Oracle® Enterprise Manager

Installation and Configuration Guide for Microsoft Systems Center Operations Manager Connector

Release 12.1 (12.1.0.2.0)

E28572-04

September 2012

Oracle Enterprise Manager Installation and Configuration Guide for Microsoft Systems Center Operations Manager Connector, Release 12.1 (12.1.0.2.0)

E28572-04

Primary Author: Leo Cloutier

Contents

A Customizing Microsoft SCOM

B Enterprise Manager Event Samples

C Enterprise Manager Event Format

D SCOM Web Service WSDL Format

Index

Preface

This *Installation and Configuration Guide for Microsoft Systems Center Operations Manager (SCOM) Connector* provides the required information to install and configure the Microsoft System Center Operations Manager (SCOM) Connector that integrates Oracle Enterprise Manager with SCOM management tools and help desk systems.

Audience

This guide is written for Oracle Enterprise Manager system administrators who want to install and configure the Microsoft SCOM Connector to enable integration between Oracle Enterprise Manager and Microsoft SCOM.

You should already be familiar with Oracle Enterprise Manager.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Related Documents

For more information, see the following books in the Oracle Enterprise Manager documentation set:

- *Oracle Enterprise Manager Cloud Control Basic Installation Guide*
- *Oracle Enterprise Manager Cloud Control Advanced Installation and Configuration Guide*
- *Oracle Enterprise Manager Cloud Control Administrator's Guide*
- *Oracle Enterprise Manager Cloud Control Upgrade Guide*
- *Oracle Enterprise Manager Framework, Host, and Services Metric Reference Manual*
- *Oracle Enterprise Manager Command Line Interface*

- *Oracle Enterprise Manager Cloud Control Extensibility Programmer's Guide*
- *Oracle Database 2 Day DBA*

The latest versions of this and other Oracle Enterprise Manager documentation can be found at:

`http://www.oracle.com/technology/documentation/oem.html`

Oracle Enterprise Manager also provides extensive online help. Click **Help** on any Oracle Enterprise Manager page to display the online Help system.

Printed documentation is available for sale in the Oracle Store at

`http://oraclestore.oracle.com/`

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

`http://otn.oracle.com/membership/`

If you already have a user name and password for OTN, then you can go directly to the documentation section of the OTN Web site at

`http://otn.oracle.com/documentation/`

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1 Introduction to the Connector

The Microsoft Systems Center Operations Manager (SCOM) Connector (version 12.1.0.2.0) integrates Oracle Enterprise Manager with Microsoft SCOM 2007 through web services. Using this connector you can create, update or close alerts on SCOM system for any event generated in Enterprise manager, which ensures an active synchronization between SCOM and Enterprise Manager. Events in Enterprise Manager are equivalent to alerts in the SCOM system.

There are two types of SCOM connectors available:

- The SCOM pre R2 Connector supports the RTM and SP1 versions of SCOM
- The SCOM R2 Connector supports SCOM R2

You must determine which connector you want to use based on the version of SCOM being used in your environment.

1.1 Event Templates

Event templates are XML transformation style sheets that map Enterprise manager event attributes to SCOM alert attributes.

The SCOM event connector contains predefined event templates for create and update event operations on the SCOM system. These templates can be customized further to accommodate more requirements. Oracle recommends that you back up these Oracle-provided templates before customizing them.

Refer to Chapter 5, "Working With Event Templates" for more details

1.2 Retry

The Retry section, available on the connector configuration page, allows users to specify whether failed operations need to be retried within a configurable expiration time.

Enabling the Retry option gives users the option to specify whether they want to resend a create or update request if the request fails the first time and the time period after which they want to give up the retry. Enterprise Manager retries every five minutes until the request is successful or the retry interval expires.

1.3 Versions Supported

This connector supports the following versions of Oracle Enterprise Manager and Microsoft SCOM:

- Oracle Enterprise Manager Cloud Control 12*c* Release 1

- Microsoft Systems Center Operations Manager version 2007 (including R2)

You can install the SCOM Agent on the Microsoft Windows (2000, 2003, 2008, XP) platform.

The base Enterprise Manager version number for the Microsoft Systems Center Operations Manager Connector Release 12.1.0.2.0 is Enterprise Manager 12*c* Release 1 (12.1.0.1 with Bundle Patch 1) and higher versions.

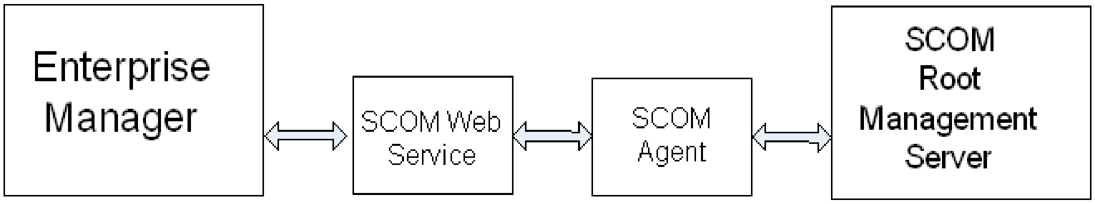
2 Installing the Connector

The Oracle Enterprise Manager Connector Framework requires a web service interface for forwarding event information to Microsoft SCOM. To integrate with Enterprise Manager, a third-party SCOM web service front-end must be installed. In addition to the web service front-end, an Oracle SCOM Agent must also be installed. Both of these components are included in the Oracle Enterprise Manager SCOM connector installation package.

You can install the web service on any Unix or Windows system that has connectivity with the SCOM server. In addition to the SCOM web service front-end, you must also install a back-end Oracle SCOM Agent. The Oracle SCOM Agent is preconfigured and is also included in the Oracle Enterprise Manager SCOM connector installation package.

Figure 2–1 shows the communication between the various components of the SCOM Connector.

Figure 2–1 Connector Communication Between Components



The following sections in this chapter discuss these topics:

- Installation Platforms
- Prerequisites
- Installing the Connector in Enterprise Manager
- Installing and Running the Oracle SCOM Agent
- Installing the Microsoft SCOM Web Service

2.1 Installation Platforms

You can install the SCOM web service on the following platforms that support Java JRE 1.6:

- Microsoft Windows

- Sun Solaris
- HP-UX
- Linux

You can install the Oracle SCOM Agent on the Microsoft Windows (2000, 2003, 2008, XP) platform.

2.2 Prerequisites

In order to install the connector, the My Oracle Support Preferred Credentials must be set to enable the download of the connector through self-update. To set the credentials, perform the following steps:

1. From the Enterprise Manager console, click **Setup**.
2. Select **My Oracle Support** in the pane.
3. Select **Set Credentials**, then enter the username and password information and click **Apply**.
4. From the Enterprise Manager console, click **Setup**.
5. Select **Extensibility** in the pane.
6. Select **Self Update** and click **Check Updates** to schedule a job to collect the metadata for the connectors.

Refer to the "Setting up Self Update" chapter in the *Oracle Enterprise Manager Cloud Control Administrator's Guide* for more details.

The following are prerequisites you should review before installing of the connector:

- A utility for unzipping .zip files is available where the SCOM Agent is to be installed
- Java JRE 6.0 Update 11 or higher is installed on the system where the SCOM Web Service will be installed.
- If you want Oracle Enterprise Manager to forward events to Microsoft SCOM, you must import the OracleEnterpriseManager.Alert.Creator Management Pack from the Microsoft SCOM server. The Management Pack file is provided with the SCOM Agent installation binaries.

2.3 Installing the Connector in Enterprise Manager

The SCOM connector is not included out-of-box so it must be downloaded from the Oracle Enterprise Manager store. All connectors in the Enterprise Manager store will be available on the Self Update page for downloading provided the My Oracle Support Preferred Credentials are configured. See Section 2.2, "Prerequisites" for instructions. The installation files for the Oracle SCOM agent and web service are packaged with the SCOM connector. The connector must be installed first in order to gain access to the installation files.

Perform the following steps to download and install (apply) the connector.

1. From the Enterprise Manager console, click **Setup**.
2. Select **Extensibility** in the pane.
3. Get a list of available connectors by selecting **Self Update** and then clicking **Management Connector**.

The SCOM R2 and pre R2 connectors should be listed here with a status of Available.

4. Click the appropriate connector to select it and then click **Download**.

   The Schedule Download window will be displayed to determine when the download should be performed.

5. Click **Select** to download immediately.

   > **Note:** If you want to schedule the download for a later time, specify the date/time when the download should take place and click **Select**. You must return to finish the remainder of this procedure after the scheduled download date/time.

6. If necessary, refresh the screen until the status of the connector changes to Downloaded.

7. Click the SCOM connector to select it and then click **Apply**.

   You will be prompted to confirm the operation and then a window appears that indicates the request has been submitted.

8. Refresh the screen until the status of the connector changes from Apply Scheduled to Applied.

2.4 Installing and Running the Oracle SCOM Agent

The following sections provide procedures for installing and running the Oracle SCOM Agent.

2.4.1 Preinstallation Requirements

The following requirements apply to the system where the Oracle SCOM Agent is installed.

- .NET framework 2.0 is installed
- .NET framework 3.0 is installed
- ASP.NET 2.0 is installed

  After .NET framework 2.0 is installed, enter the following command to install ASP.NET 2.0:

  ```
  %SystemRoot%\Microsoft.NET\Framework\v2.0.xxxxx\aspnet_regiis -i
  ```

  > **WARNING: This command upgrades other applications running under IIS to version 2.0.**

- Internet Information Services (IIS) 5.0 or higher is installed
- Operations Manager 2007 Console is installed
- ASP.NET 2.0 extensions are enabled in IIS

WARNING: Before proceeding, enabling ASP.NET 2.0 extensions may affect other applications that are running under IIS in an earlier version.

