



Management User's Guide

Version 6.3, December 2005

IONA, IONA Technologies, the IONA logo, Orbix, Orbix/E, Orbacus, Artix, Orchestrator, Mobile Orchestrator, Enterprise Integrator, Adaptive Runtime Technology, Transparent Enterprise Deployment, and Total Business Integration are trademarks or registered trademarks of IONA Technologies PLC and/or its subsidiaries.

Java and J2EE are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

CORBA is a trademark or registered trademark of the Object Management Group, Inc. in the United States and other countries. All other trademarks that appear herein are the property of their respective owners.

While the information in this publication is believed to be accurate, IONA Technologies PLC makes no warranty of any kind to this material including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. IONA Technologies PLC shall not be liable for errors contained herein, or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

COPYRIGHT NOTICE

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, photocopying, recording or otherwise, without prior written consent of IONA Technologies PLC. No third party intellectual property right liability is assumed with respect to the use of the information contained herein. IONA Technologies PLC assumes no responsibility for errors or omissions contained in this book. This publication and features described herein are subject to change without notice.

Copyright © 2005 IONA Technologies PLC. All rights reserved.

All products or services mentioned in this manual are covered by the trademarks, service marks, or product names as designated by the companies who market those products.

Updated: 23-Jun-2008

Contents

List of Figures **vii**

Preface **ix**

Part I Overview

Chapter 1 Introduction to Orbix Management	3
Orbix Management Tools	4
IONA Administrator Web Console	8
Orbix Management Service	10
IONA Configuration Explorer	11
Orbix Configuration Authority	13
Orbix Management Tasks	14

Part II Managing Applications

Chapter 2 Getting Started with the Web Console	19
Before Starting the Web Console	20
Starting the Web Console	21
Navigating the Web Console	25
IONA Administrator Icons	28
Chapter 3 Managing Applications in the Web Console	31
Monitoring Managed Applications	32
Controlling Managed Servers	36
Managing the ORB Core	40

Chapter 4	Managing Events in the Web Console	45
	Starting the Events Console	46
	Viewing Events	47

Part III Managing Configuration

Chapter 5	Finding Configuration Information	53
	Orbix Configuration Authority	54
	Starting the Configuration Authority	56
	Viewing Configuration Information	58

Chapter 6	Managing Configuration Settings	61
	IONA Configuration Explorer	62
	Viewing Configuration Settings	66
	Finding Configuration Settings	70
	Modifying Configuration Settings	72
	Creating Configuration Settings	75
	Deleting Configuration Settings	79

Chapter 7	Management Service Configuration	81
	Management Service Configuration	82
	Configuring the Event Log	84
	Configuring Resource Agents	85
	Configuring Event Listeners	86
	Configuring Event History	87
	Configuring Event Filters	88
	Configuring Event Processing	90
	Configuring the Management Service Web Server	92

Part IV Integrating the Management Service

Chapter 9 Enterprise Performance Logging	95
Introduction	96
Configuring Performance Logging	98
Logging Message Formats	104
Chapter 10 SNMP Integration	109
Introduction to SNMP	110
Orbix SNMP Integration	113
Configuring the SNMP Gateway	115
Glossary	119
Index	125

CONTENTS

List of Figures

Figure 1: Management Overview	6
Figure 2: IONA Administrator Web Console	8
Figure 3: IONA Administrator Architecture	9
Figure 4: IONA Configuration Explorer	11
Figure 5: IONA Configuration Explorer Architecture	12
Figure 6: Orbix Configuration Authority	13
Figure 7: The Login Dialog	22
Figure 8: IONA Administrator Web Console	24
Figure 9: The Navigation Tree	25
Figure 10: The Details Pane	26
Figure 11: Viewing Attributes of Managed Servers	26
Figure 12: Refresh Button	27
Figure 13: New Window Button	27
Figure 14: Viewing a Managed Server in the Web Console	34
Figure 15: Setting Attributes in the Web Console	37
Figure 16: Example Operation in the Web Console	38
Figure 17: Invoking an Operation	38
Figure 18: Shutdown Operation	39
Figure 19: Server Load Details	41
Figure 20: The Event Log	42
Figure 21: The Event Log Viewer	43
Figure 22: Events Button	46
Figure 23: Events Console	46
Figure 24: Refresh Button	47
Figure 25: Full Details of an Event	49
Figure 26: Orbix Configuration Authority	54

LIST OF FIGURES

Figure 27: Viewing a Configuration Variable	59
Figure 28: The Configuration View Tab	62
Figure 29: Viewing Configuration Settings	67
Figure 30: Viewing Configuration Scopes and Namespaces	68
Figure 31: Viewing Configuration Variables	69
Figure 32: Example Search	71
Figure 33: Result of Example Search	71
Figure 34: Loaded Configuration Repository Domain	73
Figure 35: Modifying a Configuration Variable	74
Figure 36: Creating a Configuration Scope	76
Figure 37: Creating a Configuration Namespace	77
Figure 38: Creating a Configuration Variable	78
Figure 39: Deleting a Configuration Variable	80
Figure 40: IONA Configuration Explorer	83
Figure 41: IONA Configuration Explorer Example	92
Figure 42: Overview of an Orbix and IBM Tivoli Integration	97
Figure 43: The SNMP Manager–Agent Model.	111
Figure 44: Overview of the SNMP Gateway	114

Preface

Overview

This guide explains how to use the Orbix management tools, and how to configure the Orbix management service. Orbix management tools enable you to manage component-based distributed enterprise applications. Orbix management tools are integrated with IONA's Adaptive Runtime Technology (ART). This enables them to provide seamless management for the Orbix product range. They can be used across different platform and programming language environments.

Help

If you need help with this or any other IONA product, contact IONA at support@iona.com. Comments on IONA documentation can be sent to docs-support@iona.com.

Audience

This guide is aimed at administrators managing distributed enterprise environments, and developers writing distributed enterprise applications. Administrators do not require detailed knowledge of the technology that is used to create distributed enterprise applications.

Related documentation

The Orbix library includes the following related guides:

- *Orbix Management Programmer's Guide*
- *Orbix Administrator's Guide*
- *IONA IBM Tivoli Integration Guide*
- *IONA BMC Patrol Integration Guide*

For information on integrating with HP Openview, see the following links on the HP website:

- [Smart Plug-in for IONA Orbix](#)
- [Administrator's and User's Guide](#)

For the latest version of all IONA product documentation, see the IONA web site:

<http://www.iona.com/docs>

Organization of this guide

This guide is divided into the following parts:

- Part I, Overview, which gives an introduction to IONA Administrator.
 - Part II, Managing Applications, which explains how to use the IONA Administrator Web Console to manage distributed applications.
 - Part III, Managing Configuration, which explains how to use the IONA Configuration Explorer and Configuration Authority. It also explains the management service configuration settings.
 - Part IV, Integrating the Management Service, which explains performance logging, Enterprise Management System integration and SNMP integration.
 - Glossary, which explains the terminology used in this book.
 - Index.
-

Assumptions

This guide assumes that the reader is familiar with the key concepts of IONA Administrator as described in the Introduction, before proceeding to the other parts. Parts II and III are fully self-contained, and neither assumes familiarity with the other.

Document conventions

This guide uses the following typographical conventions:

Constant width	Constant width font represents commands and literal names of items (such as classes, functions, and variables). For example, constant width text might be used for the <code>java it.server</code> command.
<i>Italic</i>	Italic words in normal text represent emphasis and new terms (for example, <i>IONA Administrator Web Console</i>).
<i>Code italic</i>	Italic words or characters in code and commands represent variable values that you must supply; for example: <code><i>install-dir\etc\domains</i></code>
Bold	Bold font is used to represent menu item selections, for example: To start a selected server, chose Control Start from the menu bar.

This guide may use the following keying conventions:

...	Horizontal or vertical ellipses in format and syntax descriptions indicate that material has been eliminated to simplify a discussion.
[]	Brackets enclose optional items in format and syntax descriptions.
{ }	Braces enclose a list from which you must choose an item in format and syntax descriptions.
	A vertical bar separates items in a list of choices enclosed in { } (braces) in format and syntax descriptions.

PREFACE

Part I

Overview

In this part

This part contains the following chapter:

Introduction to Orbix Management

page 3

Introduction to Orbix Management

The Orbix management tools enable you to manage component-based distributed enterprise applications. This chapter introduces these tools and outlines typical administration tasks.

In this chapter

This chapter contains the following sections:

Orbix Management Tools	page 4
IONA Administrator Web Console	page 8
Orbix Management Service	page 10
IONA Configuration Explorer	page 11
Orbix Configuration Authority	page 13
Orbix Management Tasks	page 14

Orbix Management Tools

Overview

The Orbix management tools enable you to manage and configure component-based distributed enterprise applications. They are integrated with IONA's *Adaptive Runtime Technology* (ART). This enables them to provide seamless management of IONA products and any applications developed using those products.

Orbix management tools are not aimed solely at any specific technology (for example, CORBA, J2EE, or Web services), but provides a generic management paradigm that enables the application to be managed without the administrator requiring knowledge of the technology used to create that application.

Orbix management scope

Orbix management tools enable you to manage and configure Orbix products, and distributed applications that have been developed using these products. For detailed information about the Orbix product range, see the IONA web site:

<http://www.iona.com/products>

Assumptions

Orbix management tools do not assume that you are familiar with any of these products. What is required is a basic understanding of distributed applications, regardless of whether they are based on CORBA, J2EE, or Web services. You can use Orbix management tools to manage any Java or C++ system that has been enabled for management.

Components

Orbix management tools include the following main components:

- “IONA Administrator Web Console”.
- “Orbix Management Service”.
- “IONA Configuration Explorer”.
- “Orbix Configuration Authority”.

This guide explains how to use these tools.

IONA Administrator Web Console	The <i>IONA Administrator Web Console</i> provides a web browser interface to the Orbix management tools. It enables you to manage applications and application events from anywhere, without the need for download or installation. It communicates with the management service using HTTP (Hypertext Transfer Protocol), as illustrated in Figure 1 .
Orbix Management Service	The <i>Orbix management service</i> is the central point of contact for accessing management information in a <i>domain</i> . A domain is an abstract group of managed server processes within a physical location. The management service is accessed by both the IONA Administrator Web Console and by the IONA Configuration Explorer. Note: Managed applications can be written in C++ or Java. The same management service process (<code>iona_services.management</code>) can be used by EJB and CORBA (Java and C++) applications.
IONA Configuration Explorer	The IONA Configuration Explorer is a Java graphical user interface (GUI) that enables you to manage your configuration settings. It communicates with your Configuration Repository (CFR) or configuration file using IIOP (Internet Inter-ORB Protocol).
Orbix Configuration Authority	The Orbix Configuration Authority provides a web browser interface to descriptive information about all Orbix configuration settings. You can browse and search for information about Orbix configuration variables in your CFR or configuration file.

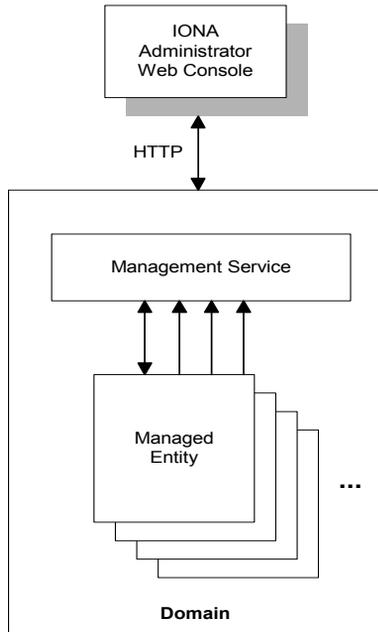


Figure 1: *Management Overview*

Additional features

Application programmers can add instructions to their code to monitor specific components in their system. This is known as adding *management instrumentation*.

In addition, Orbix also provides support for integration *Enterprise Management Systems* (for example, IBM Tivoli).

Adding management instrumentation

IONA products provide default instrumentation that publishes core information to the management service for any application built using these products.

However, programmers might also wish to add custom instrumentation to an application to suit their needs. Orbix therefore enables full instrumentation of server code. For information on how to write instrumentation code, see the *Management Programmer's Guide*.

Integrating with Enterprise Management Systems

Performance logging plugins enable Orbix to integrate effectively with Enterprise Management Systems (EMS), such as IBM Tivoli™, HP OpenView™, CA Unicenter™, or BMC Patrol™.

These systems enable system administrators and production operators to monitor enterprise-critical applications from a single management console. This enables them to quickly recognize the root cause of problems that may occur, and take remedial action. For details of how to configure performance logging, see [Chapter 9](#).

IONA Administrator Web Console

Overview

The IONA Administrator Web Console provides a standard web browser interface to explore and manage distributed applications. The IONA Administrator Web Console uses HTML and JavaScript to create a standard explorer view to represent the data.

Figure 2 shows an example IONA Administrator Web Console interface.

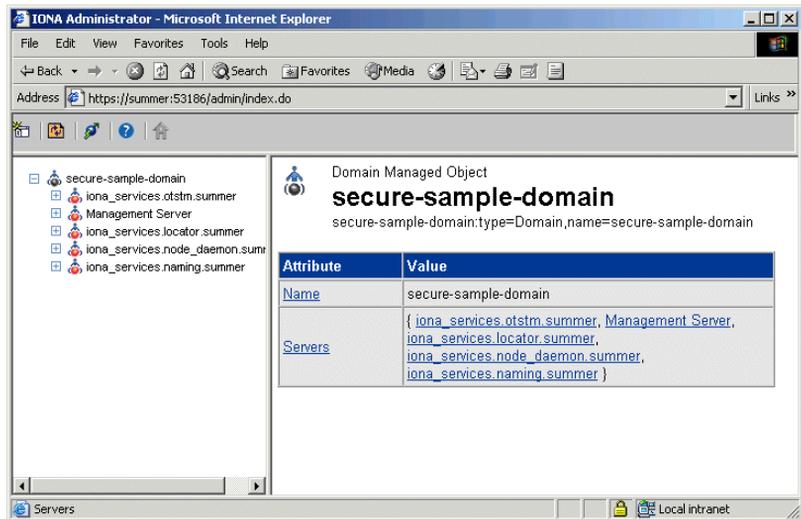


Figure 2: IONA Administrator Web Console

Multiple applications and domains

You can use one instance of the IONA Administrator Web Console to manage multiple applications in a single domain. You also can use multiple instances of the web console to manage multiple domains from a single machine. This is shown in Figure 3.

Interaction with the management service

Each Orbix management service makes management data available using a special URL. The management service is the central point of contact for management information in each domain. It publishes information about all managed servers within its domain.

Management architecture

Figure 3 gives an overview of the management architecture. Each IONA Administrator Web Console interacts with one management service only. This means that each console can administer the managed servers in one of the two domains only.

Multiple instances of the web console can interact with the same management service through the same HTTP port.

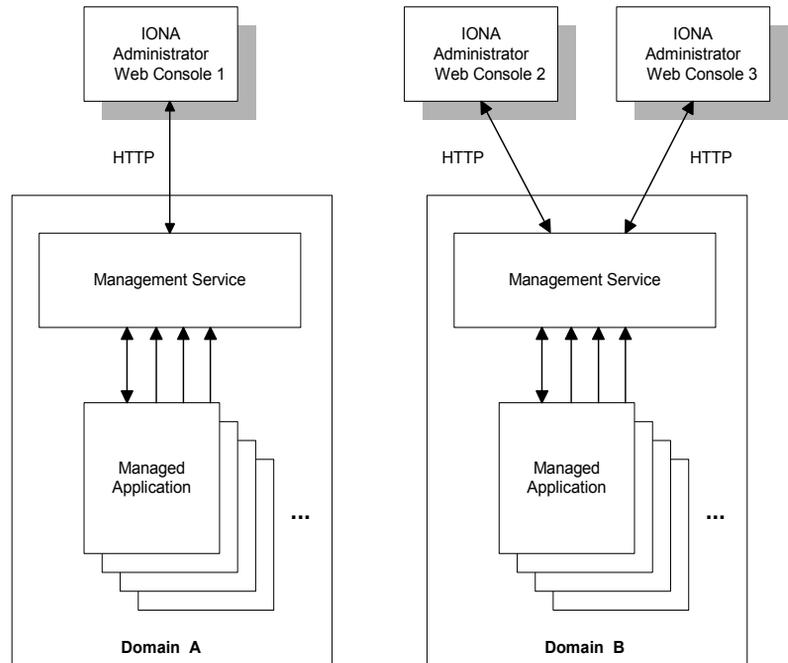


Figure 3: *IONA Administrator Architecture*

Orbix Management Service

Overview

The Orbix management service is the central point of contact for accessing management information in a domain. The management service acts as a buffer between managed applications and management tools.

Management information

The management service maintains key state information, reducing the need to constantly access the managed applications, and thereby improving performance.

The management service stores and publishes information about all managed servers within its domain. It exposes attributes, operations, and events for all managed servers in a domain. The management service also stores information about user roles and passwords for each user in a domain.

Note: Each domain can have only one management service.

Key features

Key features provided by the management service are:

- Centralized repository for all management information.
- Centralized collection of event logging information.
- Persistent storage of event log and agent information.
- Load management gateway plugins (for example, an SNMP plugin).
- Capability to terminate server processes.

For more detailed information, see [Chapter 7](#).

IONA Configuration Explorer

Overview

The IONA Configuration Explorer is an intuitive Java GUI that enables you to view, modify, and search for configuration settings.

In [Figure 4](#), the **Contents** pane on the left shows the configuration scopes and namespaces displayed for a domain named `my-domain`. The **Details** pane on the right displays the configuration variables and their values. Clicking on a icon on the left displays its associated variables on the right.

