
XML 231
actionconfig-sample.xml 32
console_operations.xml 33
consoleoper_config.dtd 33
consoleoper-ext.xml 33
definition
element 224
value 225
DTD 250
elements
action 242
actionconfig 232, 234
columnheading 230
console 241
criterion 229
display 235
enabletime 247
escalationlevel 248
filename 229
filterconfig 228, 245
ics_config 226
isa 229
nl 241
nlconfig 227
pathconfig 246
policyconfig 244
program_name 234
retiretime 247
status_criteria 236
trace 236
user 242
userconfig 238
userprofileconfig 239
exporting configurations 198
icim_xml.dtd 33
ics-config.dtd 33
ics-config-sample.xml 32
ics-default.xml 33
importing configurations 195
importing with sm_config 195
instance_maptype-sample.xml 33
mapgif-sample.xml 32
nlconfig-sample.xml 32
policyconfig-sample.xml 32
profileconfig-sample.xml 32
Sample configuration files 224
sample configuration files 196
userconfig-sample.xml 32