To enable ASP.NET extensions, perform the following steps:

1. Open the IIS Manager and expand the local computer in the left pane.
2. Right-click on Web Service Extensions and select **Allow all Web service extensions for a specific application ...**
3. Select **ASP.NET v2.0.0xxxxx** from the pull-down list, where xxxxx is the version of the .NET framework 2.0 that is installed.
4. Click **OK** to enable the ASP.NET extensions.

2.4.2 Setting Up the Agent Account

Before installing the Oracle SCOM Agent, you need to set up an account for the Oracle SCOM Agent to access the SCOM API. The account must satisfy the following requirements:

- Must be a domain account
- Must be used exclusively by the Oracle SCOM Agent
- Must have local administrator permissions
- Must be a member of a group designated as an Ops Mgr Administrator (See "Ops Mgr Administrator Groups" below)
- Must be a member of an Ops Mgr role that has an Author profile (See "Ops Mgr user role for Oracle SCOM Agent" below)

Ops Mgr Administrator Groups

To determine the groups that are designated as an Ops Mgr Administrator, perform the following steps:

1. In the Administration pane of the Ops Mgr 2007 console, select **Administration**, then **Security**, then **User Roles**. The User Roles should be displayed in the center pane.
2. Right-click on Operations Manager Administrators and select **Properties**. The General tab lists the groups that have administrative permissions in the User Role members window. The account the Agent uses must be a member of one of the groups listed here, or you need to add one of its groups to this list.

Ops Mgr user role for Oracle SCOM Agent

To add a new Ops Mgr 2007 user role for the Oracle SCOM Agent, perform the following steps:

1. In the Administration pane of the Ops Mgr 2007 console, select **Administration**, then **Security**, then **User Roles**. Right-click on **User Roles** and select **New User Role**, then **Author**. The Create User Role wizard window appears.
2. Enter the name of the Oracle SCOM Agent in the **User Role Name** field.
3. Click **Add** to add a user role member. The Select Users or Groups pop-up window appears.

4. Enter the domain user account information in the **Enter the object names to select** dialog box.

   Since this is a domain user account, you need to specify the account as <DOMAIN>\<username>, where <DOMAIN> is the Windows domain where the account was created, and <username> is the user name of the account set up for the Oracle SCOM Agent.

5. Click **OK** to add the Oracle SCOM Agent account as a user role member.
6. Click **Next** to go to the Approve targets page.
7. Select the targets that you want the Oracle SCOM Agent to access.

   These are the target types for the events that the Oracle SCOM Agent will handle. Typically, you would use the default "All targets are automatically approved, including targets in Management Packs imported in the future."

8. Click **Next** to go to the Approve groups page.
9. Select the groups that you want the Oracle SCOM Agent to access. You would generally want to use the default of all groups.
10. Click **Next** to go to the Approve tasks page.
11. Click **Next** to go to the Approve views page.
12. Click **Next** to go to the Summary page.
13. Click **Create** to create the Oracle SCOM Agent user role.

2.4.3 Installing the Agent

The SCOM Web Service uses the back-end Oracle SCOM Agent to access the SCOM API. The Oracle SCOM Agent must be installed on a Windows system that has connectivity to the SCOM Root Management Server (RMS). The Oracle SCOM Agent is preconfigured to interface with the SCOM Web Service and requires minimal configuration.

> **Note:** There should only be one installed instance of the Oracle SCOM Agent.

To install the Oracle SCOM Agent, perform the following steps:

1. Obtain a copy of the agent installation file from the Management Server host machine as specified in Section 2.4.6, "Exporting the Agent Installation File" and put the SCOMAgent.zip installation file into the directory where you want to install the Oracle SCOM Agent.
2. Unzip the contents of the SCOMAgent.zip file to any directory.

   This creates the SCOMAgentInstaller.msi and the OracleEnterpriseManager.Alert.Creator.xml files in the designated directory.

3. Navigate to the directory and run the Oracle SCOM Agent installer by double-clicking on the SCOMAgentInstaller.msi file.

   This starts the installer and displays the Welcome to the Oracle SCOM Agent Setup Wizard page.

4. Click **Next** to display the Service Type page.

5. Select the type of service to be installed under IIS. The default and recommended service type is Web Site.

   This installs the service as a stand-alone web site. Selecting Virtual Directory installs the service as a Virtual Directory under an existing web site.

6. Click **Next** to display the Select Installation Folder page.

7. Enter the location to install the Agent or accept the default location of C:\iWaveSoftware\SCOMConnectorAgent.

8. Click **Next** to display the Oracle SCOM Agent Options page.

9. Enter the name to use when registering the connector in SCOM, or accept the default value of Oracle SCOM Agent.

10. Click **Next** to display the Confirm Installation page.

11. Click **Next** to display the SCOM Management Group Configuration page, and enter the information for the following required fields on the form:

    - Host name or IP address of the RMS server
    - Domain of the account to use when connecting to the SCOM API
    - User name of the account to use when connecting to the SCOM API
    - Password of the account to use when connecting to the SCOM API

    Note: The information for the account from Section 2.4.2, "Setting Up the Agent Account" should be entered in the Domain, Username, and Password fields.

12. Click **Done** to proceed. The window displayed next depends on the Service Type you selected in step 5 above.

    - If you selected Web Site:

      The Web Site Configuration window appears. This window contains the Web Site Name and Port Number fields, and both have default values. Accept the default values or change them to the desired values.

    - If you selected Virtual Directory:

      The Virtual Directory Configuration window appears. This window defines the name of the virtual directory to create and the web site within which it will be installed. Accept the default values or change them to the desired values.

13. For either window, click **OK** to proceed and display the Web Service Credentials window. This window defines the credentials to specify when testing the newly installed Oracle SCOM Agent.

14. Specify valid Windows account credentials and click **OK**.

    The installer now invokes the web service to verify that it is operational. A window pops up with the results of the test.

15. Close the window. The Installation Successful window appears. This window lists the URL of the Oracle SCOM Agent.

    Note the URL. You will need this whenever you install the SCOM Web Service.

16. Click **OK** to continue. The Installation Complete window appears.

17. Click **Close** to complete the Agent installation process.

2.4.4 Installing the Alert Creator Management Pack

A management pack file named OracleEnterpriseManager.Alert.Creator.xml is also included in the Agent installation zip file. This management pack is required to create alerts in SCOM. Follow the steps below to import the management pack into SCOM.

1. In the Administration pane of the OpsMgr console, select **Administration**, then **Management Packs**.
2. Right-click on **Management Packs** and select **Import Management Pack…** The Select Management Packs to Import window appears.
3. Navigate to the directory where the OracleEnterpriseManager.Alert.Creator.xml file is located.
4. Select the **OracleEnterpriseManager.Alert.Creator.xml** file and click **Open**. The Import Management Packs window appears.

> **Note:** Depending on the version of SCOM that you are running, you might see the following error when you attempt to open the management pack:
>
> ```
> OracleEnterpriseManager Alert Creator
> The OracleEnterpriseManager Alert Creator Management Pack will fail
> to import because it depends on the following Management Packs:
> System.Library version 6.0.6278.0
> Microsoft.SystemCenter.Library version 6.0.6278.0
> System.Health.Library version 6.0.6278.0
> Microsoft.Windows.Library version 6.0.6278.0
> Please add them to the list by clicking on the Add... button and
> searching in your machine.
> ```
>
> If you see this error, you need to use a text editor to change four management packs referenced in the References section of the management pack file. The version number information must be updated to match the version numbers used in your system

5. Click **Import** to import the management pack.
6. Click **Close** after the management pack has been imported.

The Agent is now configured to insert alerts into SCOM.

2.4.5 Starting and Stopping the Agent

The installer automatically starts the Agent. To stop the agent, open the IIS manager, select the web site where it was installed, then click **Stop**. To start, click **Start**.

2.4.6 Exporting the Agent Installation File

The agent installation file is included in the Self Update archive that was installed in Section 2.3, "Installing the Connector in Enterprise Manager". To extract the installation file, perform the following steps.

1. Determine the command required to export the agent installation file. To do this, perform the following steps:
   - From the Enterprise Manager console, click **Setup**.
   - Select **Extensibility** in the pane.

- Select **Self Update** and then click **Management Connector**.
- Click the SCOM connector name, then select **Export** from the **Actions** list.

A popup window will appear with the command required to export the file.

2. Open a command window on the Operations Manager host machine and navigate to a temporary directory where the file can be copied.
3. Log into emcli using the following command. You will be asked to provide the password information for the sysman account.

```
emcli login -username=sysman
```

4. Run the emcli export_update command from your system, changing <*dirname*> to the full path of the temporary directory.

   A zip file will be created. The file name will be comprised of the id value specified in the export_update command with a .zip extension.

5. Extract the agent installation file from the zip file using the following command:

```
unzip *.zip archives/SCOMAgent.zip
```

2.5 Installing the Microsoft SCOM Web Service

The SCOM web service acts as a front-end for all data flowing into and out of SCOM. Oracle Enterprise Manager posts calls to the web service whenever it needs to create or update an event.

You can install the SCOM web service on any Unix or Windows system that has connectivity to the Oracle SCOM Agent and the Oracle Enterprise Manager server.

2.5.1 Installing and Running the Microsoft SCOM Web Service on Unix

The following sections explain how to install and then subsequently run the Web Service.

2.5.1.1 Prerequisites

The following prerequisites must be met before proceeding to the next section.