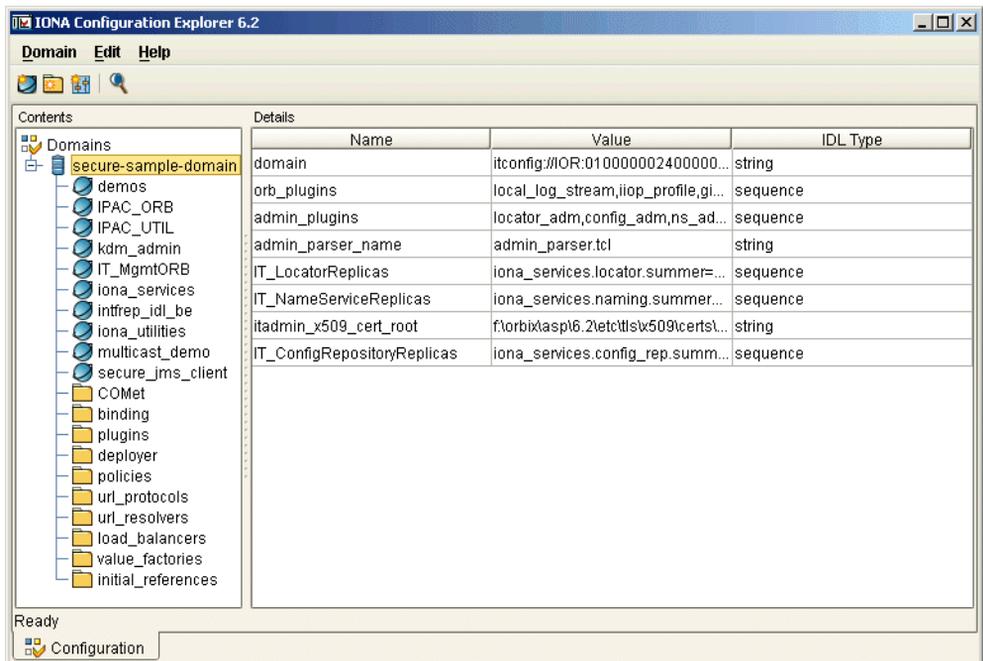


Figure 4: IONA Configuration Explorer

Multiple Domains

You can use a single instance of the IONA Configuration Explorer to manage configuration of multiple domains, both locally and on remote host machines. The IONA Configuration Explorer communicates with CFRs in any domains that it can contact. It can also read file-based domains where they are locally visible.

IONA Configuration Explorer architecture

Figure 5 shows an IONA Configuration Explorer architecture. IONA Configuration Explorer 1 interacts with both a shared CFR-based domain and a local file-based domain, and can therefore manage configuration in either domain. IONA Configuration Explorer 2 only manages the CFR-based domain.

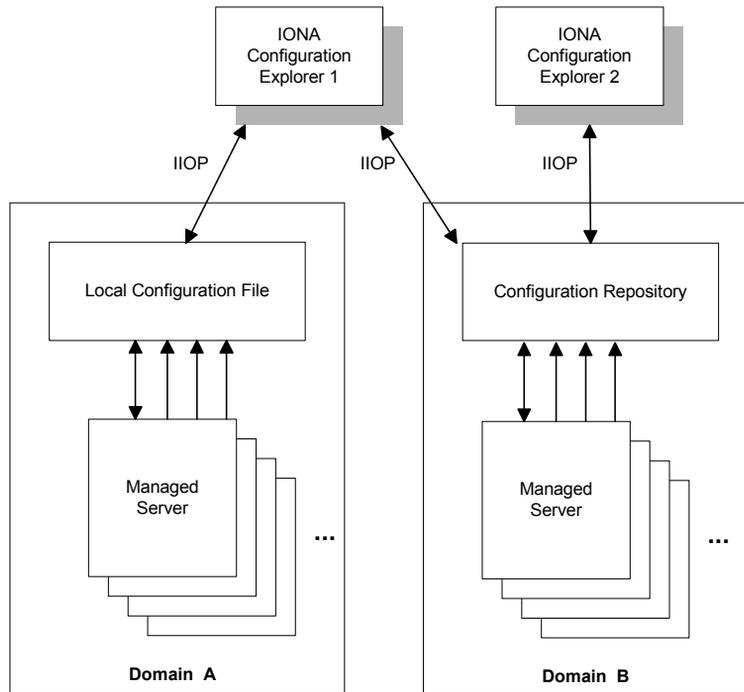


Figure 5: IONA Configuration Explorer Architecture

Orbix Configuration Authority

Overview

The Orbix Configuration Authority displays text descriptions of all Orbix configuration settings. Its web browser interface enables you to navigate to and search for configuration information, as shown in [Figure 5](#).

The navigation tree, on the left of the screen displays a hierarchical list of configuration namespaces and variables. The details pane, on the right, displays information about the configuration variables associated with the selected node on the tree.

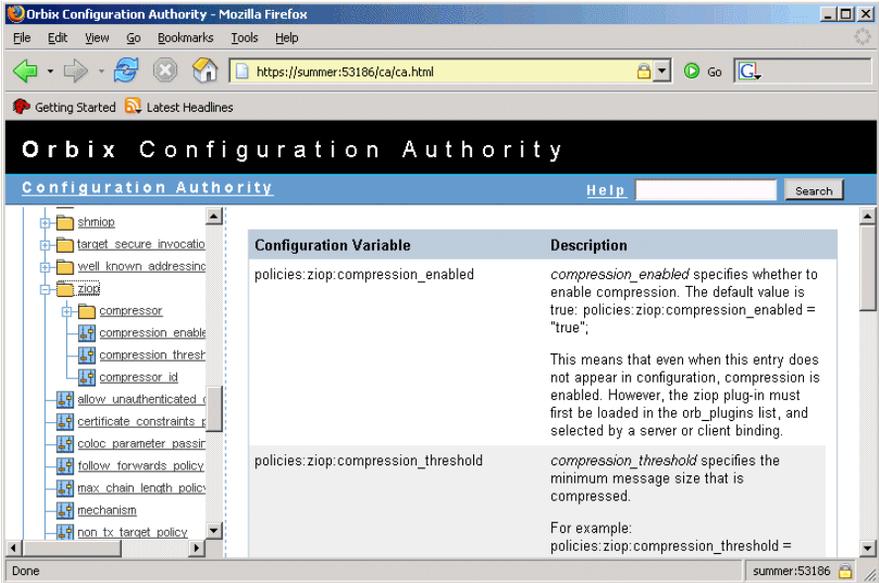


Figure 6: Orbix Configuration Authority

The **Search** box located at the top right of the screen enables you to search for information about configuration variables containing a specified text string.

For more detailed information about the Orbix Configuration Authority, see [Chapter 5](#).

Orbix Management Tasks

Overview

Typical Orbix management tasks that you can perform include:

- [“Managing domains”](#).
- [“Managing servers”](#).
- [“Monitoring events”](#).
- [“Managing configuration settings”](#).
- [“Getting started”](#)

This section gives a quick overview of these tasks, and shows where you can find further information in this book.

Managing domains

Typical domain management tasks include:

- Viewing domains.
- Monitoring domain status (whether it is running or stopped).

[Chapter 3](#) explains how to manage domains using the IONA Administrator Web Console.

Managing servers

Typical server management tasks include:

- Viewing servers.
- Monitoring server status (whether it is running or inactive).
- Controlling servers (shutting down, setting attributes, and invoking operations).

[Chapter 3](#) explains how to manage servers using the IONA Administrator Web Console.

Monitoring events

Typical event management tasks include:

- Selecting a domain in which to manage events.
- Viewing full details of an event.
- Setting event viewing options. For example, you can set the number of events viewed, set the kind of events viewed.

[Chapter 4](#) explains how to manage events using the IONA Administrator Web Console.

Managing configuration settings

Typical configuration management tasks include:

- Loading up a domain.
- Viewing configuration settings.
- Searching your configuration.
- Finding text descriptions of configuration variables.

[Chapter 5](#) explains how to find text descriptions of configuration variables using the Orbix Configuration Authority. [Chapter 6](#) explains how to manage configuration settings using the IONA Configuration Explorer. [Chapter 7](#) explains how to manage configuration settings for the management service.

Getting started

For details of getting started with the IONA Administrator Web Console, see [Chapter 2](#).

Part II

Managing Applications

In this part

This part contains the following chapters:

Getting Started with the Web Console	page 19
Managing Applications in the Web Console	page 31
Managing Events in the Web Console	page 45

Getting Started with the Web Console

This chapter explains how to get started with the IONA Administrator Web Console. It describes how to start and navigate through the web console, and how to get help,

The IONA Administrator Web Console is a standard web browser interface that enables you to explore and manage distributed applications. The IONA Administrator Web Console uses HTML to create a standard explorer view to represent the data. For an overview of the IONA Administrator Web Console, see [Chapter 1](#).

In this chapter

This chapter contains the following sections:

Before Starting the Web Console	page 20
Starting the Web Console	page 21
Navigating the Web Console	page 25
IONA Administrator Icons	page 28

Before Starting the Web Console

Overview

This section explains the requirements for starting the web console. Before starting, you should:

1. “Check your browser version”.
 2. “Ensure your configuration is correct”.
 3. “Ensure your managed server is running”.
-

Check your browser version

The recommended web browsers for use with the web console are:

- Microsoft Internet Explorer 5.5 or later.
- Netscape Navigator 6.0 or later.
- Mozilla Firefox 1.0.

Note: Older versions, or other browsers with similar support for JavaScript, can also be used. However, some visual aids may not be available.

Ensure your configuration is correct

When you have successfully installed and configured Orbix, your system is configured to run the IONA Administrator Web Console.

To ensure that your system is configured correctly, type the following command:

```
install-dir\etc\bin\domain-name_env
```

Ensure Orbix services are running

Before starting the IONA Administrator Web Console, you must ensure that your Orbix services are running.

To run your Orbix services, Use the following command:

```
install-dir\etc\bin\start_domain-name_services
```

Ensure your managed server is running

IONA Administrator enables *runtime* management. This means that any additional instrumented server that you wish to manage (other than the Orbix services) must also be running.

Starting the Web Console

Overview

This section describes how to start the web console from your web browser, and how to log in as an administrator. It includes the following:

- [“Starting in a non-secure domain”](#).
 - [“Starting in a secure \(iSF\) domain”](#).
 - [“Troubleshooting the web console”](#).
 - [“Logging into a non-iSF domain”](#).
 - [“Logging into an iSF domain”](#).
-

Starting in a non-secure domain

To start the IONA Administrator Web Console, type the following URL in the **Address** field of your browser:

```
http://localhost:53185/admin
```

You can start the web console by specifying the address of any management service host in your browser. To start the web console, use the following URL:

```
http://host:port_number/admin
```

The variable *host* is the name or IP address of the host that the domain's management service is running. The variable *port_number* is the port number of the web server configured for this domain. The default port number is 53185.

Example addresses are:

```
http://localhost:53185/admin
http://hamlet.myco.com:53185/admin
http://192.165.146.12:53185/admin
```

Starting in a secure (iSF) domain

To start the IONA Administrator Web Console in a secure (IONA Security Framework) domain, type the following URL in the **Address** field of your browser:

```
https://localhost:53186/admin
```

Accessing the web console over https provides an extra level of security. If you deploy a domain that is either secure or semi-secure, the deployer adds configuration to allow the web console to be accessed over a secure https connection.

Troubleshooting the web console

non-ISF domain The management service requires the following configuration setting:

```
iona_services{
  management{
    policies:well_known_addressing_policy:http:addr_list =
["my-host:53185", "localhost:53185"];
  };
};
```

The variable *my-host* refers to your hostname.

iSF domain The equivalent setting is as follows:

```
iona_services{
  management {
    policies:well_known_addressing_policy:https:addr_list =
["my-host:53186"];
  };
};
```

In this case, you would direct your browser to:

`https://my-host:53186`

The login dialog

When you start up the IONA Administrator Web Console, the **Enter Network Password** dialog appears. This dialog is shown in [Figure 7](#).



Figure 7: *The Login Dialog*

Logging into a non-iSF domain

To log into a domain that does not use iSF (the IONA Security Framework), perform the following steps:

Step	Action
1	In the User Name field, type <code>Administrator</code> .
2	In the Password field, type <code>IONA</code> .
3	Press the OK button.

Note: The **User Name** and **Password** are case sensitive.

Logging into an iSF domain

When logging into an iSF domain, the user name and password are authenticated against the iSF server. If this server is installed to use the default file system provider, perform the following steps:

Step	Action
1	In the User Name field, type: <code>Administrator</code> (for newly deployed Service Pack 1 domains) or <code>IONAAdmin</code> (for existing, pre-Service Pack 1 domains)
2	In the Password field, type: <code>IONA</code> (for newly deployed Service Pack 1 domains) or <code>admin</code> (for existing, pre-Service Pack 1 domains)
3	Press the OK button.

Note: The **User Name** and **Password** are case sensitive.

The IONA Administrator Web Console

When you have logged in, the IONA Administrator Web Console appears in a browser window, as shown in [Figure 8](#).



Figure 8: IONA Administrator Web Console

Navigating the Web Console

Overview

Navigating through the IONA Administrator Web Console is straightforward. The main components of the console are:

- Navigation tree.
- Details pane.
- Toolbar icons.

This section describes how to use these components to navigate through the system.

The navigation tree

The navigation tree on the left of the console is the starting point for exploring distributed applications. It enables you to navigate to the managed servers that you require.

In [Figure 9](#), the navigation tree displays some managed IONA services.



Figure 9: *The Navigation Tree*

Viewing tree node details

The details pane on the right of the console window shows the details of the selected tree node. [Figure 10](#) shows details for the naming service.

Server Managed Object

iona_services.naming.summer

secure-sample-domain:type=Server,name=iona_services.naming.summer,Domain=secure-sample-domain

Attribute	Value
Name	iona_services.naming.summer
Domain	secure-sample-domain
ActiveProcesses	{ iona_services.naming.summer }
State	Running

Figure 10: *The Details Pane*

Viewing the contents of selected nodes

To view the contents of the selected tree node, simply click the node in the navigation tree. This displays the contents in the details pane on the right. As long as there are child nodes in the tree, this display shows a list of the contained child nodes.

Viewing attributes of managed servers

To drill down further into a managed server, click any of the hyperlinks in the **Value** column in the details pane. The hyperlinks in the **Attribute** column show detailed information about the attribute. [Figure 11](#) shows the information displayed for a `ActiveProcesses` attribute.



Figure 11: *Viewing Attributes of Managed Servers*

Refreshing the details pane

To perform a forced refresh on the details pane, use the **Refresh** button on the toolbar, as shown in [Figure 12](#):



Figure 12: *Refresh Button*

Note: Performing a forced refresh overrides any browser caching. Browser caching should always be disabled when using the IONA Administrator Web Console.

Launching a new window

To launch another instance of the web console, click the **New Window** button, as shown in [Figure 13](#):



Figure 13: *New Window Button*

Launching another instance of the web console is very useful when you want to display multiple managed servers simultaneously.

IONA Administrator Icons

Overview

This section explains the icons used in the IONA Administrator Web Console:

- “[Toolbar icons](#)”.
- “[Managed server icons](#)”.

Toolbar icons

The icons used in the web console toolbar are:

Table 1: *Toolbar Icons (Sheet 1 of 2)*

Icon	Description
	New Window Opens a new browser window that runs the IONA Administrator Web Console independently from the current window.
	Refresh Performs a refresh on the Details Pane . This refresh overrides caching.
	Backward Brings you to the previously viewed page.
	Forward Reverses a Backward move.
	Events Console Displays the Events Console in a separate window.

Table 1: *Toolbar Icons (Sheet 2 of 2)*

Icon	Description
	Quick Overview Help Displays the quick help in the Details pane.

Managed server icons

The managed server icons are shown in [Table 2](#):

Table 2: *Managed Server Icons*

Icon	Description
	Orbix managed server object.
	Orbix managed server process.
	Orbix ORB (Object Request Broker).
	Orbix POA (Portable Object Adapter).

Managing Applications in the Web Console

This chapter explains how to use the IONA Administrator Web Console to monitor and control applications in a domain.

The IONA Administrator Web Console displays the manageable components of distributed applications in a single domain. You can use the IONA Administrator Web Console to manage servers within a domain.

In this chapter

This chapter contains the following sections:

Monitoring Managed Applications	page 32
Controlling Managed Servers	page 36
Managing the ORB Core	page 40

Monitoring Managed Applications

Overview

This section explains the key concepts and shows how to monitor managed applications in a domain. It includes the following:

- [“Domains”](#).
 - [“Managed servers and processes”](#).
 - [“Management instrumentation”](#).
 - [“Enabling default instrumentation”](#).
 - [“MBeans”](#).
 - [“Viewing managed servers”](#).
 - [“Attributes of managed servers”](#).
 - [“Drilling into a managed server”](#).
 - [“Monitoring status”](#).
-

Domains

A *domain* is an abstract grouping of components of a distributed application. Typically a domain describes all the components that run on hosts within the same physical location, whether this is the same LAN, the same building, the same town, or the same country.

A domain can contain any number of managed servers. A managed server can be an instance of an EJB application server, or any other registered process.

Note: A domain is equivalent to an Orbix configuration domain.

Managed servers and processes

A *managed server* is a set of replicated managed processes. A managed process is a physical process which contains an ORB and which has loaded the management plugin. The managed server can be an EJB application server, CORBA server, or any other instrumented server that can be managed by IONA Administrator. A domain can contain any number and any types of managed servers.