- Oracle Java Runtime Environment (JRE) version 6 Update 11 or higher is installed.
- JAVA_HOME environment variable is set to the JRE installation directory.

2.5.1.2 Installing the Web Service on Unix

To install the web service on a Unix platform, perform the following steps:

1. Create a directory where you want to install the web service.
2. Obtain a copy of the web service installation file from the Management Server host machine as specified in Section 2.5.3, "Exporting the Web Service Installation File".
3. Copy the SCOM_webservices_adapter.jar file to the web service installation directory.
4. Open a terminal and change the working directory to the installation directory.
5. Enter the following command to unzip and extract the .jar file:

```
$JAVA_HOME/bin/jar xvf SCOM_webservices_adapter.jar
```

   This creates the adapters directory that contains the installation files.

> **Note:** If the system where the SCOM web service is being installed does not have the JDK installed, you cannot extract the jar file contents. You need to copy the jar file to a system that has the JDK installed and transfer the files after they have been extracted.

6. Enter the following command to change the working directory:

   ```
   cd adapters/endpoints/SCOM2007
   ```

7. Enter the following command to run the installation script:

   ```
   sh ./install.sh
   ```

8. When the script prompts whether you want to use HTTPS:
   - If you specify Y, the web service is set up to use HTTPS port number 8443.
   - If you specify N, the web service is set up to use HTTP port number 8080.
9. When the script prompts for the user name of the web service, enter the user name that must be provided to access the SCOM Web Service.
10. When the script prompts for the password of the web service, enter the password that must be provided to access the SCOM Web Service.
11. After the script prompts for the URL of the Oracle SCOM Agent, enter the URL that was noted when you installed the Oracle SCOM Agent.
12. After the script prompts for the username and password to use when accessing the Agent, enter a valid windows username and password.
13. After the script displays the message "SCOM Web Service Complete," press Enter to complete the installation.
14. If the web service was configured to run using the HTTPS protocol, you must set up SSL as specified in Section 6.1, "Configuring Enterprise Manager to use SSL".

The web service framework is now installed and ready to start.

2.5.1.3 Running the Web Service on Unix

To run the Microsoft SCOM Web Service framework commands listed with the following tasks, first change the working directory to the following in the installation directory:

adapters/bin

- **Start** — ./service.sh start
- **Shut Down** — ./service.sh stop
- **Restart** — ./service.sh restart
- **Check Status** — ./service.sh status

2.5.1.4 Testing the Web Service on Unix

Perform the following steps to verify that the Microsoft SCOM Web Service is functional.

1. Open a terminal and change the working directory to the adapters/bin directory in the installation directory.
2. Enter the following command to run the test script:

```
./testAdapter.sh
```

3. When the utility prompts for the web service password, enter the password you specified for the SCOM web service in step 10 of section Section 2.5.1.2, "Installing the Web Service on Unix".

If the test completes successfully, the last line the utility displays is "Test completed successfully."

2.5.2 Installing and Running the Microsoft SCOM Web Service on Windows

The following sections explain how to install and then subsequently run the Web Service.

2.5.2.1 Prerequisites

The following prerequisites must be met before proceeding to the next section.

- Oracle Java Runtime Environment (JRE) version 6 Update 11 or higher is installed.
- JAVA_HOME environment variable is set to the JRE installation directory.
- Zip utility, such as WinZip, is installed for unzipping a zip file.

2.5.2.2 Installing the Web Service on Windows

To install the web service on a Windows platform, perform the following steps:

1. Create a directory where you want to install the web service.
2. Obtain a copy of the web service installation file from the Management Server host machine as specified in Section 2.5.3, "Exporting the Web Service Installation File".
3. Copy the SCOM_webservices_adapter.jar file to the web service installation directory.
4. Open a terminal and change the working directory to the installation directory.
5. Download the SCOM_webservices_adapter.jar file from the Oracle Technology Network to the installation directory, then extract the component .jar files.
6. Enter the following command to unzip and extract the .jar file:

   ```
   %JAVA_HOME%\bin\jar xvf SCOM_webservices_adapter.jar
   ```

   This creates the adapters directory that contains the installation files.

   > **Note:** If the system where the SCOM web service is being installed does not have the JDK installed, you cannot extract the jar file contents. You need to copy the jar file to a system that has the JDK installed and transfer the files after they have been extracted.

7. Enter the following command to change the working directory:

   ```
   cd adapters\endpoints\SCOM2007
   ```

8. Enter the following command to run the installation script:

   ```
   .\install.bat
   ```

9. When the script prompts whether you want to use HTTPS:

   - If you specify Y, the web service is set up to use HTTPS port number 8443.

- If you specify N, the web service is set up to use HTTP port number 8080.

10. When the script prompts for the user name of the web service, enter the user name that must be provided to access the SCOM Web Service.
11. When the script prompts for the password of the web service, enter the password that must be provided to access the SCOM Web Service.
12. After the script prompts for the URL of the Oracle SCOM Agent, enter the URL that was noted when you installed the Oracle SCOM Agent.
13. After the script prompts for the username and password to use when accessing the Agent, enter a valid windows username and password.
14. After the script displays the message "SCOM Web Service Complete," click **Enter** to complete the installation.
15. If the web service was configured to run using the HTTPS protocol, you must set up SSL as specified in Section 6.1, "Configuring Enterprise Manager to use SSL".

The following steps are optional. If you want the web service to run as a Windows service, perform the following steps.

1. Change the working directory to the adapters\bin directory in the installation directory.
2. Enter the following command to install the web service as a Windows service:

```
service.bat install
```

The web service framework is now installed and ready to start.

2.5.2.3 Running the Web Service on Windows

Standalone Service

To start the SCOM web service framework when set up as a standalone application (not set up to run as a Windows service):

1. Change the working directory to the adapters\bin directory in the installation directory.
2. Run the following command:

```
startAdapters.bat
```

To shut down the SCOM web service framework, close the window where you started the adapter.

Windows Service

To start the SCOM web service framework when set up to run as a Windows service, enter the following command:

```
net start iWaveAdapters
```

To shut down the SCOM web service framework, enter the following command:

```
net stop iWaveAdapters
```

2.5.2.4 Testing the Web Service on Windows

Perform the following steps to verify that the Microsoft SCOM Web Service is functional.

1. Open a terminal and change the working directory to the adapters\bin directory in the installation directory.
2. Enter the following command to run the test script:

   ```
   .\testAdapter.bat
   ```

3. When the utility prompts for the web service password, enter the password you specified for the SCOM web service in step 11 of section Section 2.5.2.2, "Installing the Web Service on Windows" (Windows).

If the test completes successfully, the last line the utility displays is "Test completed successfully."

2.5.3 Exporting the Web Service Installation File

The web service installation file is included in the zip file that was extracted in section 2.4.6 "Exporting the Agent Installation File". To extract the installation file, perform the following steps.

1. Open a command window on the Operations Manager host machine and navigate to the temporary directory where the zip file was created.
2. Extract the web service installation file using the following command:

   ```
   unzip *.zip archives/SCOM_webservices_adapter.jar
   ```

3

Configuring the Connector

This chapter provides procedures to configure the two sub-pages of the main Configure Management Connector page, then explains how to perform other tasks to complete the configuration process.

This chapter discusses the following topics:

3.1 Creating a Connector Instance

Once the SCOM Connector is installed in Enterprise Manager, you must create one or more instances of this connector type. A connector instance refers to the configured instance of the connector with specific details, such as connector name, web-service end-points, username/password, and so forth of the SCOM system. Multiple connector instances for the same connector type are supported, which provides the flexibility to configure multiple SCOM systems with Enterprise Manager.

For example, you could have a production database and test database that Enterprise Manager monitors, and you can create two different connector instances pointed to two different SCOM systems (one for production targets and one for test targets) with events from production targets sent to one SCOM instance and events from test targets sent to the another SCOM instance. You can associate each instance with separate event rules. Using this setup, you can forward Enterprise Manager events generated from these two databases to two different SCOM systems.

Follow the steps below to create a connector instance:

1. From the Enterprise Manager console, click **Setup**.
2. Select **Extensibility** in the pane.
3. Select **Management Connectors** in the left pane.
4. Select a connector type of SCOM preR2 or SCOM R2 from the Create Connector drop-down list then click **Go**.
5. Enter the connector name and description on the next screen then click **OK**.

6. The connector instance shows up as unconfigured in the Connector Setup page. Figure 3–1 shows an unconfigured SCOM R2 connector instance.

Figure 3–1 Unconfigured SCOM R2 Connector Instance

3.2 Configuring the Connector

To configure the connector, follow these steps:

1. As Super Administrator from the Enterprise Manager console, click **Setup**.
2. Click **Extensibility**, then select **Management Connectors**.The Management Connectors page appears and lists all created connector instances.

   > **Note:** The status column with a green tick mark indicates that the connector is already configured.

3. Click the name of the SCOM Connector.

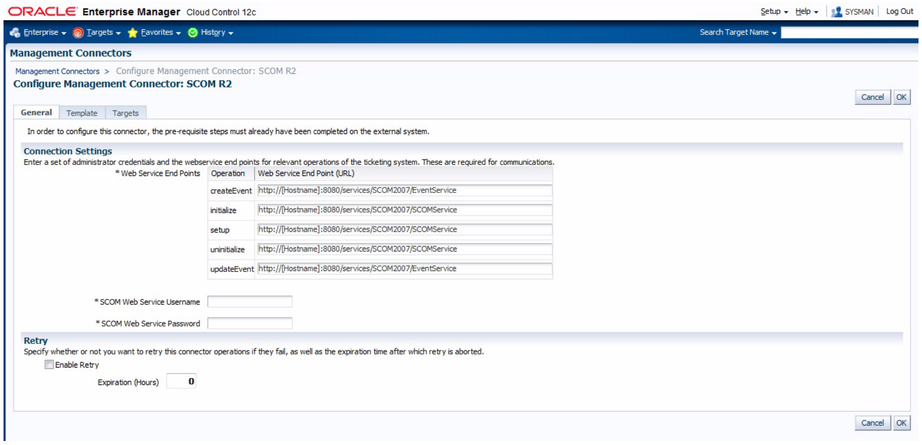
   The General tab of the Configure Management Connector page appears, as shown in Figure 3–2.

4. Provide the required settings. See Section 3.4, "Providing General Settings" for details.
5. Click **OK**.

   The Management Connectors page reappears. The row for the SCOM Connector should have a check mark in the Configured column.

   > **Note:** If you chose HTTPS as the protocol to establish a connection between SCOM and Enterprise Manager, make sure you have performed the steps specified in Section 6.1, "Configuring Enterprise Manager to use SSL".

Figure 3–2 General Tab of Configure Management Connector Page



Configure Management Connector: SCOM R2

In order to configure this connector, the pre-requisite steps must already have been completed on the external system.

Connection Settings

Enter a set of administrator credentials and the webservice end points for relevant operations of the ticketing system. These are required for communications.

* Web Service End Points

Operation	Web Service End Point (URL)
createEvent	http://[Hostname]:8080/services/SCOM2007/EventService
initialize	http://[Hostname]:8080/services/SCOM2007/SCOMService
setup	http://[Hostname]:8080/services/SCOM2007/SCOMService
uninitialize	http://[Hostname]:8080/services/SCOM2007/SCOMService
updateEvent	http://[Hostname]:8080/services/SCOM2007/EventService

* SCOM Web Service Username

* SCOM Web Service Password

Retry

Specify whether or not you want to retry this connector operations if they fail, as well as the expiration time after which retry is aborted.

☐ Enable Retry

Expiration (Hours) 0

3.3 Enabling And Disabling the Connector

You can enable or disable a connector instance without needing to reconfigure or remove the connector from the system.

You can enable or disable a connector from the Management Connector page by selecting the connector and clicking either Enable or Disable. After a connector is disabled, a red cross icon under the Status column appears. By default, the connector instance is enabled as soon as it is configured.

> **Note:** Only enabled connectors are available to forward events.

3.4 Providing General Settings

The following sections explain how to provide various configuration details.

3.4.1 Connection Settings

The SCOM Connector communicates with SCOM 2007 through the SCOM adapter installed in section 2.5. All fields are mandatory.

Web Service End Points – URL to use when performing event operations that require interaction with the SCOM adapter. The adapter uses two different URLs depending on the operation being performed. One URL is used for the registration of the SCOM Agent with Microsoft SCOM (setup, initialize, and uninitialize operations), and defaults to the following value:

http://[Hostname]:8080/services/SCOM2007/SCOMService

The other URL is used to create/update alert information in SCOM (createEvent and updateEvent operations), and defaults to the following value:

http://[Hostname]:8080/services/SCOM2007/EventService

Modify the URL for the operations based on whether the SCOM web service was set up using HTTP or HTTPS.

For HTTP:

- Replace [Hostname] in the URL with the hostname or IP address of the system where the SCOM adapter was installed.

For HTTPS:

- Change the protocol from http to https.
- Replace [Hostname] in the URL with the hostname or IP address of the system where the SCOM adapter was installed.
- Change the port number from 8080 to 8443.

Example

If the SCOM web service was installed on a system with a hostname of scomws, the following URLs would be configured for the createEvent and updateEvent operations.

For HTTP:

```
http://scomws:8080/services/SCOM2007/EventService
```

For HTTPS:

```
https://scomws:8443/services/SCOM2007/EventService
```

SCOM Web Service Username - User name that was configured for accessing the SCOM adapter in step 9 of the appropriate web service installation section (Section 2.5.1.2, "Installing the Web Service on Unix" and Section 2.5.2.2, "Installing the Web Service on Windows")

SCOM Web Service Password - Password that was configured for accessing the SCOM adapter in step 10 of the appropriate web service installation section (Section 2.5.1.2, "Installing the Web Service on Unix" and Section 2.5.2.2, "Installing the Web Service on Windows").

3.4.2 Retry

You can enable and disable the Retry option and configure its interval value. By default, the Retry is disabled. See Section 1.2, "Retry" for details. This setting applies to all events that the SCOM Connector processes. The retry period interval accepts the number of hours as any integer value. For example, 1 or 2.

3.5 Deleting the Connector

You can delete a connector from the Management Connector page by selecting the connector and clicking Delete. After a connector is deleted, all associated events rules are no longer configured to forward the event. Also, references to the event connector are removed from the event rules.

Additionally, if the connector deleted is the only connector instance of that connector type, the corresponding connector type is also deleted.

3.6 Testing the Microsoft SCOM Connector

Perform the following steps to verify that the connector is configured correctly:

1. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator permissions.

2. From the Enterprise Manager console, click **Setup**, then **Extensibility**, then **Management Connectors** from the drop-down menus. The Management Connectors page is displayed.
3. Click the name of the SCOM Connector.

   The General tab of the Configure Management Connector page appears.
4. Select and copy the URL specified for the createEvent or updateEvent operation.
5. Open an internet browser on the system where the Oracle Enterprise Manager server is installed.
6. In the address window, enter the URL that was copied in step 4 above. Add *?wsdl* to the end of the URL. The URL should appear similar to the following example:

   ```
   http://[Hostname]:8080/services/SCOM2007/EventService?wsdl
   ```

   [Hostname] is the actual host name or IP address where the SCOM adapter is installed.

   If the WSDL is loaded, this confirms that the connector is configured correctly for sending event information to SCOM.

4 Forwarding Enterprise Manager Events to SCOM

Events generated or updated in Oracle Enterprise Manager are not forwarded to SCOM unless you create rules to invoke the SCOM connector. A rule identifies the conditions that must be met before the connector is invoked.

The following section describes procedures that explain how to set up rules to create events and update events in SCOM-based events in Enterprise Manager:

- Setting Event Rules
- Viewing Events Via SCOM Console

4.1 Setting Event Rules

This section explains how to create a new rule in Enterprise Manager that invokes the SCOM connector. Log into the Enterprise Manager console with an account that has Super Administrator permissions and then perform the following steps:

1. From the Enterprise Manager console, click **Setup**, then **Incidents**, then **Incident Rules**.
2. Identify an existing rule set where the new rule can be added or create a new rule set as specified in the following sub steps:
   a. Click **Create Rule Set**. The Create Rule Set page is displayed.
   b. Enter a name and description, then click **Save**. The Incident Rules page is displayed again with the new rule set listed.
3. Click the rule set that you want to add the rule, then click **Edit**.
4. Click the Rules tab then click **Create**. The Select Type of Rule to Create popup is displayed.

   As you proceed below, the wizard interface explains various rules and their usage.

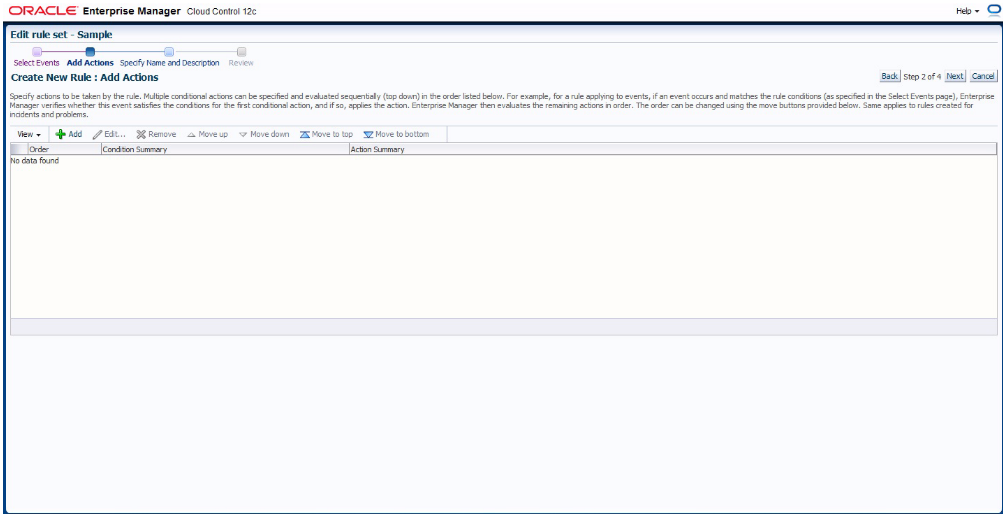
   For detailed information, refer to the "Incident Management" chapter in the *Oracle Enterprise Manager Cloud Control Administrator's Guide*.
5. Select Incoming events and updates to events then click **Continue**. The Select Events page is displayed.
6. Specifying the set of events that you want to forward, then click **Next**. The Add Actions page is displayed as shown in Figure 4–1.
7. Click **Add**. The Add Conditional Actions page is displayed as shown in Figure 4–2.

8. Select the SCOM Connector instance in the Forward to Event Connectors section then click **>**, then click **Continue**. The Add Actions page is displayed again with the new action listed.
9. Click **Next**. The Specify Name and Description page will be displayed.
10. Enter a name and description for the rule then click **Next**. The Review page is displayed.
11. Click **Continue** if everything looks correct. An information popup will be displayed that says, *Rule has been successfully added to the current rule set. Newly added rules are not saved until the Save button is clicked.*

> **Note:** You can click Back and make corrections to the rule if necessary.