Management instrumentation

A server process becomes a managed server when it contains core management *instrumentation*. This consists of instructions in the server code that enables management of specific server components.

Developers should see the *Management Programmer's Guide* for details of how to add custom management instrumentation to a server application.

Enabling default instrumentation

By enabling the default management instrumentation on your server, you can display server information in the IONA Administrator Web Console, without adding any custom instrumentation code.

To enable the default instrumentation for your server, set the following configuration variables:

```
plugins:orb:is_managed = true;  
plugins:it_mgmt:managed_server_id:name = your_server_name;
```

Set *your_server_name* to the server name that you want to appear in the IONA Administrator Web Console.

MBeans

An *MBean* is a term used by Java Management eXtensions (JMX) to describe a generic manageable object, a *managed bean*. MBeans are identified by a unique name and can have a number of manageable attributes and operations. All managed entities within a domain are represented by JMX MBeans.

The *Process MBean* is the starting point for navigation through a sever in the IONA Administrator Web Console (`iona_services.otstm.summer` in [Figure 14](#)). It is the first-level MBean that is exposed for management of an application.

Viewing managed servers

You always start monitoring a managed server by selecting it from the navigation tree. [Figure 14](#) shows the initial display for an Object Transaction Service (OTS) managed service.

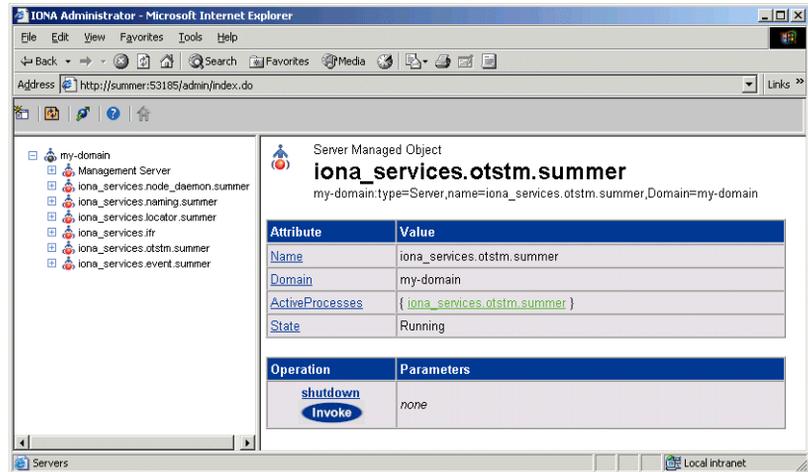


Figure 14: Viewing a Managed Server in the Web Console

Attributes of managed servers

The details pane on the right of the screen shows the attributes of the managed entity represented by the Process MBean. The grey background of the attribute value cells indicate that all attributes are read-only.

A white text box or a drop down list for the attribute value indicates that an attributed can be modified. For more information, see [“Setting attributes” on page 36](#).

Drilling into a managed server

Attribute values that are represented by a hyperlink indicate a reference to other managed entities. For example, [Figure 14](#), shows a reference to the `iona_services.otstm` active process).

Click an icon or hyperlink to open the referenced managed entity. This is also known as *drilling into a managed server*. When you drill into a managed server, each managed entity shows its attributes, and the navigation tree is populated. [Figure 15 on page 37](#) shows the results of drilling into the managed server down to its managed `EventLog`.

Monitoring status

You can use the IONA Administrator Web Console to monitor and control the status of managed servers and domains (for example, to monitor whether a server is active or inactive).

Monitoring status other than status attributes exposed by MBeans is not currently supported by the IONA Administrator Web Console.

Controlling Managed Servers

Overview

Controlling a managed server means the ability modify its attributes, and invoke operations on it. This section includes the following:

- “Managed server attributes”.
 - “Setting attributes”.
 - “Example server attributes”.
 - “Managed server operations”.
 - “Example operations”.
 - “Invoking operations”.
 - “Shutting down servers”.
-

Managed server attributes

You can set attributes in the details pane of the currently displayed managed entity. A text box for the attribute value or a drop-down list indicates that an attribute can be modified. You can get more information on the attribute by clicking on the attribute name in the **Attribute** column.

Setting attributes

To set an attribute, perform the following steps:

Step	Action
1	Select the value field on the right hand side of the details pane (for example, the Filters field in Figure 15).
2	Enter the new value in the text field. Alternatively, for drop-down lists, click the arrow to select one of the values from the list.
3	Click the Set button to apply your changes. You can make changes to multiple attribute values before applying them.

To revert back to the currently active values, click the **Reset** button.

Example server attributes

Figure 15 shows some example attributes for a managed **EventLog**.

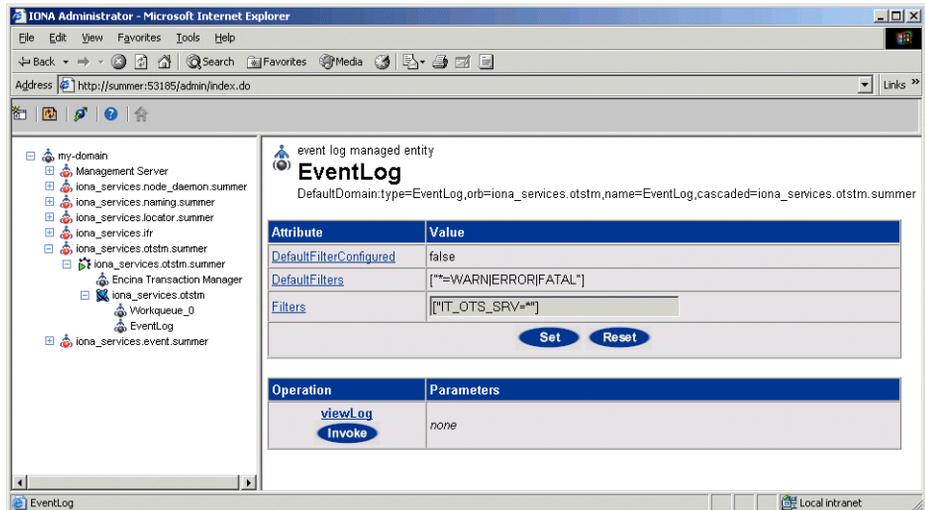


Figure 15: Setting Attributes in the Web Console

Managed server operations

Managed servers can expose one or more operations to the management system. You can then invoke these operations from the console. Operations are represented in the details pane by the following:

- Operation name.
- **Invoke** button.
- Input parameter types.
- Return parameter type.

Example operations

Figure 16 shows an example operation for the **Encina Transaction Manager**. The **Dump** operation takes a transaction log filename as input and returns the contents of the file.



Figure 16: Example Operation in the Web Console

Invoking operations

To invoke an operation, perform the following steps:

Step	Action
1	If the operation takes parameters, type your chosen parameters in the Parameters text box in the details pane. Operations can take a single parameter, multiple, or no parameters.
2	Click the Invoke button in the details pane. Figure 17 shows the dialog displayed for the Update operation.



Figure 17: Invoking an Operation

Shutting down servers

You can shut down any managed server in the console by performing the following steps:

Step	Action
1	In the IONA Administrator navigation tree, select the server that you wish to shut down. In Figure 17 , this is the OTS transaction service.
2	In the details pane, click the Invoke button for the shutdown operation, as shown in Figure 17 .

This shuts down the server, and it will no longer be displayed in the IONA Administrator Web Console.

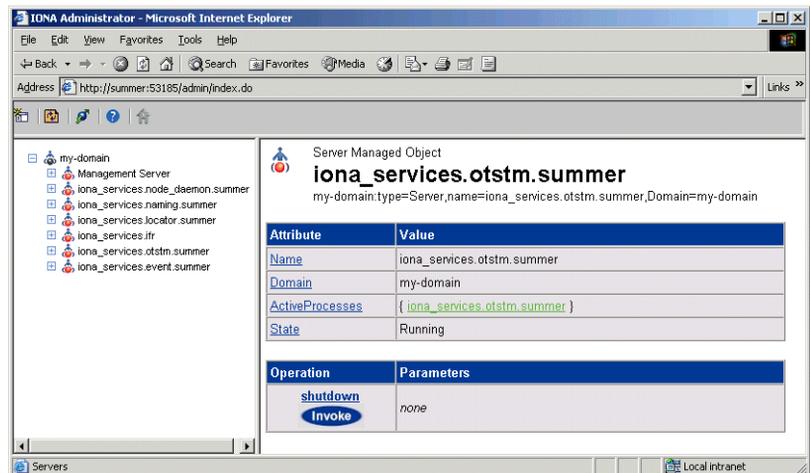


Figure 18: *Shutdown Operation*

Managing the ORB Core

Overview

For many managed servers, you can view their managed ORBs in the navigation tree of the IONA Administrator Web Console. This section explains how to manage various ORB core information. It includes the following:

- “Monitoring the server load”.
 - “Server load attributes”.
 - “Sampling throughput between invocations”.
 - “Managing server logging”.
 - “Viewing an event log”.
 - “Modifying logging filters”.
-

Monitoring the server load

It is important for administrators to know what kind of load key services are under. There are many possible ways of estimating the load on a server, but two key indicators are:

- the number of threads used by the ORB’s automatic work queues.
- the request throughput of the ORB.

You can view this information by clicking on the ORB MBean below the managed process in the navigation tree, as shown in [Figure 19](#).

Server load attributes

The attributes displayed for the server load include the following:

Number of threads. You can view the number of threads that the ORB is currently using in all of its automatic work queues using the `TotalNumberOfThreadsInUse` attribute. This gives an instantaneous snapshot of the number of threads in use.

Thread pool size. Thread pools grow to meet demand and then shrink as demand for threads reduces. The `TotalThreadPoolSize` attribute is therefore a reflection of recent demands on the thread pool.

Thread queue length. The `TotalQueueLength` is an indication of the number of items waiting to be serviced by a thread.

Request throughput. The `IncomingRequests` attribute shows the number of requests initiated on the ORB. The `OutgoingRequests` shows number of requests initiated by the ORB. Refreshing the display enables you to monitor differences in these values, however, it can be difficult to estimate what this means in throughput terms.

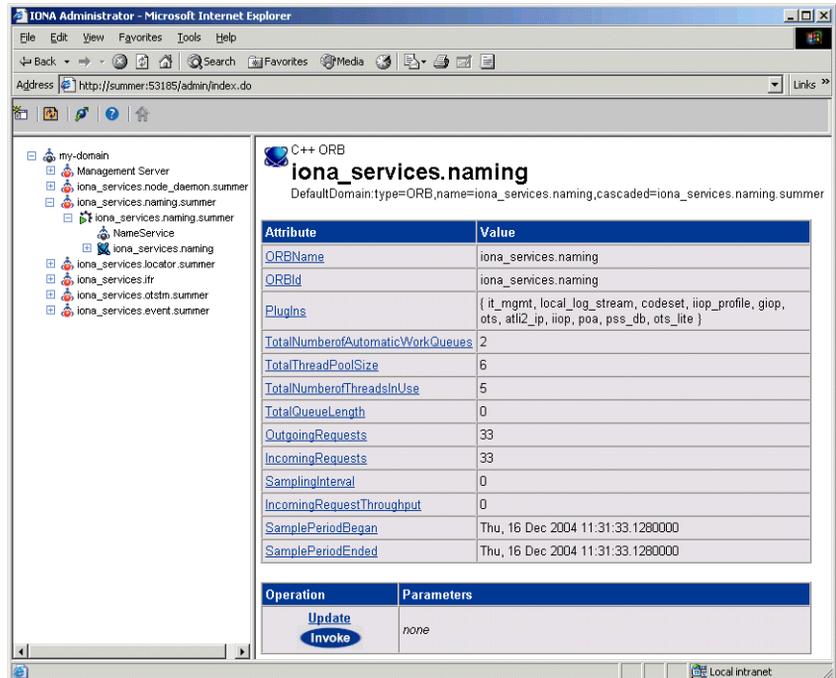


Figure 19: Server Load Details

Sampling throughput between invocations

You can use the **Update** operation to sample the throughput between subsequent invocations. For example, click **Update**, wait for a few seconds, and click **Update** again. The `IncomingRequestThroughput` parameter will indicate the rate of requests per second processed in the interval between the two calls (rounded to the nearest whole number).

The `SamplingInterval` attribute records the length of the sampling period in milliseconds; while the `SamplePeriodBegan` and `SamplePeriodEnded` attributes indicate respectively when the sample period began and ended.

Managing server logging

For Java and C++ servers, the ORB's event log is instrumented. The **EventLog** MBean is displayed in the navigation tree as a child of the ORB MBean, as shown in [Figure 20](#). This enables you to perform the following tasks:

- View the event log.
- Modify the logging filters.

The screenshot shows the IONA Administrator web console. The navigation tree on the left includes the following items:

- my-domain
 - Management Server
 - iona_services.node_daemon.summer
 - iona_services.naming.summer
 - iona_services.locator.summer
 - iona_services.itr
 - iona_services.otstm.summer
 - iona_services.otstm
 - Encina Transaction Manager
 - Workqueue_0
 - EventLog
 - iona_services.event.summer

The main content area displays the **EventLog** MBean configuration page. The title is "event log managed entity" and the JMX path is "DefaultDomain:type=EventLog,orb=iona_services.otstm,name=EventLog,cascaded=iona_services.otstm.summer".

Attribute	Value
DefaultFilterConfigured	false
DefaultFilters	[["WARN ERROR FATAL"]]
Filters	[["*_OTS_SRV+*"]]

Buttons: **Set** **Reset**

Operation	Parameters
viewLog	none

Buttons: **Invoke**

Figure 20: The Event Log

Viewing an event log

To view the event log for a Java or C++ server, perform the following steps:

1. In the navigation tree, click the **EventLog** MBean.
2. In the details view, click the **viewLog** operation. This displays recent logging events for this ORB. [Figure 20](#) shows a naming service log.
3. Click **<Prev** and **Next>** to navigate the log. Clicking next will bring up more recent events). Clicking on **Back to Details** returns you to the main details page for that event log.

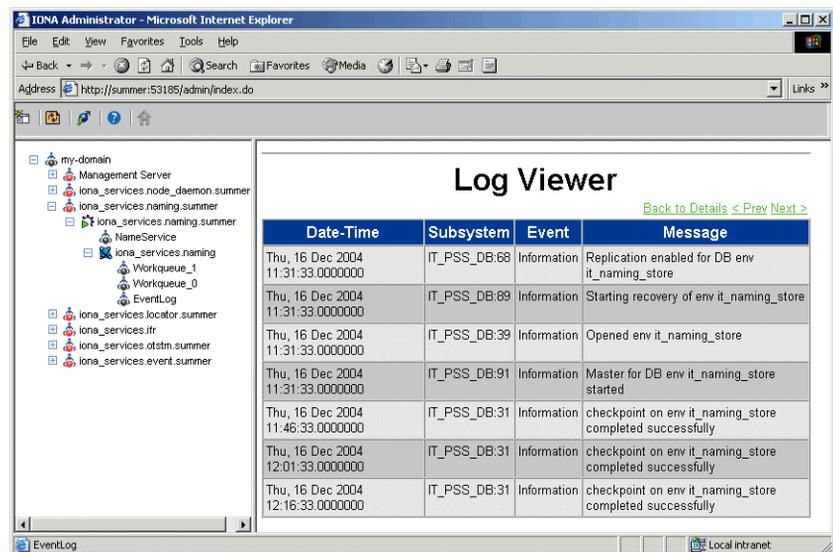


Figure 21: *The Event Log Viewer*

Modifying logging filters

You can change your event log filters dynamically by setting the **Filters** attribute. This applies to both Java and C++ servers. This will have no effect at all if you enter an illegal value. The value must be a list of strings, for example:

```
["IT_ClassLoading=*", "IT_IIOP_TLS=WARN+ERROR+FATAL"]
```

This capability is very useful if your server behaves unexpectedly and you need to turn up the provided level of logging without restarting the server.

Managing Events in the Web Console

This chapter explains how to use the IONA Administrator Web Console to monitor events. It explains how to start its Events Console, and view events for a domain.

The IONA Administrator Web Console's **Events Console** enables you to view events generated by managed servers. The events console shows an up-to-date list of events in reverse chronological order. You can customize the severity of events and apply filters to selectively view events.

In this chapter

This chapter contains the following sections:

Starting the Events Console	page 46
Viewing Events	page 47

Starting the Events Console

Overview

This section explains how to start the IONA Administrator Web Console's **Events Console**.

Using the Events Button

To start the **Events Console**, click the **Events** button in the IONA Administrator Web Console toolbar, as shown in [Figure 22](#).



Figure 22: *Events Button*

If an events console is already open, subsequent clicks on this button bring the web console to the foreground.

Example Events Console

An example **Events Console** started from the web console is shown in [Figure 23](#). The events are shown in a list starting with the most recent event at the top.

Date	Time	Severity	Event Source	Event Name
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	iona_services.management.server	com.iona.management.ProcessRegistered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	iona_services.management.server	com.iona.management.ProcessRegistered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	iona_services.management.server	com.iona.management.ProcessRegistered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered
12/16/04	11:31 AM	Info	iona_services.management.server	com.iona.management.ProcessRegistered
12/16/04	11:31 AM	Info	Management Server	JMX.mbean.registered

Figure 23: *Events Console*

Viewing Events

Overview

This section explains how to use the IONA Administrator Events Console. It includes the following:

- [“Viewing Events in a Domain”](#).
- [“Refreshing the Event List”](#).
- [“Setting the Number of Events Displayed”](#).
- [“Setting the Event Threshold”](#).
- [“Information Displayed in the Event List”](#).
- [“Viewing Full Details of an Event”](#).
- [“Filtering Events”](#).