12. Click **Save** to save the changes to the rule set.

Figure 4–1 Add Actions Page



ORACLE Enterprise Manager Cloud Control 12c
Help ▾

Edit rule set - Sample

Select Events
Add Actions
Specify Name and Description
Review

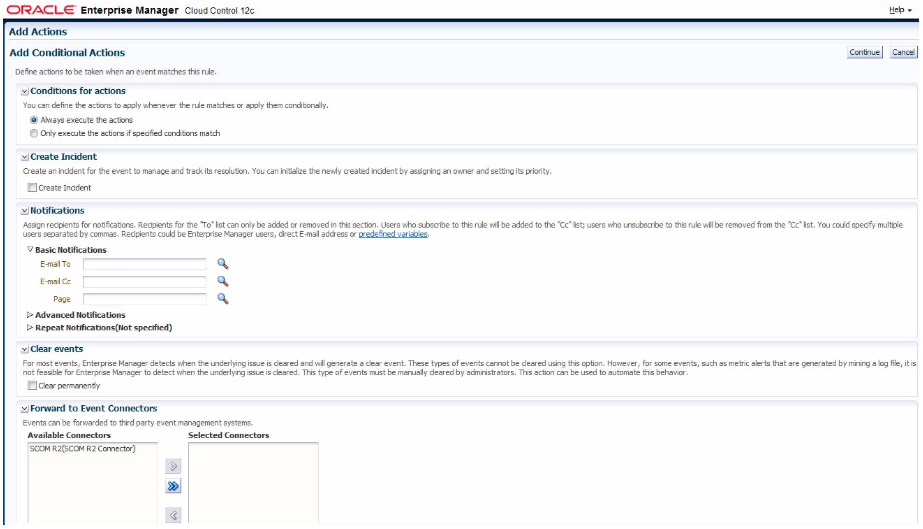
Create New Rule : Add Actions

Back
Step 2 of 4
Next
Cancel

Specify actions to be taken by the rule. Multiple conditional actions can be specified and evaluated sequentially (top down) in the order listed below. For example, for a rule applying to events, if an event occurs and matches the rule conditions (as specified in the Select Events page), Enterprise Manager verifies whether this event satisfies the conditions for the first conditional action, and if so, applies the action. Enterprise Manager then evaluates the remaining actions in order. The order can be changed using the move buttons provided below. Same applies to rules created for incidents and problems.

Order	Condition Summary	Action Summary
No data found		

Figure 4–2 Add Conditional Actions Page



Add Actions

Add Conditional Actions

Define actions to be taken when an event matches this rule.

Conditions for actions

You can define the actions to apply whenever the rule matches or apply them conditionally.

● Always execute the actions
○ Only execute the actions if specified conditions match

Create Incident

Create an incident for the event to manage and track its resolution. You can initialize the newly created incident by assigning an owner and setting its priority.

□ Create Incident

Notifications

Assign recipients for notifications. Recipients for the "To" list can only be added or removed in this section. Users who subscribe to this rule will be added to the "Cc" list; users who unsubscribe to this rule will be removed from the "Cc" list. You could specify multiple users separated by commas. Recipients could be Enterprise Manager users, direct E-mail address or predefined variables.

▽ Basic Notifications

E-mail To
E-mail Cc
Page

▷ Advanced Notifications
▷ Repeat Notifications(Not specified)

Clear events

For most events, Enterprise Manager detects when the underlying issue is cleared and will generate a clear event. These types of events cannot be cleared using this option. However, for some events, such as metric alerts that are generated by mining a log file, it is not feasible for Enterprise Manager to detect when the underlying issue is cleared. This type of events must be manually cleared by administrators. This action can be used to automate this behavior.

□ Clear permanently

Forward to Event Connectors

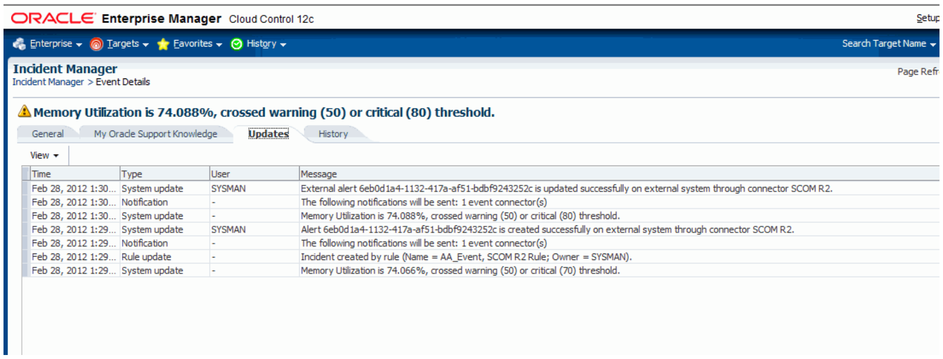
Events can be forwarded to third party event management systems.

Available Connectors	Selected Connectors
SCOM R2(SCOM R2 Connector)	

The following process occurs after the above created event rule is satisfied:

- A request is sent to the SCOM Connector when a event is created per the criteria mentioned in the rule.
- The alert is created or updated on the SCOM console.
- In Enterprise Manager, the event is annotated with a comment on the Updates tab to indicate that an event in SCOM has been created or updated as seen in Figure 4–3.

Figure 4–3 Incident Manager Event Update



Incident Manager

Incident Manager > Event Details

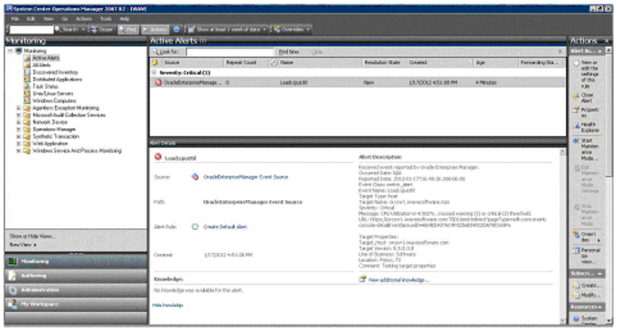
Memory Utilization is 74.088%, crossed warning (50) or critical (80) threshold.

Time	Type	User	Message
Feb 28, 2012 1:30...	System update	SYSMAN	External alert 6eb0d1a4-1132-417a-af51-bdbf9243252c is updated successfully on external system through connector SCOM R2.
Feb 28, 2012 1:30...	Notification	-	The following notifications will be sent: 1 event connector(s)
Feb 28, 2012 1:30...	System update	-	Memory Utilization is 74.088%, crossed warning (50) or critical (80) threshold.
Feb 28, 2012 1:29...	System update	SYSMAN	Alert 6eb0d1a4-1132-417a-af51-bdbf9243252c is created successfully on external system through connector SCOM R2.
Feb 28, 2012 1:29...	Notification	-	The following notifications will be sent: 1 event connector(s)
Feb 28, 2012 1:29...	Rule update	-	Incident created by rule (Name = AA_Event, SCOM R2 Rule; Owner = SYSMAN).
Feb 28, 2012 1:29...	System update	-	Memory Utilization is 74.066%, crossed warning (50) or critical (70) threshold.

4.2 Viewing Events Via SCOM Console

Whenever an alert is created in SCOM from an event that originates in Oracle Enterprise Manager, a link is provided in the alert text. To view the Oracle Enterprise Manager event that triggered the message, copy the URL and paste in a browser. You will be asked to log into Oracle Enterprise Manager. After logging in, the Oracle Enterprise Manager event information is displayed as seen in Figure 4–4.

Figure 4–4 Oracle Enterprise Manager Event Information



Look for: Find Now Clear

Source	Repeat Count	Name	Resolution State	Created	Age	Forwarding Sta...
Severity: Critical (1)						
OracleEnterpriseManage...	0	Load:cpuUtil	New	1/17/2012 4:51:08 PM	4 Minutes	

Alert Details

Load:cpuUtil

Source: OracleEnterpriseManager Event Source

Path: OracleEnterpriseManager Event Source

Alert Rule: Create Default Alert

Created: 1/17/2012 4:51:08 PM

Alert Description

Received event reported by Oracle Enterprise Manager:
Occurred Date: N/A
Reported Date: 2012-01-17 16:48:36 [illegible]
Event Class: metric_alert
Event Name: Load:cpuUtil
Target Type: host
Target Name: [illegible]
Severity: Critical
Message: CPU Utilization is 4.902%, crossed warning (1) or critical (3) threshold.
URL: [illegible]

Target Properties:
Target_Host: [illegible]
Target Version: [illegible]
Line of Business: Software
Location: Frisco, TX
Comment: Testing target properties

Knowledge:

No knowledge was available for this alert.

View additional knowledge...

Hide knowledge

5

Working With Event Templates

This chapter provides details on the default event templates shipped along with the SCOM Connector. The event templates specify the mappings between an Enterprise Manager event attributes and SCOM web service attributes:

- Familiarization with Event Templates
- Customizing Event Templates

5.1 Familiarization with Event Templates

This section describes the default mappings between the Enterprise Manager event data fields and the Microsoft SCOM alert data fields.

This section discusses the following topics:

- Default Templates
- Severity Mapping
- Priority Mapping

5.1.1 Default Templates

The event templates XML Style Sheet (XSL) files contain the mappings between the two systems. These files are located in the Self Update archive that was installed in Section 2.3, "Installing the Connector in Enterprise Manager". To extract the installation file, perform the steps in Section 5.2.3.1, "Exporting the Template Files".