Viewing Events in a Domain

Events are always shown on a per-domain basis. To view events from a different domain, start a web console connecting to the domain's management service and launch the events console from there. See [“Before Starting the Web Console” on page 20](#) for more details.

Refreshing the Event List

The event display shows an up-to-date list of events when first started. The display is not updated automatically. To refresh the display, click the **Refresh** button in the toolbar, as shown in [Figure 24](#).



Figure 24: *Refresh Button*

Setting the Number of Events Displayed

To set the maximum number of events being retrieved from the management server, click the drop-down box at the **Display Events** field at the top of the console.

Setting the Event Threshold

The **Threshold** setting specifies the lowest severity of events that you want to include in the displayed list. There are four severities:

- Critical
- Error
- Warning
- Info

The highest event severity is `Critical` and the lowest is `Info`.

To set the events threshold, click the **Threshold** drop-down box at the top left of the console.

Information Displayed in the Event List

The event list shows the following summary information about each event:

- Date and time of the event.
 - Severity of the event.
 - Agent that created the event.
 - Name of the event.
-

Viewing Full Details of an Event

To get comprehensive details of a particular event, simply click the event in the event list. [Figure 25](#) shows displays full details for a an event from an example Bank application. This application sends a `ManagediBankAuthorisation.LoginFailed` event when a user login fails.

Filtering Events

You can also customize the severity of events and apply filters to selectively view events by modifying **shared** filters for a domain. For more information, see [“Management Service Configuration” on page 81](#).

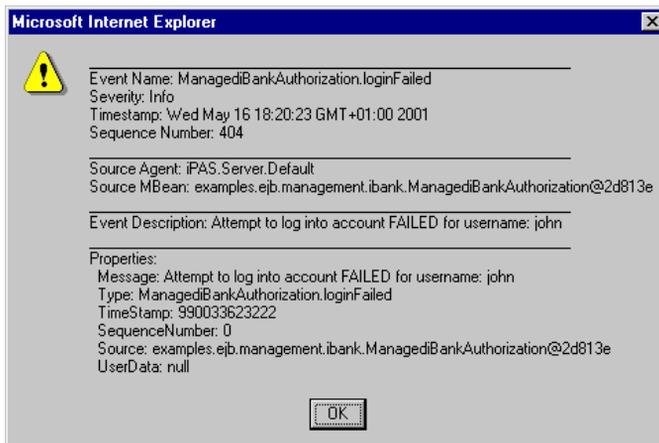


Figure 25: Full Details of an Event

Part III

Managing Configuration

In this part

This part contains the following chapters:

Finding Configuration Information	page 53
Managing Configuration Settings	page 61
Management Service Configuration	page 81

Finding Configuration Information

This chapter explains how to use the Orbix Configuration Authority to find information about Orbix configuration settings.

In this chapter

This chapter contains the following sections:

Orbix Configuration Authority	page 54
Starting the Configuration Authority	page 56
Viewing Configuration Information	page 58

Orbix Configuration Authority

Overview

This section introduces the Orbix Configuration Authority, shown in [Figure 26](#). The Configuration Authority provides a web browser interface to descriptive information about all Orbix configuration settings. This section includes the following topics:

- “Configuration Authority components”.
- “Configuration Authority icons”.

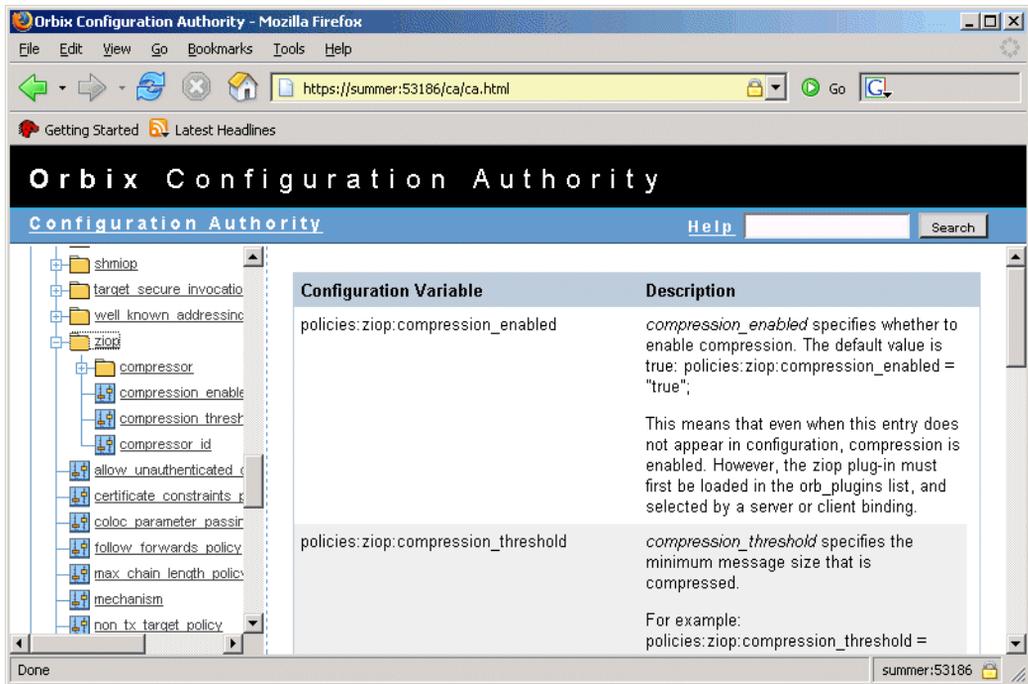


Figure 26: Orbix Configuration Authority

Configuration Authority components

The Orbix Configuration Authority, as shown in [Figure 26](#), is divided into three main areas.

- Navigation tree** The navigation tree, on the left of the screen, is the starting point for exploring for configuration information. This is used to display a hierarchical list of configuration namespaces and variables.
- Details pane** The details pane, on the right of the screen, displays information about the configuration variables associated with the selected node on the tree.
- Search box** The **Search** box located at the top right of the screen enables you to search for information about configuration variables containing a specified text string.

Configuration Authority icons

The icons used in the Orbix Configuration Authority are shown in [Table 3](#).

Table 3: *Configuration Authority Icons.*

Icon	Description
	Configuration namespace.
	Configuration variable.

Starting the Configuration Authority

Overview

This section describes how to start the Orbix Configuration Authority from directly from your web browser. It includes the following:

- [“Starting from your browser”](#).
- [“Troubleshooting”](#).

Note: Before starting the Orbix Configuration Authority, check the steps described in [“Before Starting the Web Console” on page 20](#).

Starting from your browser

To start the Orbix Configuration Authority, type the following URL in the **Address** field of your web browser:

```
http://localhost:53185/ca
```

You can start the web console by specifying the address of any management service host in your browser. To start the web console, use the following URL:

```
http://host:port_number/ca
```

The variable *host* is the name (or IP address) of the host that the domain's management service is running. The variable *port_number* is the port number of the management service configured for this domain. The default port number is 53185.

Example addresses are:

```
http://localhost:53185/ca
http://hamlet.myco.com:53185/ca
http://192.165.146.12:53185/ca
```

In a secure domain, the default address is as follows:

```
https://hostname:53186/ca
```

For for more details, see [“Starting in a secure \(iSF\) domain” on page 21](#).

Troubleshooting

The management service requires the following configuration setting for the web browser used by the Configuration Authority:

```
iona_services{
  management{
    policies:well_known_addressing_policy:http:addr_list =
      ["host:port-number", "localhost:port-number"];
  };
};
```

The variable *my-host* refers to your hostname; *port-number* refers to the management service port number that is configured for your domain.

Viewing Configuration Information

Overview

You can browse for information using the navigation tree and details pane. This section explains the following:

- “Viewing for all variables in a namespace”.
 - “Viewing for a specified variable”.
 - “Searching the Configuration Authority”.
 - “Viewing the entire contents”.
 - “Printing a hard copy”.
-

Viewing for all variables in a namespace

To view information about all the variables contained in a specific namespace, click the namespace folder in the navigation tree.

This displays text descriptions for all the variables in that namespace in the right pane.

[Figure 26 on page 54](#) shows the information displayed for the variables in the `destinations` namespace. This is used to configure the Java Messaging Service (JMS).

Viewing for a specified variable

To view information about a particular variable, drill down to the variable icon in the navigation tree. This displays a text description for the variable in the right pane.

[Figure 27](#) shows the information displayed for the `Comet:Service:NameService` variable. This is used to configure the naming service used to bridge from CORBA to Microsoft COM (Common Object Model).

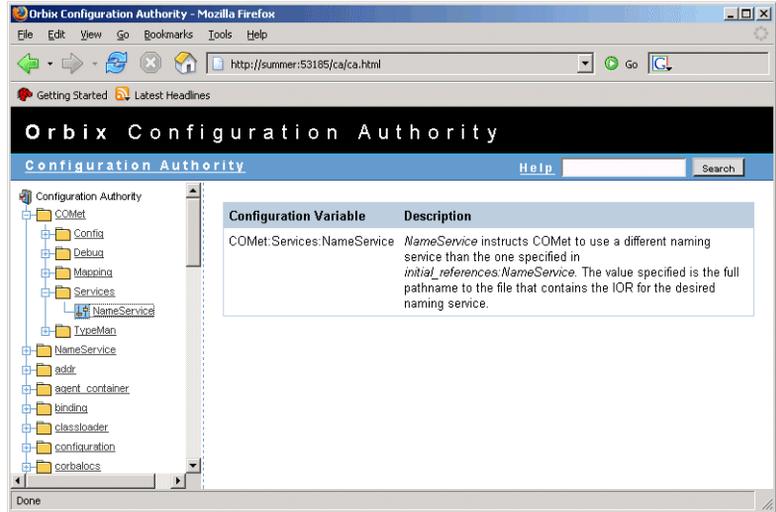


Figure 27: Viewing a Configuration Variable

Searching the Configuration Authority

Use the **Search** box in the top right of the screen to search the Configuration Authority for all variable names that contain your specified search text.

For example, enter the text `classloader` to find all configuration variables that contain the text `classloader` in its name.

Viewing the entire contents

To view the entire contents of the Orbix Configuration Authority, press the **Search** button without typing in any search string. This displays all information about all Orbix configuration variables stored in the Configuration Authority.

Printing a hard copy

Viewing the entire contents of the Configuration Authority displays all the information in a single HTML file, sorted alphabetically by configuration namespace. You can print the result of for a hard copy of all the information in the Configuration Authority.

Note: Ensure that your system is configured to print a Landscape orientation before printing the entire contents.

Managing Configuration Settings

This chapter explains how to use the IONA Configuration Explorer to manage your Orbix configuration settings in multiple domains.

The IONA Configuration Explorer enables you to view, search for, and edit configuration settings. For details of how to find information about specific Orbix configuration settings, refer to [Chapter 5](#). Alternatively, see the *Orbix Configuration Reference*.

In this chapter

This chapter contains the following sections:

IONA Configuration Explorer	page 62
Viewing Configuration Settings	page 66
Finding Configuration Settings	page 70
Modifying Configuration Settings	page 72
Creating Configuration Settings	page 75
Deleting Configuration Settings	page 79

IONA Configuration Explorer

Overview

This section introduces the IONA Configuration Explorer, shown in [Figure 28](#). It includes the following sections:

- “IONA Configuration Explorer components”
- “Contents pane icons”.
- “Toolbar icons”.
- “Starting IONA Configuration Explorer”.

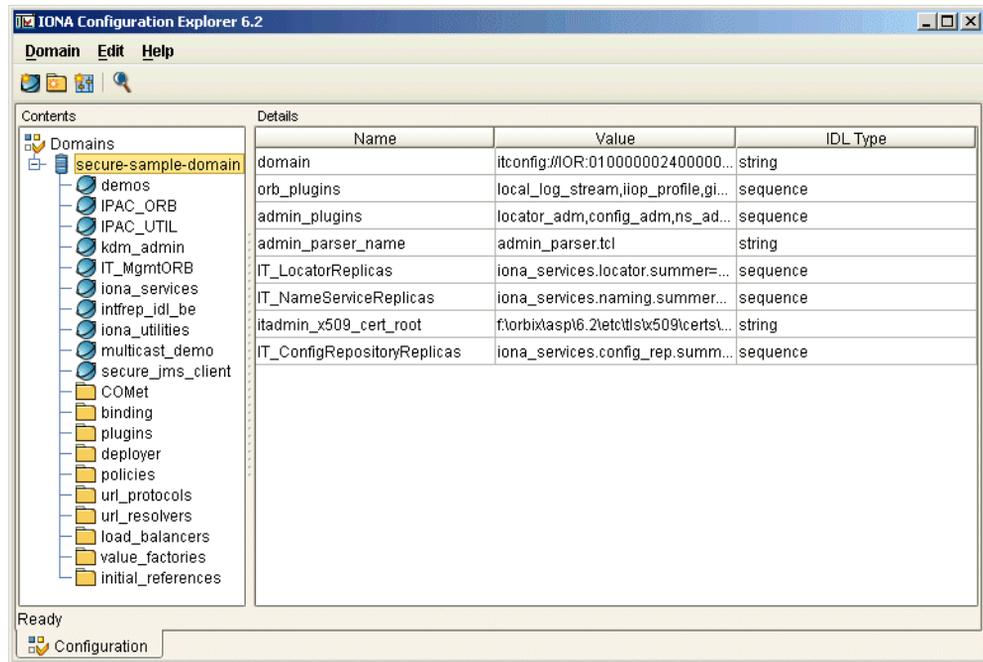


Figure 28: *The Configuration View Tab*

IONA Configuration Explorer components

The IONA Configuration Explorer is divided into three main areas.

Contents pane	The Contents pane, on the left of the screen, is the starting point for exploring your configuration. This is used to display a hierarchical list of configuration domains, scopes, and namespaces.
Details pane	The Details pane, on the right of the screen, displays the configuration variables associated with the selected node on the tree, and also enables you to edit these variables. Figure 28 shows a blank pane because a configuration domain has not been loaded yet.
Menu and Toolbar	The menu and toolbar, located at the top of the screen, enable you to perform various actions in a domain (for example, searching your configuration, or modifying a configuration setting).

Note: In a file-based domain, you can view and find configuration settings. In a configuration repository domain, you can also create, modify and delete configuration settings.

Contents pane icons

The icons used in the **Contents** pane are shown in [Table 4](#).

Table 4: *Navigation Tree Icons.*

Icon	Description
	Unloaded configuration domain.
	File-based configuration domain.
	Configuration repository domain.
	Configuration scope.
	Configuration namespace.

Toolbar icons

The icons used in the IONA Configuration Explorer toolbar are shown in [Table 5](#).

Table 5: *Toolbar Icons.*

Icon	Description
	Create a configuration scope.
	Create a configuration namespace.

Table 5: *Toolbar Icons.*

Icon	Description
	Create configuration variable.
	Find a configuration setting.
	Delete a configuration setting.
	Help.

Starting IONA Configuration Explorer

To start IONA Configuration Explorer, perform the following steps:

1. Change to the following directory:
`install-dir\asp\version\bin\`
2. Enter the following command:
`itconfigexplorer`

Viewing Configuration Settings

Overview

You can view the contents of a domain using the navigation tree and details pane. This section explains the following:

- [“Loading up a domain”](#).
- [“Drilling into the tree”](#).
- [“Viewing Configuration Variables”](#).

Loading up a domain

Before you can view your configuration settings, you must first load up your selected domain in the navigation tree.

To load up a domain, click the domain in the navigation tree. The domain icon in the navigation tree changes to a loaded domain, and the variables in the root configuration scope are displayed in the details pane.

[Figure 29](#) shows a loaded domain in the navigation tree, and the root level settings for this domain in the **Details** pane. In this case, the domain is a local file-based domain named `my-domain`.

Viewing Configuration Scopes and Namespaces

You can view configuration scopes and namespaces in a domain by simply expanding or contracting the loaded domain in the navigation tree.

To expand a domain in the tree, click the + sign on the left. You can also double-click a the domain icon to expand it. For example, [Figure 29](#) shows the result of double-clicking on the `my-domain` icon.

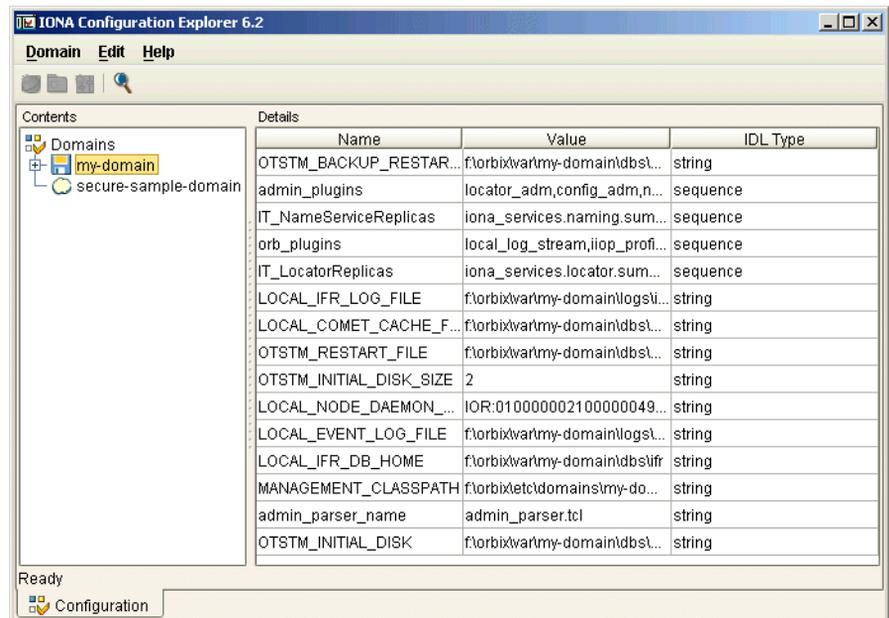


Figure 29: Viewing Configuration Settings

Drilling into the tree

You can view sub-scopes and sub-namespaces by drilling further into the navigation tree. For example, the navigation tree in [Figure 30](#) shows the contents of the `iona_services.management` scope.