Table 5–1 lists the XSL files that perform the mappings and provides a summary of each.

Table 5–1 XSL Files that Perform Mappings

File	Description
createEvent_request.xsl	Transforms the Oracle Enterprise Manager event data to the Microsoft SCOM alert format for the createEvent operation.
updateEvent_request.xsl	Transforms the Oracle Enterprise Manager event data to the Microsoft SCOM alert format for the updateEvent operation.

The following sections provide details about the default templates:

- Create Event Template
- Update Event Template
- Formatted Enterprise Manager Event Contents

5.1.1.1 Create Event Template

The Oracle Enterprise Manager Connector Framework invokes the createEvent operation whenever an event is generated in Oracle Enterprise Manager and a rule is configured to invoke the SCOM connector. createEvent_request.xsl is invoked during the process to transform the data from Oracle Enterprise Manager format to SCOM alert format. Table 5–2 lists the default field mappings between the Microsoft SCOM alert and the Oracle Enterprise Manager event.

Table 5–2 Create Event Template Mappings

SCOM Field Name	Required	Value
Name	Yes	Set to the event EventName field.
Description	Yes	Set to the formatted contents of the event as specified in the section below entitled “Formatted Enterprise Manager Event Contents”.
Priority	Yes	Set the priority based on the event priority. See the section entitled "Priority Mapping" below for details.
Severity	Yes	Set the severity based on the event severity. See the section entitled "Severity Mapping" below for details.

5.1.1.2 Update Event Template

The Oracle Enterprise Manager Connector Framework invokes the updateEvent operation whenever an event is generated in Oracle Enterprise Manager and a rule is configured to invoke the SCOM connector.

updateEvent_request.xsl is invoked during the process to transform the data from Oracle Enterprise Manager format to SCOM alert format. Table 5–3 lists the default field mappings between the Microsoft SCOM alert and the Oracle Enterprise Manager event.

Table 5–3 Update Event Template Mappings

SCOM Field Name	Required	Value
ID	Yes	Set to the event ExternalEventId field.
Resolution State	Yes	Set to 255 (Closed) if the event severity is Clear. Otherwise, set to 0 (New).
Alert History	No	Set to "Oracle Enterprise Manager cleared alert" if the event severity is Clear. Set to "Oracle Enterprise Manager changed alert severity to warning" if the event severity is Warning. Set to "Oracle Enterprise Manager changed alert severity to critical" if the event severity is Critical. Not set for other severity values.

5.1.1.3 Formatted Enterprise Manager Event Contents

Listed below is the format used by the templates for the Description fields. Text delimited by angle brackets (< and >) designates an Enterprise Manager Event field name that will be replaced by the actual field contents. Lines marked with double asterisks indicate they will only be present if the designated event field contains data. The Notes section below contains information about some of the fields.

```
Received event reported by Oracle Enterprise Manager:
Occurred Date: <OccurredDate>
ReportedDate: <ReportedDate>
```

```
Event Class: <EventClass>
Event Name: <EventName>
Target Type: <SourceInfo/TargetInfo/TargetType>
Target Name: <SourceInfo/TargetInfo/TargetName>
Severity: <Severity>
Message: <Message>
URL: <EventURL>

Target Properties:**
    <TargetProperty/Name>: <TargetProperty/Value>**

Event Context:**
    <EventContextAttributes/StringAttribute/Name>: <
EventContextAttributes/StringAttribute /Value>**
```

Notes

- The Target Properties section will only be present if the event contains target property values. Listed below are some sample target property name/value pairs:

  Line of business: Finance

  Owner: Finance-DB Team

- The Event Context section will only be present if the event contains event context attribute values. There can be StringAttribute or NumberAttribute values that contain a name/value pair.
- URL is set to the URL of the event details page in Enterprise Manager. Copy the URL to a browser to directly view the event in Enterprise Manager.

5.1.2 Severity Mapping

The SCOM alert severity is set based on the Enterprise Manager event severity value. The following table shows the values used when creating the alert in SCOM.

When the event severity is ...	Set the alert severity to ...
Fatal	Error
Critical	Error
Warning	Warning
Minor Warning	Warning
Informational	Information
Clear	Information

5.1.3 Priority Mapping

The SCOM alert priority is set based on the Enterprise Manager event priority value. The following table shows the values used when creating the alert in SCOM.

When the event priority is ...	Set the alert priority to ...
Fatal	High
Critical	High
Warning	Normal
Minor Warning	Normal

When the event priority is ...	Set the alert priority to ...
Informational	Low
Clear	Low

5.2 Customizing Event Templates

Although the default mappings are sufficient for most implementations, you can change them as needed. The following sections discuss:

- XML Format of Oracle Enterprise Manager Events
- Managing Templates in Oracle Enterprise Manager 12c
- Changing a Mapping

It is assumed that you already have a good understanding of XSL.

For reference information on the SCOM mappings, see Appendix A, "Customizing Microsoft SCOM".

5.2.1 XML Format of Oracle Enterprise Manager Events

Example 5–1 shows the format that the Oracle Enterprise Manager Connector Framework provides when an event is created or updated in Oracle Enterprise Manager.

Example 5–1 XML Format of Events

```
<EMEvent xmlns="http://xmlns.oracle.com/sysman/connector">
  <ExternalEventID/>
  <SystemAttributes>
    <EventClass/>
    <OccurredDate/>
    <ReportedDate/>
    <EventName/>
    <Severity/>
    <SeverityCode/>
    <SourceInfo>
      <TargetInfo>
        <TargetName/>
        <TargetTypeLabel/>
        <TargetProperty>
          <Name/>
          <Value/>
        </TargetProperty>
      </TargetInfo>
    </SourceInfo>
    <Message/>
    <EventURL/>
  </SystemAttributes>
  <EventContextAttributes>
    <NumberAttribute>
      <Name/>
      <Value/>
    </NumberAttribute>
    <StringAttribute>
      <Name/>
      <Value/>
    </StringAttribute>
  </EventContextAttributes>
</EMEvent>
```

Table 5–4 provides a description of the fields shown in Example 5–1.

Table 5–4 Field Descriptions for XML Format

Field	Description
ExternalEventId	Unique identifier of the message in HP Operations Manager. This will only be set for updates.
EventClass	Identifies the type of event
OccurredDate	Date/time the event occurred
ReportedDate	Date/time the event was reported by OMS
EventName	Descriptive name of the event
Severity	Severity of the event: Fatal, Critical, Warning, Minor_ Warning, Informational, or Clear
SeverityCode	Severity code for the event: FATAL, CRITICAL, WARNING, MINOR_WARNING, INFORMATIONAL, or CLEAR
TargetName	Target name that is a unique instance of the target type
TargetTypeLabel	Display name of the target type for which the event was generated
TargetProperty	Additional target properties that do not have a specific field in the event model (name:value pair)
Message	Description of the event
EventURL	Link to the web page for the event
EventContextAttributes	Additional event properties that do not have a specific field in the event model (name:value pair)

5.2.2 Managing Templates in Oracle Enterprise Manager 12*c*

Oracle Enterprise Manager 12*c* (Release 12.1.0.2 and later) includes the functionality to allow you to manage templates within Cloud Control. Templates, which are included as part of the connector installation, are automatically registered and are shown within Enterprise Manager.

If you need to use new a template that is not included as part of the connector installation, then you will need to register it using the `emctl` command. See Changing a Mapping for more information. Once you have registered the template, you will be able to manage it through Enterprise Manager.

Using Enterprise Manager, you can edit a template, view the XML code, delete a template, and restore a deleted template.

To begin:

1. Log in to Enterprise Manager Cloud Control.
2. From the Setup menu, select **Extensibility**, then **Management Connectors**.
3. On the Management Connectors page, select the **Microsoft SCOM Connector**. Click **Configure**.
4. On the Configure Management Connector page, click the **Template** tab.

Note: This feature is available on Oracle Enterprise Manager Release 12.1.0.2.0 or later only. To verify your installed release version:

1. Log in to Enterprise Manager.
2. From the Help menu, select **About Cloud Control**.
3. The pop-up window displays the installed version of Oracle Enterprise Manager.

5.2.2.1 Edit a Template with Enterprise Manager

To edit an existing template:

1. Log in to Enterprise Manager Cloud Control.
2. From the Setup menu, select **Extensibility**, then **Management Connectors**.
3. On the Management Connectors page, select the **Microsoft SCOM Connector**. Click **Configure**.
4. On the Configure Management Connector page, click the **Template** tab.
5. Select a template and click **Edit**.
6. After you edit the Description or Template Contents fields, click **Save**. On the confirmation page, click **Yes** to complete the template save.

5.2.2.2 View Template Code in Enterprise Manager

To view template code:

1. Log in to Enterprise Manager Cloud Control.
2. From the Setup menu, select **Extensibility**, then **Management Connectors**.
3. On the Management Connectors page, select the **Microsoft SCOM Connector**. Click **Configure**.
4. On the Configure Management Connector page, click the **Template** tab.
5. Select a template and click **View**.

The event templates are in XSLT format. A basic knowledge of XSLT is required to understand the code.

5.2.2.3 Delete a Template with Enterprise Manager

To delete a template:

1. Log in to Enterprise Manager Cloud Control.
2. From the Setup menu, select **Extensibility**, then **Management Connectors**.
3. On the Management Connectors page, select the **Microsoft SCOM Connector**. Click **Configure**.
4. On the Configure Management Connector page, click the **Template** tab.
5. Select a template and click **Delete**.
6. Click **Yes** to confirm the delete operation.