Collapsing the tree

To close a tree node, click the `-` sign on the left, or double-click an expanded folder. For example, in [Figure 30](#), double-click `my-domain`.

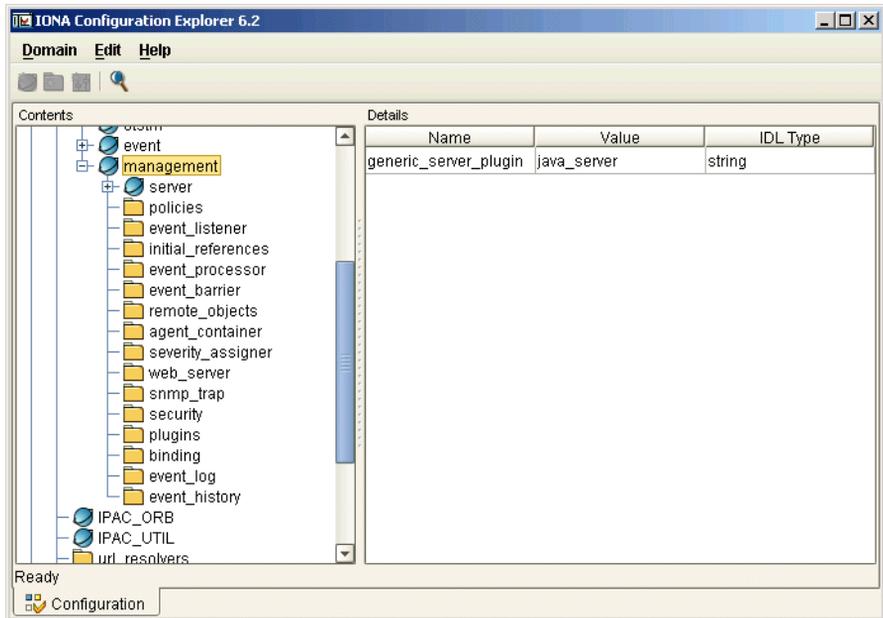


Figure 30: Viewing Configuration Scopes and Namespaces

Viewing Configuration Variables

The details pane on the right of the **Configuration View** displays the configuration variables contained directly within the currently selected scope or namespace.

To view a configuration variable in a domain, navigate to its scope or namespace in the navigation tree. The variable details appear in the details pane. For example, [Figure 31](#) displays the variables contained in the `event_processor` namespace, in the `iona_services:management` scope. The details pane displays the variable name, value, and IDL type.

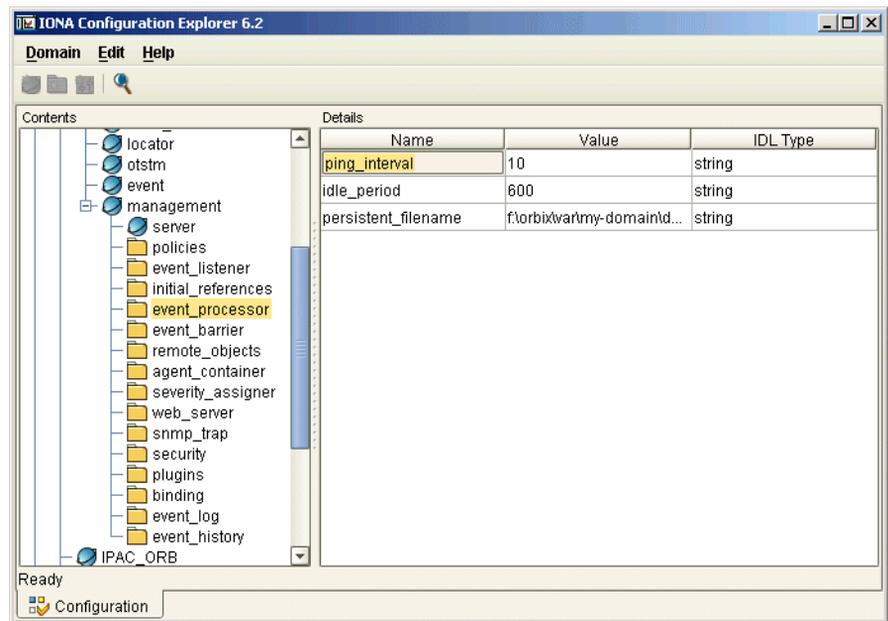


Figure 31: Viewing Configuration Variables

Finding Configuration Settings

Overview

This section explains how to find configuration settings in a loaded domain. It includes the following:

- “Finding a text string in a domain”.
- “Example search”.
- “Performing repeat searches”.

For details of how to load up a domain, see “Loading up a domain” on page 66.

Finding a text string in a domain

To search a domain for occurrences of a particular text string, perform the following steps:

Step	Action
1	Select your domain in the navigation tree.
2	Select Edit Find from the main menu. The Find dialog appears, as shown in Figure 32 . Alternatively, click the Find button in the toolbar: <div style="text-align: center;">  </div>
3	Enter your chosen text in the Find text box. Figure 32 shows a search for the string <code>web_server</code> .
4	Press the Find button. Figure 33 shows the result of this search. The <code>web_server</code> configuration namespace and its configuration variables are displayed.

Example search

Figure 32 shows an example search in the **Find** dialog.



Figure 32: Example Search

Performing repeat searches

To repeat your last search, select **Edit | Find Again** from the main menu. Alternatively, press the **F3** button.

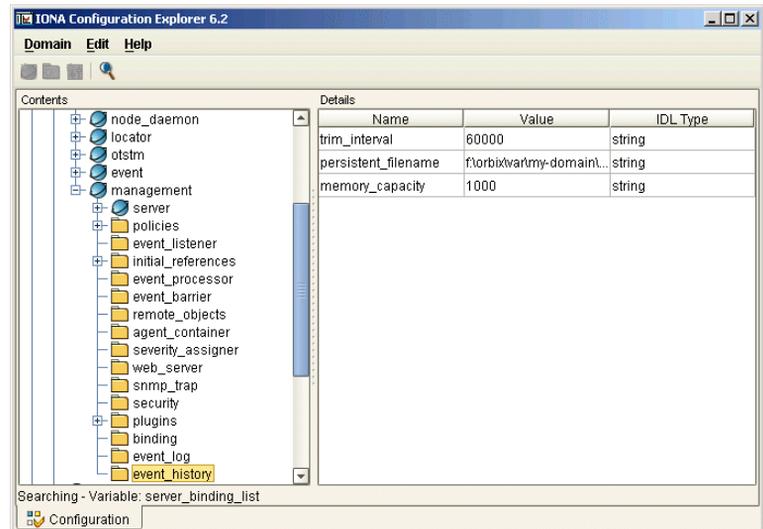


Figure 33: Result of Example Search

Modifying Configuration Settings

Overview

You can modify the contents of a domain using the navigation tree and details pane. This section explains how to modify configuration variable settings. It includes the following:

- “Loading up a domain”.
- “Configuration variable settings”.
- “Modifying configuration variables”.

Note: You can modify settings in a configuration repository (CFR) domain only. You can not use this tool to modify settings in a file-based domain. You should edit your configuration file instead.

The default configuration settings are suitable for most environments. For detailed information about configuration settings, see [Chapter 5](#).

Loading up a domain

Before you can modify your configuration settings, you must first load up your selected domain in the navigation tree.

To load up a domain, click the domain in the navigation tree. The domain icon in the navigation tree changes to a loaded domain, and the variables in the root configuration scope are displayed in the details pane.

[Figure 34](#) shows a loaded configuration repository domain, named `configrep_domain`, in the navigation tree. The root level settings for this domain are shown in the details pane.

Note: Your CFR must first be running before it can load up in the IONA Configuration Explorer.

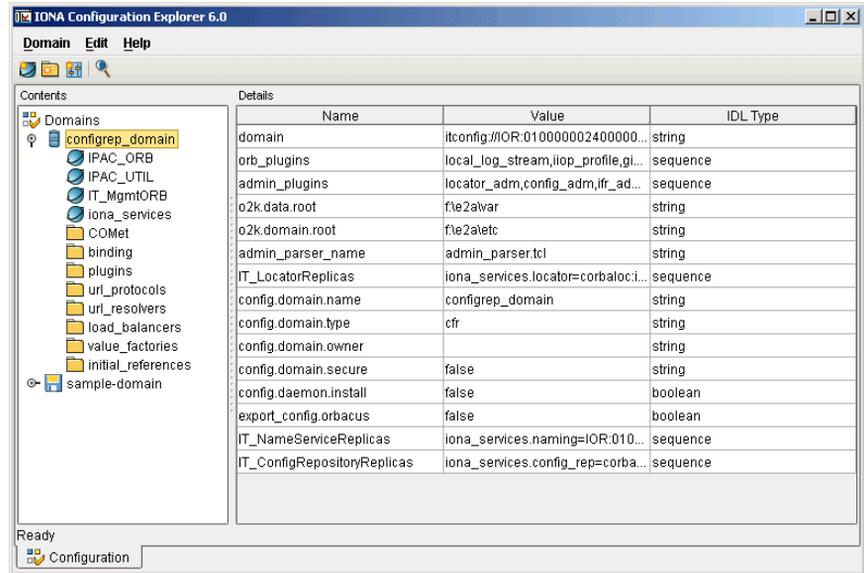


Figure 34: Loaded Configuration Repository Domain

Configuration variable settings

You can modify configuration variable settings in the **Configuration View** details pane. **For each configuration variable, the details pane provides the following text fields:**

- **Name**
- **Value**
- **IDL Type**

Modifying configuration variables To modify configuration variable settings, perform the following steps:

Step	Action
1	Expand your selected domain in the navigation tree.
2	Navigate to the appropriate scope or namespace. The contained variables appear in the details pane.
3	Click the appropriate text field in the details pane. This is shown in Figure 35 . or Click the IDL Type field to make a selection from the drop-down box.
4	Type your variable setting. Figure 35 shows setting a port number for the naming service.

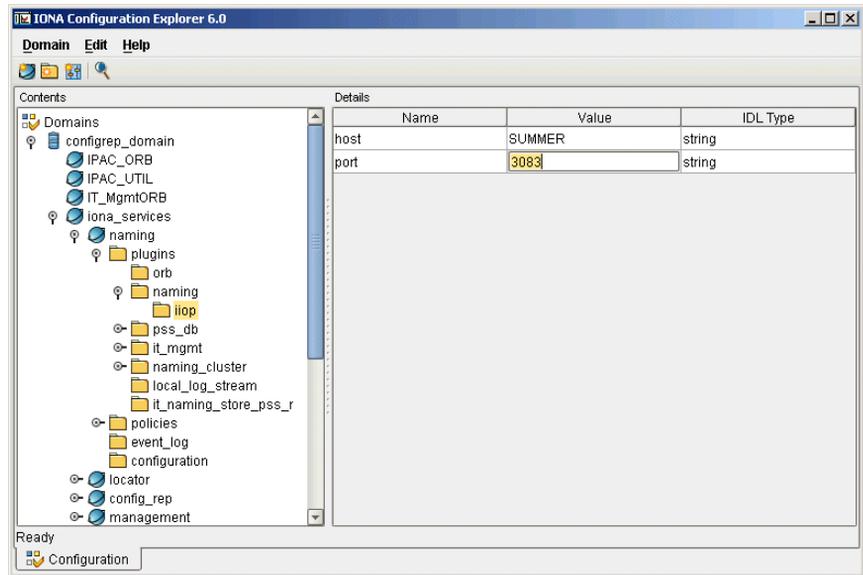


Figure 35: Modifying a Configuration Variable

Creating Configuration Settings

Overview

You can create configuration settings using toolbar icons and the menu bar. This section explains how to create configuration scopes, namespaces, and variables. It includes the following:

- [“Loading up a domain”](#).
- [“Creating scopes”](#).
- [“Creating namespaces”](#).
- [“Creating variables”](#).

Note: You can create settings in a CFR domain only. You can not use this tool to create settings in a file-based domain. You should edit your configuration file instead.

Loading up a domain

Before you can create configuration settings, you must first load up your selected domain in the navigation tree.

To load up a domain, click the domain in the navigation tree. The domain icon in the navigation tree changes to a loaded domain, and the variables in the root configuration scope are displayed in the details pane, as shown in [Figure 34 on page 73](#).

Creating scopes

To create a configuration scope in a domain, perform the following steps:

Step	Action
1	Expand your selected domain in the navigation tree.
2	Navigate to the configuration scope or namespace in which you want to create the scope.
3	Click the Create a new scope button in the toolbar:  Alternatively, you can select Edit New Scope in the main menu.

Step	Action
4	Type your chosen new scope name in the Create a new scope dialog, as shown in Figure 37 .
5	Press Enter .



Figure 36: *Creating a Configuration Scope*

Creating namespaces

To create a configuration namespace, perform the following steps:

Step	Action
1	Expand your selected domain in the navigation tree.
2	Navigate to the configuration scope or namespace in which you want to create the namespace.
3	Click the Create a new namespace icon in the toolbar:  Alternatively, you can select Edit New Namespace in the main menu. Figure 37 shows the newly created namespace.
4	Click the new namespace name to edit it.
5	Type your chosen new namespace name.
6	Press Enter .

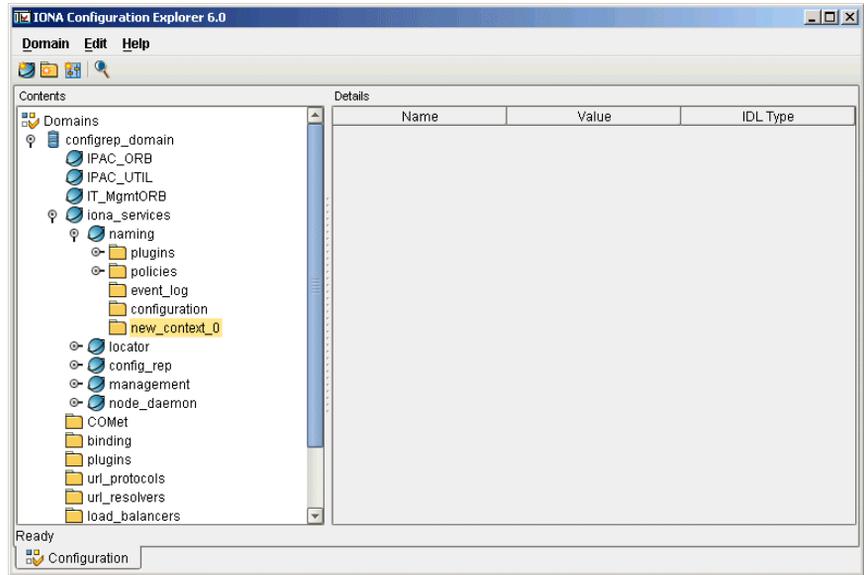


Figure 37: Creating a Configuration Namespace

Creating variables

To create a configuration variable in a domain, perform the following steps:

Step	Action
1	Expand your selected domain in the navigation tree.
2	Navigate to the configuration scope or namespace in which you want to create the variable.
3	Click the Create a new variable icon in the toolbar:  Alternatively, you can select Edit New Variable in the main menu. Figure 38 shows a newly created variable in the root scope of the production domain.

Step	Action
4	<p>Populate the new variable fields in the details pane with settings.</p> <p>Click the two text fields to type your settings. Click the IDL Type field to make a selection from the drop-down box.</p>

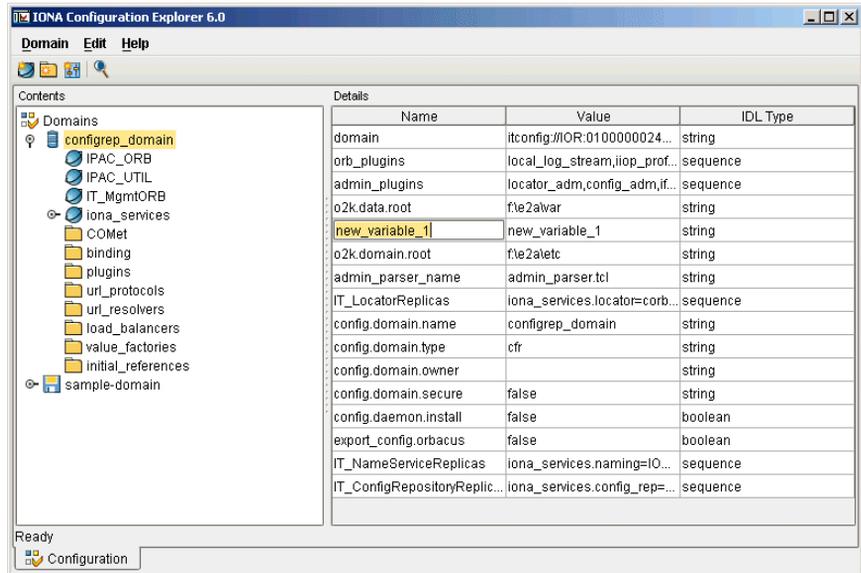


Figure 38: Creating a Configuration Variable

Deleting Configuration Settings

Overview

This section explains how to delete configuration namespaces, and variables. It includes the following:

- [“Deleting scopes and namespaces”](#).
- [“Deleting variables”](#).