5.2.2.4 Restore a Deleted Template with Enterprise Manager

To restore a deleted template:

1. Log in to Enterprise Manager Cloud Control.
2. From the Setup menu, select **Extensibility**, then **Management Connectors**.
3. On the Management Connectors page, select the **Microsoft SCOM Connector**. Click **Configure**.
4. On the Configure Management Connector page, click the **Template** tab.
5. Select a template and click **Restore**.
6. Click **Yes** in the confirmation page that appears in order to restore.

> **Note:** You can restore out-of-box templates that you have may edited or customized to their default values. You cannot restore new templates that you have created (using the `emctl register_template` command) described in Registering Templates.

5.2.3 Changing a Mapping

This section explains how to customize the mapping between Enterprise Manager and the Microsoft SCOM web service. The procedure provides the steps required for changing a mapping.

1. Export the default template XSLT files from Enterprise Manager to use as a baseline. Follow the instructions in Section 5.2.3.1, "Exporting the Template Files" to obtain copies of the default template files.
2. Copy the XSL file you want to change to a new file to hold the custom template.
3. Make the desired mapping changes to the template file. See Section 5.2.3.2, "Modifying Templates" for guidance on mapping changes.
4. Register the new template with Enterprise Manager as specified in Section 5.2.3.3, "Registering Templates".

The files are now ready for integration. You do not need to stop and start OMS. The changes will automatically be picked up.

See Section 5.2.3.4, "Template Customization Example" for an example of how to customize templates.

For information on customizing the SCOM, see Section A, "Customizing Microsoft SCOM".

5.2.3.1 Exporting the Template Files

The connector installation jar file is included in the Self Update archive that was installed in Section 2.3, "Installing the Connector in Enterprise Manager". To extract the installation file, perform the following steps.

1. Determine the command required to export the adapter installation file. To do this, perform the following steps:

   - From the Enterprise Manager console, click **Setup**.
   - Select **Extensibility** in the pane.
   - Select **Self Update**, then click **Management Connector.**
   - Click the SCOM connector, then select **Export** from the **Actions** list.

   A popup window appears with the command required to export the file. See Figure 5–1 below for an example of the popup window.

2. Open a command window on the Management Server host machine and navigate to a temporary directory where the file can be copied.

3. Log into emcli using the following command. You will be asked to provide the password information for the sysman account.

   ```
   emcli login -username=sysman
   ```

4. Run the emcli export_update command from your system, changing <*dirname*> to the full path of the temporary directory.

   A zip file will be created. The file name will be comprised of the id value specified in the export_update command with a .zip extension.

5. Extract the connector installation jar file from the zip file using the following command.

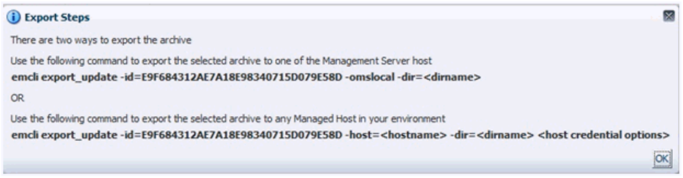
   ```
   unzip *.zip archives/*connector.jar
   ```

6. Extract the templates from the connector jar file using the following command:

   ```
   $JAVA_HOME/bin/jar xf archives/*.jar createEvent_request.xsl
   updateEvent_request.xsl
   ```

> **Note:** If the system where the connector file is located does not have the JDK installed, you cannot extract the jar file contents. You must copy the jar file to a system that has the JDK installed and transfer the template files after they have been extracted.

Figure 5–1 Export Steps



There are two ways to export the archive

Use the following command to export the selected archive to one of the Management Server host

```
emcli export_update -id=E9F684312AE7A18E98340715D079E58D -omslocal -dir=<dirname>
```

OR

Use the following command to export the selected archive to any Managed Host in your environment

```
emcli export_update -id=E9F684312AE7A18E98340715D079E58D -host=<hostname> -dir=<dirname> <host credential options>
```

5.2.3.2 Modifying Templates

A template is an XSLT file that transforms an input XML format to an output XML format. To make the mapping changes, you need to understand the input XML format being transformed and the output XML being generated by the template. The input document is the XML document that contains the Enterprise Manager event information. The output document is the request being sent to the SCOM adapter. The output XML document must conform to the data format specified in the SCOM EventService WSDL.

Appendix C, "Enterprise Manager Event Format" contains details on the format of the event data generated by Enterprise Manager. Appendix D, "SCOM Web Service WSDL Format" contains details on the XML format required by the SCOM EventService web service. Section A.1.1, "Mappings Between XML Format and Alert Field Names" provides the mapping between the SCOM EventService web service fields and the field names in SCOM. Listed below are the steps you should follow to modify the template file.

1. Review Section A.1.1, "Mappings Between XML Format and Alert Field Names" and Appendix D, "SCOM Web Service WSDL Format" to identify the names of the fields in the SCOM adapter that you are planning to modify. For example if you

wanted to populate the Custom1 field in SCOM, look the field up in Section A.1.1, "Mappings Between XML Format and Alert Field Names" and you would see that the SCOM adapter expects the data in the /create/event/extended-fields/string-field XML path

2. If you are planning on using data from the event, review Appendix C, "Enterprise Manager Event Format" to identify the field(s) you want to use in the mapping. Appendix B, "Enterprise Manager Event Samples" has sample transactions that could be very helpful in identifying what data is present in the XML document from Enterprise Manager.
3. Open the custom template file in a text editor or a utility that can be used to build XSLT files.
4. Make the necessary changes to the custom template file and save.
5. Unless the change is very simple, you should test the changes that you have made using the sample data from Appendix B. This will require a utility that performs XSLT translations.

5.2.3.3 Registering Templates

You need to reregister your customized templates before they are recognized in Enterprise Manager.

For each customized template, run the following emctl register_template connector command as a user with execute privilege on emctl and the ability to read the template:

```
$ORACLE_HOME/bin/emctl register_template connector
-t <template.xsl> -repos_pwd <reposPassword>
-cname <connectorName> -iname <internalName>
-tname <templateName> -ttype 2 [-d <description>]
```

Where:

```
<template.xsl> is the full pathname of the template file
<reposPassword> is the Enterprise Manager root (SYSMAN) password
<connectorName> is the name of the connector instance installed in step 5 of
section 3.1 “Creating a Connector Instance”.
<internalName> is the internal name to use for the template
<templateName> is the name to display for the template
<description> is the description of the template
For example: This command updates the createEvent template for the connector
instance with the name of “SCOM R2”:
emctl register_template connector
-t /home/oracle/customized_templates/createEvent_request.xml
-repos_pwd sysmanpass
-cname "SCOM R2"
-iname "createEvent"
-tname "Create Event Request"
-ttype 2
-d "Demo template"
```

5.2.3.4 Template Customization Example

By default, the alert name in the Microsoft SCOM alert is set to the Oracle Enterprise Manager metric column, and no custom fields are mapped. The following example shows how to change the alert name and add mappings for custom fields. The changes made to the default mapping are:

The alert name is modified to use a hard-coded value of Event From Oracle Enterprise Manager.
Custom field 1 is set to the Target Type.
Custom field 2 is set to the Target Name.
Custom field 3 is set to the Metric Name.
Custom field 4 is set to the Metric Column.

1. Navigate to the Management Connector Updates page and export the SCOM Connector. Copy the first emcli command listed in the Export Steps window for later use.

   ```
   emcli export_update -id=DD4E9161C5E7129F9641447FB4F0497B
   -omslocal -dir=<dirname>
   ```

2. Open a command window on the OMS server machine and change the working directory to the temporary directory.

   ```
   cd /home/oracle/customized_templates
   ```

3. Log into emcli.

   ```
   emcli login -username=sysman
   ```

4. Run the emcli command from step 1 to export the zip file to the temporary directory.

   ```
   emcli export_update -id=DD4E9161C5E7129F9641447FB4F0497B
   -omslocal -dir=/home/oracle/customized_templates
   ```

5. Extract the connector jar file from the zip file.

   ```
   unzip *.zip archives/*connector.jar
   ```

6. Extract the template files from the jar file. This command will extract the createEvent and updateEvent template files.

   ```
   $JAVA_HOME/bin/jar xf archives/*.jar createEvent_request.xsl
   updateEvent_request.xsl
   ```

7. Make a copy of the createEvent_request.xsl file and name it *custom_createEvent_request.xsl*.

8. Make a copy of the updateEvent_request.xsl file and name it *custom_updateEvent_request.xsl*

9. Open the custom_createEvent_request.xsl file in your text editor.

10. Change the appropriate sections to reflect the new mapping.

    Before Changes

    The code below shows the impacted sections in the file before the changes.