Note: You can delete settings in a CFR domain only. You can not use this tool to delete settings in a file-based domain. You should edit your configuration file instead.

The default configuration settings are suitable for most environments. For detailed information about configuration settings, see [Chapter 5](#).

Deleting scopes and namespaces

To delete configuration scopes or namespaces in a domain, perform the following steps:

Step	Action
1	Expand your selected domain in the navigation tree.
2	Navigate to the appropriate configuration namespace.
3	Select Edit Delete in the main menu. Alternatively, you can press the Delete key.

Note: Deleting a namespace deletes all the contained scopes or namespaces.

Deleting variables

To delete configuration variables in a domain, perform the following steps:

Step	Action
1	Expand your selected domain in the navigation tree.
2	Navigate to the appropriate configuration scope or namespace.
3	Click a variable field in the details pane.
4	Select Edit Delete in the main menu. In Figure 39 the variable created in “ Creating a Configuration Variable ” on page 78 has been deleted.

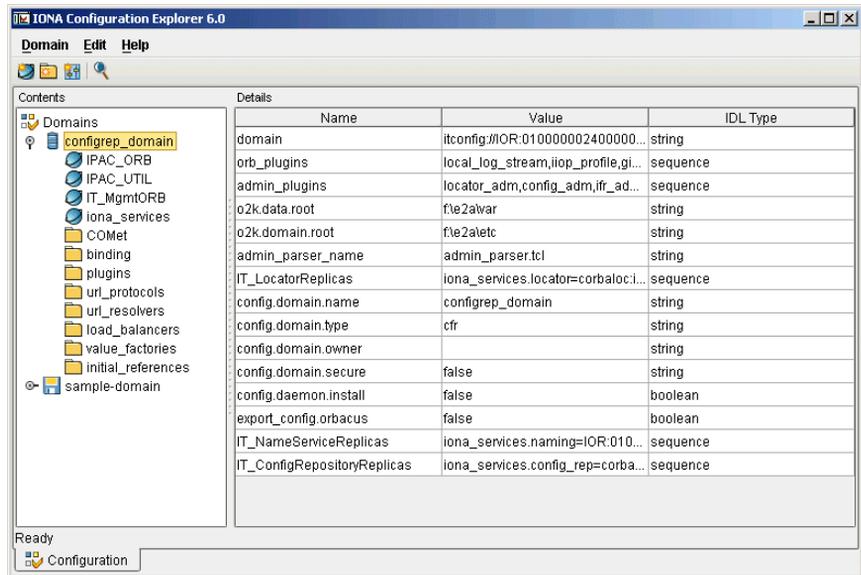


Figure 39: *Deleting a Configuration Variable*

Management Service Configuration

This chapter describes how to configure the Orbix management service. The management service is the central point of contact for management tools accessing managed applications (IONA Administrator Web Console).

In this chapter

This chapter contains the following sections:

Management Service Configuration	page 82
Configuring the Event Log	page 84
Configuring Resource Agents	page 85
Configuring Event Listeners	page 86
Configuring Event History	page 87
Configuring Event Filters	page 88
Configuring Event Processing	page 90
Configuring the Management Service Web Server	page 92

Management Service Configuration

Overview

This section explains how the Orbix management service gets its configuration, and shows an example in the IONA Configuration Explorer. You can configure the Orbix management service by directly editing your configuration file. Configuration files are stored in the `install-dir/etc/domains` directory

Management service scope

The management service gets its configuration from the `iona_services:management` configuration scope in your configuration file. If the configuration variables in the `iona_services:management` scope are not configured correctly, the management service starts up, and sends warnings to the event log and standard error. Depending on the particular variable, a default value is used, or the feature is not enabled (for example, if a persistent filename is not configured, persistent storage is not enabled).

Example configuration file

The following extract from a configuration file shows example configuration variables in the `iona_services:management` scope:

```
management
{
  event_log:filters=["IT_MGMT_SVC=INFO_HI,WARN,ERR,FATAL"];

  plugins:local_log_stream:filename =
    "install-dir/var/domain-name/logs/mgmt_svc.log";
  .
  .
  .
};
```

This chapter explains how to use the `iona_services:management` variables to configure management service features.

IONA Configuration Explorer example

Figure 40 shows `management` configuration variables in the IONA Configuration Explorer.

For information about how to use the IONA Configuration Explorer to manage configuration variables, see [Chapter 6](#).

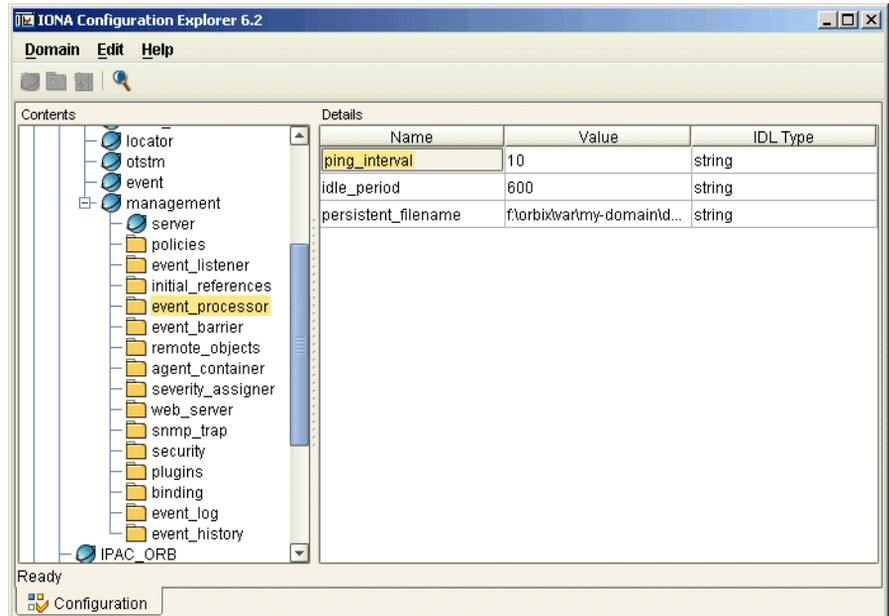


Figure 40: IONA Configuration Explorer

Configuring the Event Log

Overview

This section explains how to enable the event log for management service events, and how to send the event log output to a file.

Configuring the event log filter

To enable event logging for the management service, you must set the required event severities for the `IT_MGMT_SVC` subsystem. You can specify these event severities using the `event_log_filters` configuration variable in the `iona_services:management` configuration scope.

The following example shows a recommended default setting in a configuration file:

```
event_log:filters="{IT_MGMT_SVC=INFO_HI+WARN+ERROR+FATAL}";
```

The following setting enables logging for all management service events:

```
event_log:filters = "{IT_MGMT_SVC=*}";
```

Sending the log output to a file

By default, Orbix logs event messages to a file. To change the location of this file, update the following variable in the `iona_services:management` configuration scope:

```
plugins:local_log_stream:filename =
    "install-dir/var/domain-name/logs/mgmt_svc.log"
```

Configuring the local log stream

When running the management service, if the configured log file does not appear, you might need to add the `local_log_stream` plugin to the `orb_plugins` variable. For example:

```
orb_plugins = ["local_log_stream", "iio_profile", "giop",
    "iio", "ots"];
```

These configuration variables must be set correctly in order for the management service to write to the event log file.

Configuring Resource Agents

Overview

This section describes how to configure resource agent files and resource agent timeouts for the management service.

You must set all configuration variables for the management service in the `management` configuration scope.

Resource agents

Managed server applications register a resource agent reference with the management service. This enables a managed server to be made available for management. Resource agent references are stored in your file system.

A resource agent is an interface to a JMX instrumentation plugin in a managed server. A resource agent is loaded into a managed server to export server MBean information to the management service and management consoles. A resource agent is the sole entry point into a managed server.

Resource agents are registered with the management service, using a unique name, when they are loaded. Resource agents normally unregister when their server terminates; however, abnormal termination can result in references remaining in your system.

Configuring a resource agent file

You can use the following variable to specify the name of the persistent file in which resource agent references are stored:

```
agent_container:persistent_filename =  
    "install-dir/var/domain-name/mgmt/persistent/agents";
```

If an `agents` file is not specified (or if the file can not be created) the persistent storage is disabled. This means that a restarted management service does not know about the registered resource agents.

Configuring resource agent timeouts

You can use the following variable to specify the timeout in seconds to wait between repeatedly pinging all agents to verify that they are running:

```
agent_container:ping_interval = "10";
```

You can specify a timeout value of any number greater than 0 seconds. The default value is 10 seconds.

Configuring Event Listeners

Overview

Client applications requesting events from the management service can register an event listener, so that all events are passed to those clients immediately. This section explains how to configure support for event listeners in the management service.

You must set all configuration variables for the management service in the `iona_services:management` configuration scope.

Configuring an event listener file

Use the following variable to specify the persistent file in which event listener references are stored:

```
event_processor:persistent_filename =  
    "install-dir/var/domain-name/mgmt/persistent/listeners";
```

If a `listeners` file is not specified (or if the file can not be created), the persistent storage is disabled. This means that a restarted management service does not know about the registered event listeners.

Configuring an event listener timeout

Use the following variable to specify the timeout in seconds to wait between repeatedly pinging all event listeners to verify that they are running:

```
event_processor:ping_interval = "10";
```

You can specify a timeout value of any number greater than 0 seconds. The default value is 10 seconds.

Configuring Event History

Overview

This section explains how to configure an event history file, the event history memory capacity, and the interval at which event history is purged. You must set all configuration variable values for the management service in the `management` configuration scope.

Event history

Events received by the management service from managed applications are stored in the file system, so that they can be retrieved later.

You can specify the event history capacity to prevent the system from becoming overloaded. The recommended capacity is 1000 events. You can also specify an interval after which the event history file is purged. The default interval is 10 minutes.

Configuring an event history file

Use the following variable to specify the persistent file in which event history is stored:

```
event_history:persistent_filename =  
    "install-dir/var/domain-name/mgmt/persistent/events";
```

If an `events` file is not specified (or if the file can not be created), the persistent storage is disabled.

Configuring event history memory

Use the following variable to specify the number of events that are stored:

```
event_history:memory_capacity = "1000";
```

The default number of events is 1000. The maximum number of events is 4000. You must specify a positive value.

Configuring event history purges

Use the following variable to specify the interval at which the event history is purged:

```
event_history:trim_interval = "600";
```

The default is 600 seconds (10 minutes). You must specify a positive value.

Configuring Event Filters

Overview

Client applications requesting events from the management service use a named event filter to maintain their own event severity mapping and event threshold values. This section explains how to specify event severity files, and an event threshold file.

Configuring an event severity file

Use the following variable to specify where event severities are stored in the file system:

```
severity_assigner:persistent_filename =  
    "install-dir/var/domain-name/mgmt/persistent/severities";
```

The `severities` file is an internally used archive.

Configuring an event filter threshold file

Use the following variable to specify where event filter thresholds are stored in the file system:

```
event_barrier:persistent_filename =  
    "install-dir/var/domain-name/mgmt/persistent/threshold";
```

The `threshold` file is an internally used archive.

Configuring a default event severity file

You can also specify default event severity mapping using a default severity file. The event severities specified in this file apply to all new filters. The default event severity file is a pure text file, containing event name and event severity pairs, for example `com.ionam.management.testevent 0`

Event severities

Valid event severities are represented by integers in the 0...3 range:

0	CRITICAL
1	ERROR
2	WARNING
3	INFO

Use the following variable to specify where the default event severity file is stored in your file system:

```
severity_assigner:default_filename =  
  "install-dir\var\domain-name\mgmt\persistent\default_severities.txt";
```

Configuring Event Processing

Overview

This section explains how to configure consolidated logging, an idle event period, and a logfile filter.

You must set all configuration variable values for the management service in the `iona_services:management` configuration scope.

Consolidated logging

Successive events with identical names and property values are treated as identical by a consolidated logging feature. This means that such follow-up events are eliminated and reported collectively at a later time (either after some delay, or before the next different event).

You should leave this consolidated event feature enabled, because it protects the system by reducing event numbers.

Enabling consolidated logging

Use the following variable to control consolidated logging:

```
event_listener:consolidated_logging = "true";
```

The default setting is `true`.

Configuring an idle event period

Idle event filters are removed and recreated internally in an identical state for further requests only. Use the following variable to specify the number of seconds an event filter is kept alive in memory:

```
event_processor:idle_period = "600";
```

Configuring the logfile filter

The event log contains a human-readable log of all the management events that have passed through the `logfile` filter (a reserved system filter). The `logfile` filter is used by the management service to generate tab-separated text files.

Use the following variable to specify where these event log text files are stored in your system:

```
event_log:filename_base =  
    "install-dir/var/domain-name/mgmt/logs/events";
```

These filenames are appended with the a timestamp in the standard format `.ddMMyyyy`. For example: `events.08102001`

Configuring the Management Service Web Server

Overview

The IONA Administrator Web Console serves as a web browser interface for HTTP-based access to the management service. This section shows how to configure the web server for this browser interface.

Configuring the web server

The web server's port number is specified by the `web_server:port_number` configuration variable in the `iona_services:management` scope. The default value is 53185. If this variable or port number is not found, the web server is disabled.

IONA Configuration Explorer example

Figure 41 shows a configuration setting for the web server's port number in the IONA Configuration Explorer.

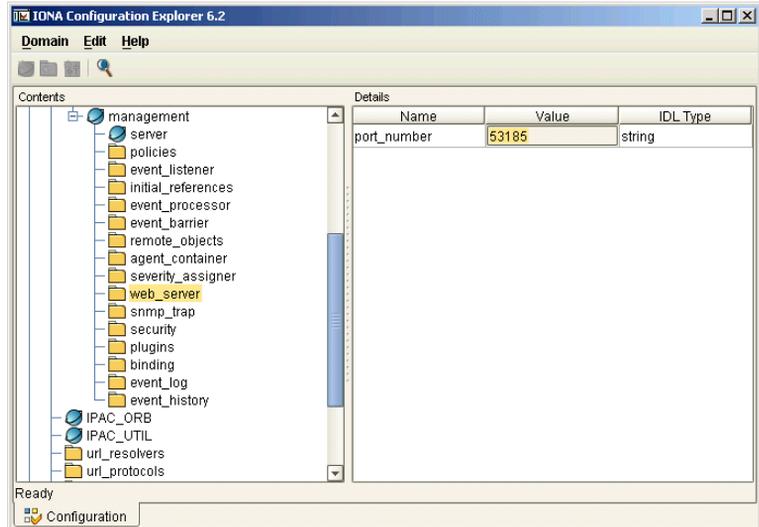


Figure 41: IONA Configuration Explorer Example

Part IV

Integrating the Management Service

In this part

This part contains the following chapters:

Enterprise Performance Logging	page 95
SNMP Integration	page 109

Enterprise Performance Logging

IONA's performance logging plugins enable Orbix to integrate effectively with Enterprise Management Systems (EMS).

In this chapter

This chapter contains the following sections:

Introduction	page 96
Configuring Performance Logging	page 98
Logging Message Formats	page 104

Introduction

Overview

Performance logging plugins enable Orbix to integrate effectively with *Enterprise Management Systems* (EMS), such as IBM Tivoli™, HP OpenView™, CA Unicenter™, or BMC Patrol™. The performance logging plugins can also be used in isolation or as part of a bespoke solution.

Enterprise Management Systems enable system administrators and production operators to monitor enterprise-critical applications from a single management console. This enables them to quickly recognize the root cause of problems that may occur, and take remedial action (for example, if a machine is running out of disk space).

Performance logging

When performance logging is configured, you can see how each Orbix server is responding to load. The performance logging plugins log this data to file or `syslog`. Your EMS (for example, IBM Tivoli) can read the performance data from these logs, and use it to initiate appropriate actions, (for example, issue a restart to a server that has become unresponsive, or start a new replica for an overloaded cluster).

Example EMS integration

[Figure 42](#) shows an overview of the IONA Tivoli integration at work. In this example, a restart command is issued to an unresponsive server.

In [Figure 42](#), the performance log files indicate a problem. The IONA Tivoli Provider uses the log file interpreter to read the logs. The provider sees when a threshold is exceeded and fires an event. The event causes a task to be activated in the Tivoli Task Library. This task restarts the appropriate server.

This chapter explains how to manually configure the Orbix performance logging plugins. It also explains the format of the Orbix performance logging messages.

For details on how to integrate your EMS environment with Orbix, see the IONA guide for your EMS (for example, the *IONA Tivoli Integration Guide*).

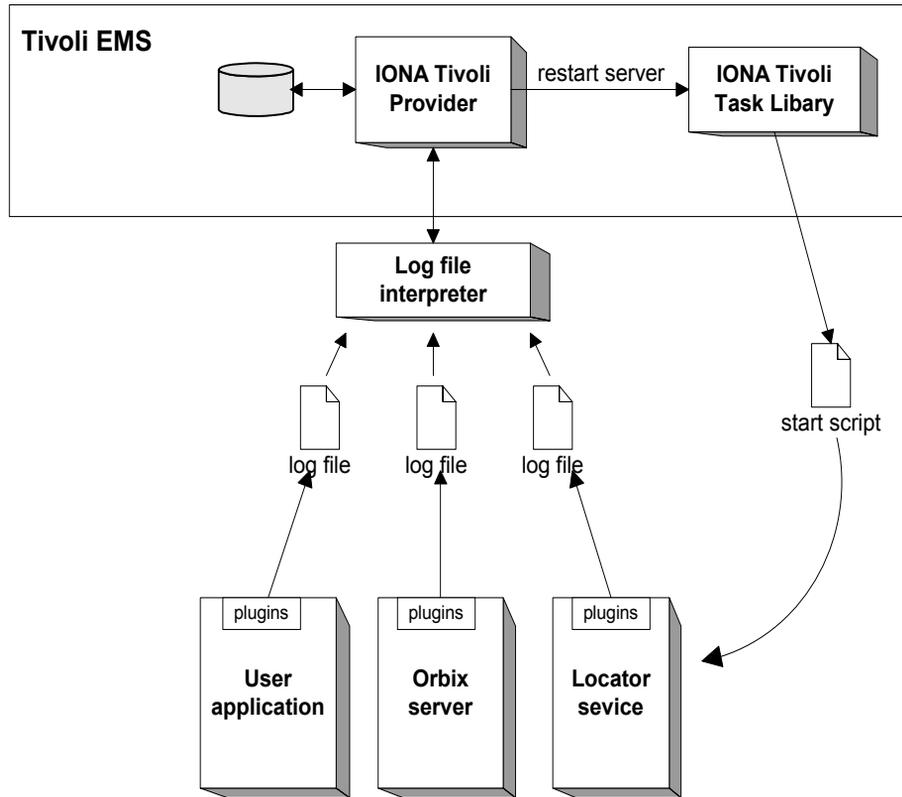


Figure 42: Overview of an Orbix and IBM Tivoli Integration

Configuring Performance Logging

Overview

This section explains how to manually configure performance logging. This section includes the following:

- “Performance logging plugins”.
- “Monitoring Orbix requests”.
- “Logging to a file or syslog”.
- “Monitoring clusters”.
- “Configuring a server ID”.
- “Configuring a client ID”.
- “Monitoring the Orbix work queue”.
- “Configuring performance logging with the GUI”.

Note: You can also use the **Orbix Configuration** GUI (`itconfigure` command) to configure performance logging automatically. The manual configuration gives you more fine-grained control.

Performance logging plugins

The performance logging component consists of three plugins:

Table 6: *Performance logging plugins*

Plugin	Description
Response time logger	Monitors response times of requests as they pass through the Orbix binding chains. This can be used to collect response times for CORBA, RMI-IIOP or HTTP calls.
Request counter	Performs the same function for Artix as the Response time logger does for Orbix.
Response time collector	Periodically harvests data from the response time logger and request counter plugins and logs the results.

Table 6: *Performance logging plugins*

Plugin	Description
MBean monitor	Periodically harvests statistics associated with MBean attributes (for example, monitoring the length of the ORB work queue).

Monitoring Orbix requests

You can use performance logging to monitor both Orbix server and client requests.

Monitoring server requests

To monitor Orbix server requests, perform the following configuration steps:

1. Add `it_response_time_logger` to the servlet binding list for the server you wish to instrument. For example:

```
binding:servlet_binding_list= [
  "it_response_time_logger + it_servlet_context + it_character_encoding
  + it_locale + it_naming_context + it_exception_mapping + it_http_sessions
  + it_web_security + it_servlet_filters + it_web_redirector + it_web_app_activator "
];
```

2. Add `it_response_time_logger` to the server binding list for the server. For example:

```
binding:server_binding_list=[
  "it_response_time_logger+it_naming_context+CSI+j2eecs+OTS+it_security_role_mapping",
  "it_response_time_logger+it_naming_context+OTS+it_security_role_mapping",
  "it_response_time_logger+it_naming_context + CSI+j2eecs+it_security_role_mapping",
  "it_response_time_logger+it_naming_context+it_security_role_mapping",
  "it_response_time_logger+it_naming_context", "it_response_time_logger"
];
```

3. Add `it_response_time_logger` to the `orb_plugins` list for the server.
For example:

```
orb_plugins=[
  "it_servlet_binding_manager", "it_servlet_context",
  "it_http_sessions", "it_servlet_filters", "http",
  "it_servlet_dispatch", "it_exception_mapping", "it_naming_context",
  "it_web_security", "it_web_app_activator",
  "it_default_servlet_binding", "it_security_service", "it_character_encoding",
  "it_locale", "it_classloader_servlet","it_classloader_mapping",
  "it_web_redirector", "it_deployer",
  "it_response_time_logger"
];
```

Monitoring client requests

To monitor Orbix client requests, add `it_response_time_logger` to the client binding list for the server. For example:

```
binding:client_binding_list = [
  "it_response_time_logger+DemoOS+OTS+POA_Coloc", "it_response_time_logger+DemoOS+POA_Coloc",
  "it_response_time_logger+OTS+POA_Coloc", "it_response_time_logger+POA_Coloc",
  "it_response_time_logger+DemoOS+OTS+GIOP+IIOP", "it_response_time_logger+DemoOS+GIOP+IIOP",
  "it_response_time_logger+OTS+GIOP+IIOP", "it_response_time_logger+GIOP+IIOP",
  "it_response_time_logger"
];
```

Logging to a file or syslog

You can configure the collector plugin to log data either to a file or to syslog. The configuration settings for logging to a file depends on whether your application is written in C++ or Java:

C++ configuration

The following example configuration for a C++ application results in performance data being logged to

`/var/log/my_app/perf_logs/treasury_app.log` every 90 seconds:

```
plugins:it_response_time_collector:period = "90";
plugins:it_response_time_collector:filename =
"/var/log/my_app/perf_logs/treasury_app.log";
```

If you do not specify the response time period, it defaults to 60 seconds.

Java configuration

Configuring the Java collector plugin is slightly different from the C++ collector) because the Java collector plugin makes use of Apache Log4J. Instead of setting `plugins:it_response_time_collector:filename`, you set the `plugins:it_response_time_collector:log_properties` to use Log4J, for example:

```
plugins:it_response_time_collector:log_properties = ["log4j.rootCategory=INFO, A1",
"log4j.appender.A1=com.iona.management.logging.log4jappender.TimeBasedRollingFileAppender",
"log4j.appender.A1.File=/var/log/my_app/perf_logs/treasury_app.log",
"log4j.appender.A1.MaxFileSize=512KB",
"log4j.appender.A1.layout=org.apache.log4j.PatternLayout",
"log4j.appender.A1.layout.ConversionPattern=%d{ISO8601} %-80m %n"
];
```

Logging to a syslog daemon

You can configure the collector to log to a syslog daemon or Windows event log, as follows:

```
plugins:it_response_time_collector:system_logging_enabled =
    "true";
plugins:it_response_time_collector:syslog_appID = "treasury";
```

The `syslog_appid` enables you to specify your application name that is prepended to all syslog messages. If you do not specify this, it defaults to `iona`.

Monitoring clusters

You can configure your EMS to monitor a cluster of servers. You can do this by configuring multiple servers to log to the same file. If the servers are running on different hosts, the log file location must be on an NFS mounted or shared directory.

Alternatively, you can use `syslogd` as a mechanism for monitoring a cluster. You can do this by choosing one `syslogd` to act as the central logging server for the cluster. For example, say you decide to use a host named `teddy` as your central log server. You must edit the `/etc/syslog.conf` file on each host that is running a server replica, and add a line such as the following:

```
# Substitute the name of your log server
user.info @teddy
```

Some syslog daemons will not accept log messages from other hosts by default. In this case, it may be necessary to restart the `syslogd` on `teddy` with a special flag to allow remote log messages.

You should consult the `man` pages on your system to determine if this is necessary and what flags to use.

Configuring a server ID

You can configure a server ID that will be reported in your log messages. This server ID is particularly useful in the case where the server is a replica that forms part of a cluster.

In a cluster, the server ID enables management tools to recognize log messages from different replica instances. You can configure a server ID as follows:

```
plugins:it_response_time_collector:server-id = "Locator-1";
```

This setting is optional; and if omitted, the server ID defaults to the ORB name of the server. In a cluster, each replica must have this value set to a unique value to enable sensible analysis of the generated performance logs.

Configuring a client ID

You can also configure a client ID that will be reported in your log messages. Specify this using the `client-id` configuration variable, for example:

```
plugins:it_response_time_collector:client-id = "my_client_app";
```

This setting enables management tools to recognize log messages from client applications. This setting is optional; and if omitted, it is assumed that that a server is being monitored.

Monitoring the Orbix work queue

The `it_mbean_monitoring` plug-in enables you to periodically harvest statistics associated with MBean attributes. This plug-in can be used to monitor the work queue MBean associated with a particular ORB. Work queues are used to control the flow incoming requests.

To monitor an ORB work queue MBean, perform the following steps:

1. Add `it_mbean_monitoring` to the `orb_plugins` list of the ORB whose work queue you wish to monitor.

```
orb_plugins = ["local_log_stream", "iiop_profile", "giop",
              "iiop", "it_mbean_monitoring"];
```

- When `it_mbean_monitoring` is on your `orb_plugins` list, you can enable monitoring of the ORB work queue using the following variable:

```
plugins:it_mbean_monitoring:workqueue = "true";
```

- The MBean attributes that are monitored by the plug-in are sampled periodically. The sampling interval is specified in milliseconds using the following variable:

```
plugins:it_mbean_monitoring:sampling_period = "100";
```

- The response time collector plug-in is used to periodically log the MBean data. You must specify the following variables for the collector:

```
plugins:it_response_time_collector:period = "10";
```

C++ applications

```
plugins:it_response_time_collector:filename = "testing_mbeans.log";
```

Java applications

```
plugins:it_response_time_collector:log_properties =
["log4j.rootCategory=INFO, A1",
"log4j.appender.A1=com.ionamangement.logging.log4jappender.TimeBasedRollingFileAppender",
"log4j.appender.A1.File=Z:\\art\\var\\filedomain\\logs\\mbean_monitoring_perf.log",
"log4j.appender.A1.MaxFileSize=512KB",
"log4j.appender.A1.layout=org.apache.log4j.PatternLayout",
"log4j.appender.A1.layout.ConversionPattern=%d{ISO8601} %-80m %n1"];
```

For more information, see also [“MBean log message formats”](#).

Configuring performance logging with the GUI

The **Orbix Configuration** GUI tool (`itconfigure` command) automatically generates performance logging configuration for the IONA services. The generated `server-id` defaults to the following format:

```
domain-name_service-name_hostname (for example,
mydomain_locator_myhost)
```

For details on how to automatically generate performance logging, see the *IONA Tivoli Integration Guide*, or the *IONA BMC Patrol Integration Guide*.

Logging Message Formats

Overview

This section describes the logging message formats used by IONA products. It includes the following:

- “Orbix log message format”.
- “Artix log message format”.
- “MBean log message formats”.
- “MBean log message formats”.

Orbix log message format

Performance data is logged in a well-defined format. For Orbix applications, this format is as follows:

```
YYYY-MM-DD HH:MM:SS server=serverID [operation=name] count=n
avg=n max=n min=n int=n oph=n
```

Table 7: *Orbix log message format arguments*

Argument	Description
server	The server ID of the process that is logging the message.
operation	The name of the operation for CORBA invocations or the URI for requests on servlets.
count	The number of operations of invoked (IIOP). or The number of times this operation or URI was logged during the last interval (HTTP).
avg	The average response time (milliseconds) for this operation or URI during the last interval.
max	The longest response time (milliseconds) for this operation or URI during the last interval.
min	The shortest response time (milliseconds) for this operation or URI during the last interval.

Table 7: *Orbix log message format arguments*

Argument	Description
int	The number of milliseconds taken to gather the statistics in this log file.
oph	Operations per hour.

Artix log message format

The format for Artix log messages is as follows:

```
YYYY-MM-DD HH:MM:SS server=serverID [namespace=nnn service=sss
port=ppp operation=name] count=n avg=n max=n min=n int=n
oph=n
```

Table 8: *Artix log message format arguments*

Argument	Description
server	The server ID of the process that is logging the message.
namespace	An Artix namespace.
service	An Artix service.
port	An Artix port.

The combination of namespace, service and port above denote a unique Artix endpoint. The description for the remainder of the fields are the same as for Orbix messages.

MBean log message formats

The format for the mbean monitoring log message is as follows:

```
12004-09-23 15:24:17,093 monitored_object=full-object-name-for-mbean
object_alias=user-friendly-name count=n avg=n max=n min=n period=n
```

Table 9: MBean log message format arguments

monitored_object	The MBean being monitored (for example, DefaultDomain:type=AutoWorkqueue,orb=_it_orb_id_1,name=Workqueue_1).
object_alias	A user-friendly name for MBean being monitored (for example, test.management.logging_mbeans.ORBWorkQueue).
count	The number of times the MBean attribute has been sampled during this logging period.
avg	The average value for the attribute being monitored.
max	The maximum value for the attribute being monitored.
min	The minimum value for the attribute being monitored.
period	The sampling interval specified in milliseconds.

Simple life cycle message formats

The server also logs simple life cycle messages. All servers share the following common format.

```
YYYY-MM-DD HH:MM:SS server=serverID status=current_status
```

Table 10: *Simple life cycle message format arguments*

Argument	Description
server	The server ID of the process that is logging the message.
status	A text string describing the last known status of the server (for example, <code>starting_up</code> , <code>running</code> , <code>shutting_down</code>).

SNMP Integration

This chapter describes the Orbix support for SNMP (Simple Network Management Protocol). It introduces basic SNMP concepts, and explains how to set up and configure the Orbix management service for integration with third-party SNMP management tools.

In this chapter

This chapter includes the following sections:

Introduction to SNMP	page 110
Orbix SNMP Integration	page 113
Configuring the SNMP Gateway	page 115

Introduction to SNMP

Overview

Orbix provides support for integration with SNMP (Simple Network Management Protocol). SNMP is the Internet standard protocol for managing nodes on an IP network. Orbix provides a gateway between the IONA Administrator management service and SNMP applications (for example, HP OpenView™). Events received from the management service are converted into SNMP management information. This section introduces the key concepts, and includes the following:

- “SNMP”.
 - “SNMP managers and agents”.
 - “Management Information Base”.
 - “SNMP operations”.
 - “SNMP management tools”.
 - “IONA Administrator SNMP agent”.
-

SNMP

Simple Network Management Protocol (SNMP) is the Internet standard protocol for managing nodes on an IP network. SNMP can be used to manage and monitor all sorts of equipment (for example, network servers, routers, bridges, and hubs). You need to be familiar with some basic SNMP concepts to use IONA Administrator’s SNMP integration.

SNMP managers and agents

The two key components in internet management are *managers* and *agents*. An SNMP manager is a console that enables an administrator to perform network management tasks. An SNMP manager is sometimes also referred to as a *Network Management Station* (NMS).

An SMNP agent is device running software that understands SMNP and interfaces with the actual device being managed. A manager controls an agent by invoking operations on the agent. The SNMP manager–agent model is shown in [Figure 43](#).

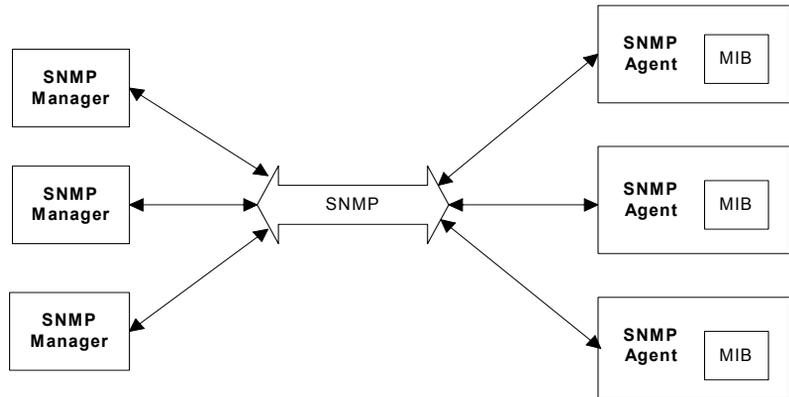


Figure 43: *The SNMP Manager-Agent Model.*

The SNMP protocol specifies how data is transferred between a manager and an agent. It specifies the format and meaning of messages exchanged by the manager and agent.

Management Information Base

Management Information Base (MIB) is the standard that specifies the data managed by a SNMP agent. MIB defines the data that a manager can request from an agent, and the actions permitted on this data.

A *MIB file* is a database of objects that can be managed using SNMP. It has a hierarchical structure, similar to a DOS or UNIX directory tree. It contains both pre-defined values and values that can be customized. A MIB is the most basic element in network management.

SNMP operations

There are five primitive SNMP operations, sometimes referred to as *Protocol Data Units* (PDUs). These are the different kinds of messages that can be sent over an network using SNMP:

- `GetRequest`
- `GetNextRequest`
- `GetResponse`
- `SetRequest`
- `Trap`

A manager can issue `GetRequest`, `GetNextRequest`, and `SetRequest` messages to access single or multiple object variables. A managed agent can send a `GetResponse` message to complete the `GetRequest`, `GetNextRequest`, or `SetRequest`.

An agent can also send an event notification, called a `Trap`, to a manager in order to notify the occurrence of specific events (for example, a system exception, or when a managed server terminates unexpectedly).

SNMP management tools

There are several available software packages that can use SNMP to interact with any agent for which a valid MIB is available. Examples of such general-purpose SNMP management tools are OpenView from Hewlett Packard, and UniCenter from Computer Associates. These SNMP management tools are widely used in large organizations. These tools provide a common management console to manage all resources on an organization's network.

To simplify discussions, all examples in this chapter use Hewlett Packard's OpenView as the administrator's SNMP management tool. However, you can use your preferred SNMP management tool with IONA Administrator.

IONA Administrator SNMP agent

The MIB and the SNMP agent supplied with IONA Administrator enable IONA application to be integrated into your SNMP management consoles. To an SNMP management tool, the IONA Administrator SNMP agent is no different from any other SNMP agent on the network. The Orbix MIB file specifies the data that can be obtained from the agent and the operations that are permitted on this data.

For more information about the MIB file supplied with Orbix, see [“Installing the Orbix MIB file” on page 118](#).

Orbix SNMP Integration

Overview

Orbix provides a gateway between the Orbix management service and SNMP management applications.

This support enables you to monitor instrumented IONA applications using third-party management consoles (for example, HP OpenView). This is especially useful if third-party management consoles are already used to monitor hardware or network configuration in your system.

From the point of view of an SNMP management application, the IONA Administrator SNMP gateway can simply be treated as an SNMP agent on the network. This section includes the following:

- [“IONA SNMP integration”](#).
- [“SNMP gateway”](#).
- [“SNMP gateway plugin”](#).

Note: IONA Administrator currently does not support get and set operations. IONA Administrator currently supports SNMP traps only.

IONA SNMP integration

Events received from the Orbix management service are converted into SNMP management information. This information is sent to designated hosts as SNMP traps, which can be received by any SNMP managers listening on the hosts. In this way, Orbix enables SNMP managers to monitor IONA-based systems.

IONA Administrator supports SNMP version 1 and 2 traps only.

SNMP gateway

IONA Administrator provides an SNMP gateway between the management service and SNMP managers. [Figure 44](#) shows the steps in this process:

1. IONA Administrator events received by the management service are passed on to the SNMP gateway.
2. The information extracted from the events are converted to SNMP traps using the MIB designed for IONA applications.
3. The SNMP traps are then sent to a list of hosts on which SNMP managers are running.

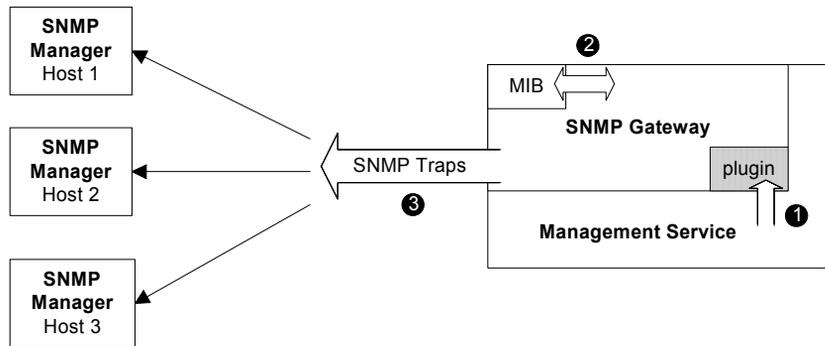


Figure 44: Overview of the SNMP Gateway

This diagram gives a simplified view of this process. The SNMP manager on each host also uses the IONA MIB file to translate the SNMP trap information into text names and values that it can understand.

SNMP gateway plugin

The SNMP gateway plugs into the management service, as shown in [Figure 44](#). The SNMP gateway enables IONA Administrator events to be sent from the management service to specific SNMP manager hosts as SNMP traps.

Configuring the SNMP Gateway

Overview

This section explains how to configure the IONA Administrator SNMP gateway. It explains the following:

- [“Configuring the SNMP gateway plugin”](#).
- [“Specifying SNMP managers”](#).
- [“SNMP manager list format”](#).
- [“Specifying SNMP management events”](#).
- [“Specifying SNMP event severities”](#).
- [“Specifying severities using the command line”](#).
- [“Installing the Orbix MIB file”](#).

Configuring the SNMP gateway plugin

To enable the management service to load the SNMP plugin, add the following variable in the `iona_services:management:server` scope:

```
ms_plugins = ["webconsole", "snmp"];
```

Specifying SNMP managers

You can specify the SNMP managers that you wish to access by setting the `snmp_trap:managers_list` configuration variable in the `iona_services:management` scope. This variable specifies the list of all SNMP managers to which you wish to send the SNMP trap.

For example:

```
snmp_trap:manager_list = ["boston:162:public:2",  
                          "dublin:9998:public:1"];
```

This example list contains two entries, and specifies sending traps to two SNMP managers.

SNMP manager list format

The format of each entry in the SNMP trap manager list is as follows:

```
"hostname:port_number:community_name:SNMP_protocol_version"
```

The *hostname* specifies the host on which the SNMP manager is running.

The *port_number* is the port that the SNMP manager uses to listen for traps.

The *community_name* specifies the SNMP community in which the SNMP managers are listening. An SNMP manager can listen in a number of *communities*, which are like user groups. In this example, the SNMP managers will only receive events if they are listening with the community name `public`. This is the default in most SNMP installations.

The *SNMP_protocol_version* can be set to either 1 or 2.

Specifying SNMP management events

You can specify which management events get sent to the SNMP gateway before they are sent on as traps to the SNMP managers.

The management service receives management events from various managed servers. It forwards these management events to a number of registered listeners using one or more event filters. These event filters assign a priority to each event and forward or discard the events based on priority.

Using this mechanism, you can control which events are sent to the subscribers registered with a particular filter. The SNMP Gateway uses an event filter called `SNMPFilter`.

Example SNMP management events

An example application permits management events all beginning with the following prefix:

```
com.bigcompany.myapp
```

An administrator wishes to pass all of these events to the SNMP gateway but block all others. This means changing the default severity assignments. Management events are assigned a severity in the range 0 (Critical) to 3 (Information).

For more information about management events and filters, see [Chapter 4](#).

Specifying SNMP event severities

To specify event severities for the example management event, you must to perform the following steps:

1. Ensure that the default severity for the filter is set to 3 (Information).
2. Set the severity of all events whose names begin with the string `com.bigcompany.myapp` to 2 (Warning).
3. Set the filter's threshold to 3. This means that the filter will permit all events with severity less than 3. In this case, only events whose names begin with the string `com.bigcompany.myapp` will be sent to the SNMP Gateway.

Specifying severities using the command line

You can specify event severities for the example management events using a simple command-line tool as follows:

Step	Action
1	Set the environment for your domain, and ensure that all services including the management service are running, and configured to load the SNMP gateway plugin.
2	Start the command line tool as follows: <pre>java itiadmin.command -ORBdomain_name mydomain</pre> You can get help at any time by typing <code>help</code> .
3	Check the default severity for the SNMP gateway filter. <pre>itiadmin.command> default_severity show -filter SNMPFilter 3</pre>
4	Assign a severity of 2 to any events whose names begin with <code>com.bigcompany.myapp</code> : <pre>itiadmin.command> severity modify com.bigcompany.myapp 2 -filter SNMPFilter</pre>
5	Modify the threshold to pass only events whose severities are less than 3 (that is, permit only <code>Warning=2</code> , <code>Error=1</code> or <code>Critical=0</code> events). <pre>itiadmin.command> threshold modify 3 -filter SNMPFilter</pre>

Installing the Orbix MIB file

A Management Information Base (MIB) file is a database of objects that can be managed using SNMP. Orbix provides the `iona_admin_mib.txt` file, which describes the MIB for Orbix. This file is available in the following directory:

install-dir/asp/version/doc/admin/

You should install the `iona_admin_mib.txt` file using your chosen third party management console (for example, HP OpenView). For information on how to import a MIB file, please consult the documentation for your chosen third-party console.

Note: SNMP is an unreliable protocol. If you are generating management events that are intended for an SNMP-based management console, you should continue to emit the event periodically until the cause of the error or event has been acknowledged or reset.

Glossary

Administration

All aspects of installing, configuring, deploying, monitoring, and managing a system.

Application Server

A software platform that provides the services and infrastructure required to develop and deploy middle-tier applications. Middle-tier applications perform the business logic necessary to provide web clients with access to enterprise information systems. In a multi-tier architecture, an application server sits beside a web server or between a web server and enterprise information systems. Application servers provide the middleware for enterprise systems.

CORBA

Common Object Request Broker Architecture. An open standard that enables objects to communicate with one another regardless of what programming language they are written in, or what operating system they run on.

Configuration

A specific arrangement of system elements and settings.

Controlling

The process of modifying the behavior of running software components, without stopping them.

Details Pane

The display pane on the right hand side of the IONA Administrator Web Console user interface.

Deployment

The process of distributing a configuration or system element into an environment.

Domain

An abstract grouping of managed server processes and hosts within a physical location. Processes within a domain share the same configuration and distributed application infrastructure. A domain is equivalent to an Orbix configuration domain.

EJB

Enterprise Java Beans. Sun Microsystems' architecture for the development and deployment of reusable, object-oriented, middle-tier components. EJBs can be either session beans or entity beans. EJB enables the implementation of a multi-tier, distributed object architecture. See

<http://java.sun.com/products/ejb/>

Event

An occurrence of interest, which is emitted from a managed entity.

Host

Generic term used to describe a computer, which runs parts of a distributed application.

Installation

The placement of software on a computer. Installation does not include Configuration unless a default configuration is supplied.

Instrumentation

Code instructions that monitor specific components in a system (for example, instructions that output logging information on screen.) When an application contains instrumentation code, it can be managed using a management tool such as IONA Administrator.

Invocation

A request issued on an already active software component.

J2EE

Java 2 Enterprise Edition. An environment for developing and deploying enterprise applications. The J2EE platform consists of services, application programming interfaces (APIs), and protocols that provide the functionality for developing multi-tiered, Web-based applications.

JRE

Java Runtime Environment. A subset of the Java Development Kit required to run Java programs. The JRE consists of the Java Virtual Machine, the Java platform core classes and supporting files. It does not include the compiler or debugger.

JMX

Java Management Extensions. Sun's standard for distributed management solutions. JMX provides tools for building distributed, Web-based solutions for managing devices, applications and service-driven networks.

Managed Application

An abstract description of a distributed application, which does not rely on the physical layout of its components.

Managed Entity

A generic manageable component. Managed entities include managed domains, servers, containers, modules, and beans.

Managed Server

A set of replicated managed processes. A managed process is a physical process which contains an ORB and which has loaded the management plugin. The managed server can be an EJB application server, CORBA server, or any other instrumented server that can be managed by IONA Administrator.

Managed Process.

A physical process which contains an ORB and which has loaded the management plugin.

Management

To direct or control the use of a system or component. Sometimes used in a more general way meaning the same as Administration.

MBean

A JMX term used to describe a generic manageable object.

Monitoring

Observing characteristics of running instances of software components. Monitoring does not change a system.

Navigation Trail

A linear list of managed servers on top of Details View, which shows the path taken to the currently displayed managed entity.

Navigation Tree

The tree on the left hand side of the IONA Administrator Web Console and IONA Configuration Explorer.

ORB

CORBA Object Request Broker. This is the key component in the CORBA architecture model. It acts as the middleware between clients and servers.

Process MBean

The is the first-level MBean that is exposed for management of an application. It is the starting point for navigation through an application in the IONA Administrator Web Console

Runtime Administration, Runtime Management

Encompasses the running, monitoring, controlling and stopping of software components.

SNMP

Simple Network Management Protocol. The Internet standard protocol developed to manage nodes on an IP network. It can be used to manage and monitor all sorts of devices (for example, computers, routers, and hubs)

Starting

The process of activating an instance of a deployed software component.

Stopping

The process of deactivating a running instance of a software component.

Web Services

Web services are XML-based information exchange systems that use the Internet for direct application-to-application interaction. These systems can include programs, objects, messages, or documents.

Web Services Container

A Web services container provides an environment for deploying and running Web services. A Web services container is typically deployed and runs in an application server.

XML

Extensible Markup Language. XML is a simpler but restricted form of Standard General Markup Language (SGML). The markup describes the meaning of the text. XML enables the separation of content from data. XML was created so that richly structured documents could be used over the web. See

<http://www.w3.org/XML/>

Index

A

- agent_container:persistent_filename 85
- agent_container:ping_interval 85
- Apache Log4J, configuration 101
- architecture
 - IONA Administrator Web Console 9
 - IONA Configuration Explorer 12
- attributes
 - setting server 36
 - viewing server 34
- automatic work queues 40

B

- browser
 - caching 27
 - recommended versions 20
- browsers, recommended versions 20

C

- C++ configuration 100
- client-id 102
- commands
 - domain-name_env 20
 - start_domain-name_services 20
- configuration namespaces
 - creating 75
 - deleting 79
 - modifying 72
 - viewing 68
- configuration repository domain 63
- configuration scope
 - iona_services:management 82
- configuration scopes
 - creating 75
 - deleting 79
 - modifying 72
 - viewing 68
- configuration variables
 - Apache Log4J, 101
 - creating 77
 - deleting 80
 - modifying 73

- viewing 69
- consolidated logging, configuring 90
- CORBA, definition 119

D

- default instrumentation, enabling 33
- details pane
 - IONA Administrator Web Console 25
 - refreshing 27
- domain-name_env command 20
- domains
 - definition 5, 32, 119

E

- EJB, definition 120
- EMS, definition 96
- Enterprise Management Systems 96
- event_barrier:persistent_filename 88
- event filters, for management service 88
- event history, for management service 87
- event_history:memory_capacity 87
- event_history:persistent_filename 87
- event_history:trim_interval 87
- event_listener:consolidated_logging 90
- event log
 - configuring 84
 - sending output to a file 84
- event_log 84
- event_log, filter 84
- event_log:filename_base 91
- EventLog MBean 42
- event_processor:idle_period 90
- event_processor:persistent_filename 86
- event_processor:ping_interval 86
- events
 - full details, viewing 48
 - monitoring 45
 - refreshing 47
 - setting threshold 48
 - severities, SNMP 117
 - viewing 47
- Events Console

starting 46

F

file-based domain 63
 filters, event
 management service log 84
 Filters attribute 43

H

https, starting the console 21

I

IBM Tivoli integration 96
 icons
 IONA Administrator server 29
 IONA Administrator toolbar 28
 IONA Configuration Explorer 64
 Orbx Configuration Authority 55
 IDL Type field 74
 IIOP 5
 IncomingRequests attribute 41
 IncomingRequestThroughput parameter 42
 instrumentation, definition 6, 120
 IONA Administrator Web Console
 components 5
 managed server icons 29
 managing servers 31
 navigating 25
 overview 8
 starting 20, 21
 toolbar icons 28
 iona_admin_mib.txt file 118
 IONA Configuration Explorer 61
 icons 64
 iona_services:management scope 82
 IONA Tivoli Provider 96
 itconfigure tool 98
 IT_MGMT_SVC subsystem 84
 it_reponse_time_logger 99, 100

J

J2EE, definition 120
 Java configuration 101
 JMX 33
 JMX, definition 121

L

life cycle message formats 107
 Log4J, configuration 101
 log file interpreter 96
 logging filters 42
 logging message formats 104
 log_properties 101

M

managed servers
 definition 32
 drilling into 35
 IONA Administrator icons 29
 monitoring 32
 viewing 33
 management scope 82
 management service
 configuring 81
 overview 5
 MBean monitor 99
 MBeans
 definition 33
 Process MBean 122
 MBeans, definition 121
 MIB
 definition 111

N

navigation tree
 IONA Administrator Web Console 25
 New Window button 27
 Number of threads 40

O

operations, invoking 37
 ORB, definition 122
 ORB core management 40
 Orbx Configuration Authority 53
 icons 55
 starting 56
 Orbx Configuration GUI 98
 orb_plugins 84, 100
 OutgoingRequests attribute 41

P

performance logging 96
 plugins:it_mgmt:managed_server_id:name 33

- plugins:it_response_time_collector:client-id 102
- plugins:it_response_time_collector:filename 100
- plugins:it_response_time_collector:log_properties 101
- plugins:it_response_time_collector:period 100
- plugins:it_response_time_collector:server-id 102
- plugins:it_response_time_collector:syslog_appID 101
- plugins:it_response_time_collector:system_logging_enabled 101
- plugins:local_log_stream:filename 84
- plugins:orb:is_managed 33
- policies:well_known_addressing_policy:http:addr_list 22, 57
- policies:well_known_addressing_policy:https:addr_list 22
- Process MBean 122
- Process MBean, definition 33

R

- Refresh button 27
- Request counter 98
- Request throughput 41
- resource agents
 - configuring for the management service 85
- Response time collector 98
- Response time logger 98
- running 107

S

- SamplePeriodBegan attribute 42
- SamplePeriodEnded attribute 42
- SamplingInterval attribute 42
- secure domain
 - logging into 23
 - starting the console 21
- server_binding_list 99
- server-id 102
- server ID, configuring 102
- server load 40
- server logging 42
- servers
 - monitoring 32
 - viewing 33
- servlet_binding_list 99
- severity_assigner:default_filename 89
- severity_assigner:persistent_filename 88
- shutting_down 107

SNMP

- agent 110
- configuration 115
- definition 110, 122
- event severities 117
- gateway 113
- Management Information Base 111
- manager 110
- plugin 114
- snmp_trap:manager_list 115
- start_domain-name_services command 20
- starting_up 107

T

- Thread pool size 40
- Thread queue length 40
- Tivoli integration 96
- Tivoli Task Library 96
- TotalNumberOfThreadsInUse attribute 40
- TotalQueueLength attribute 40
- TotalThreadPoolSize attribute 40

U

- Update operation 42

V

- viewLog operation 43

W

- web browser
 - caching 27
 - recommended versions 20
- web server
 - for management service 92
- web_server:port_number 92
- Web Services, definition 122

X

- XML, definition 123