    ```
    <summary>
      <xsl:value-of select="a:SystemAttributes/a:EventName"/>
    </summary>
    ...
    <extended-fields>
      <!-- SCOM alert custom fields -->
      <!-- Uncomment fields to be set and replace "VALUE" with the actual
       value -->
      <!--
      <string-field name="CustomField1">VALUE</string-field>
      <string-field name="CustomField2">VALUE</string-field>
      <string-field name="CustomField3">VALUE</string-field>
    ```

```
       <string-field name="CustomField4">VALUE</string-field>
       <string-field name="CustomField5">VALUE</string-field>
       <string-field name="CustomField6">VALUE</string-field>
       <string-field name="CustomField7">VALUE</string-field>
       <string-field name="CustomField8">VALUE</string-field>
       <string-field name="CustomField9">VALUE</string-field>
       <string-field name="CustomField10">VALUE</string-field>
       -->
     </extended-fields>
```

After Changes

The code below shows the Extended Fields section in the file after the changes. The changes are shown in bold.

```
<!-- SCOM alert title -->
     <summary>Event From Oracle Enterprise Manager</summary>
     ...
     <extended-fields>
       <!-- SCOM alert custom fields -->
       <!-- Uncomment fields to be set and replace "VALUE" with the actual
        value -->
       <string-field name="CustomField1">
         <xsl:value-of select="a:TargetType"/>
       </string-field>
       <string-field name="CustomField2">
         <xsl:value-of select="a:TargetName"/>
       </string-field>
       <string-field name="CustomField3">
         <xsl:value-of select="a:MetricName"/>
       </string-field>
       <string-field name="CustomField4">
         <xsl:value-of select="a:MetricColumn"/>
       </string-field>
       <!--
       <string-field name="CustomField5">VALUE</string-field>
       <string-field name="CustomField6">VALUE</string-field>
       <string-field name="CustomField7">VALUE</string-field>
       <string-field name="CustomField8">VALUE</string-field>
       <string-field name="CustomField9">VALUE</string-field>
       <string-field name="CustomField10">VALUE</string-field>
       -->
     </extended-fields>
```

11. Save your changes after making the updates.
12. Open the custom_updateEvent_request.xsl file and make the same changes for the custom fields. In this case, you cannot set the alert name because it is only valid for creates.
13. Save your changes after making the updates.
14. To set up for testing, create the create.xml and update.xml files and open in a text editor.
15. Copy the contents of the XML document in the “Create Sample” section of Appendix B into the create.xml file and save.
16. Copy the contents of the XML document in the “Update Sample” section of Appendix B into the update.xml file and save.
17. Test the new templates using a utility that performs XSLT translations using each of the create.xml and update.xml files as the input XML document.

18. Examine the output XML document and verify the results are correct.

> **Note:** You can also use an XML client to send the XML output document to the SCOM web service to verify that it is a format accepted by the web service.

19. If the XML document has problems, make any necessary corrections and repeat steps 17 and 18.

20. At the OMS server system, run the following command to reregister your customized createEvent template.

```
emctl register_template connector
-t /home/oracle/customized_templates/custom_createEvent_request.xsl
-repos_pwd sysmanpass
-cname "SCOM R2"
-iname "createEvent"
-tname "Create Event Request"
-ttype 2
-d " This is the customized request xsl file for the createEvent method"
```

21. At the OMS server system, run the following command to reregister your customized updateEvent template.

```
emctl register_template connector
-t /home/oracle/customized_templates/custom_updateEvent_request.xsl
-repos_pwd sysmanpass
-cname "SCOM R2"
-iname "updateEvent"
-tname "Update Event Request"
-ttype 2
-d " This is the customized request xsl file for the updateEvent method"
```

6

Enabling SSL for HTTPS

This section contains information used in configuring the SCOM connector to use the HTTPS protocol.

6.1 Configuring Enterprise Manager to use SSL

If the SCOM web service was configured to run using the HTTPS protocol, you must perform the following steps to set up SSL:

1. Install a SSL certificate in the SCOM web service keystore. You must either install a self-signed certificate or install a certificate obtained from a Certificate Authority (CA).

   - To install a self-signed certificate, perform the steps specified in Section 6.2, "Installing a Self-Signed Certificate".
   - To install a certificate from a CA, perform the steps specified in Section 6.3, "Installing a Certificate from a Certificate Authority".

2. Import the SSL certificate from the SCOM web service keystore into the Enterprise Manager keystore as specified in Section 6.4, "Importing the Web Service Certificate into Enterprise Manager".

6.2 Installing a Self-Signed Certificate

Perform the following steps to generate and install a self-signed SSL certificate for the SCOM adapter web service:

1. Open a command prompt window and change the working directory to the adapters/conf directory in the SCOM web service installation directory.

2. Execute the following command to delete the default SSL entry from the SCOM web service keystore.

   Unix

   ```
   $JAVA_HOME/bin/keytool -delete -alias iwave -keypass iwavepw
   -storepass iwavepw -keystore keystore.jks
   ```

   Windows

   ```
   "%JAVA_HOME%\bin\keytool" -delete -alias iwave -keypass
   iwavepw -storepass iwavepw -keystore keystore.jks
   ```

3. Enter the following command to generate a new certificate and place it in the SCOM web service keystore. You will need to replace <*hostname*> with the host name or IP address of the system where the SCOM web service is installed.

> **Note:** The host name in the certificate must match the host name or IP address used by the web service. If they do not match, a failure will occur when Enterprise Manager tries to invoke the web service

Unix

```
$JAVA_HOME/bin/keytool -genkey -alias iwave -keyalg RSA
-keysize 1024 -dname "CN=<hostname>, OU=Development, O=iWave
Software, L=Frisco, ST=TX, C=US" -keypass iwavepw -storepass
iwavepw -keystore keystore.jks
```

Windows

```
"%JAVA_HOME%\bin\keytool" -genkey -alias iwave -keyalg RSA
-keysize 1024 -dname "CN=<hostname>, OU=Development, O=iWave
Software, L=Frisco, ST=TX, C=US" -keypass iwavepw -storepass
iwavepw -keystore keystore.jks
```

6.3 Installing a Certificate from a Certificate Authority

Perform the following steps to request and install a signed SSL certificate for the SCOM web service:

1. Request a certificate for the SCOM web service from a Certificate Authority, such as VeriSign.

   > **Note:** In the certificate request, make sure to specify the host name or IP address of the system where the SCOM web service is installed. The host name in the certificate must match the host name or IP address used by the web service. If they do not match, a failure will occur when Enterprise Manager tries to invoke the web service.

2. After you obtain the certificate from the Certificate Authority, perform the following steps to install the certificate:

   - Open a command prompt window and change the working directory to the adapters/conf directory in the SCOM web service installation directory.
   - Enter the following command to install the certificate, where <certificateFile> is the full path name of the file provided by the Certificate Authority:

     Unix

     ```
     $JAVA_HOME/bin/keytool -importcert -alias iwave -file
     <certificateFile> -keypass iwavepw -storepass iwavepw
     -keystore keystore.jks
     ```

     Windows

     ```
     "%JAVA_HOME%\bin\keytool" -importcert -alias iwave -file
     <certificateFile> -keypass iwavepw -storepass iwavepw
     -keystore keystore.jks
     ```

6.4 Importing the Web Service Certificate into Enterprise Manager

Perform the following steps to import the SCOM web service SSL certificate into the Enterprise Manager keystore.

1. Open a command prompt window and change the working directory to the adapters/conf directory in the SCOM web service installation directory.
2. Issue the following command to extract the SSL certificate from the SCOM web service keystore and place in the SCOMws.cer certificate file.

   Unix:

   ```
   $JAVA_HOME/bin/keytool -exportcert -rfc -alias iwave -file
   SCOMws.cer -keystore keystore.jks -storepass iwavepw
   ```

   Windows:

   ```
   "%JAVA_HOME%\bin\keytool" -exportcert –rfc -alias iwave -file
   SCOMws.cer -keystore keystore.jks -storepass iwavepw
   ```

3. Transfer the certificate file SCOMws.cer to the system where Enterprise Manager is installed.
4. Append the contents of the SCOMws.cer file to:

   ```
   $INSTANCE_HOME/sysman/config/b64LocalCertificate.txt
   ```

5. Ensure that only the following is appended to the b64LocalCertificate.txt file (that is, do not include blank lines or comments or any other special characters):

   -----BEGIN CERTIFICATE-----

   <<<Certificate in Base64 format>>>

   -----END CERTIFICATE-----

6. Restart OMS by running the following commands:

   ```
   emctl stop oms
   ```

   ```
   emctl start oms
   ```

   > **Note:** Do not run the emctl secure oms/agent command after adding the external certificate to the *b64LocalCertificate.txt* file. If you run the emctl secure command later, then repeat steps 4 through 6 to make sure the external certificate exists in the *b64certificate.txt* file.

7 Troubleshooting the Connector

This chapter provides information to assist in troubleshooting integration issues with Microsoft SCOM. The chapter focuses on troubleshooting issues in the web service front-end and the back-end Agent.

This chapter discusses the following topics:

- Diagnosing the Problem
- Troubleshooting Web Service Startup Errors
- JVM Errors
- Using the Correct URL for SCOM Web Service Operations
- Troubleshooting Web Service Operations Errors
- Errors from Oracle Enterprise Manager

Before you start the troubleshooting steps, you must insure that you have done the following:

1. Install the SCOM Connector as specified in Section 2.3, "Installing the Connector in Enterprise Manager".
2. Install and start the Oracle SCOM Agent as specified in Section 2.4, "Installing and Running the Oracle SCOM Agent".
3. Install, start, and test the SCOM Web service as specified in Section 2.5, "Installing the Microsoft SCOM Web Service".
4. Create a connector instance as specified in Section 3.1, "Creating a Connector Instance".
5. Configure the connector instance as specified in Section 3.2, "Configuring the Connector".
6. Set up one or more rules to forward events to the connector instance.

If all the actions above have been completed and the connector is not working, perform the steps in Section 7.1, "Diagnosing the Problem".

7.1 Diagnosing the Problem

Follow these steps to diagnose a problem.

1. Verify that the SCOM Web Service has been successfully started and the WSDL for the SCOM Web Service can be accessed from the machine where it is installed. Performing the following steps to do this:

a. Perform the steps in Section 7.4, "Using the Correct URL for SCOM Web Service Operations" to determine the URL used by the SCOM Web Service.

b. Open a browser and paste the URL from the previous step in the address window. Append *?WSDL* to the end of the URL and attempt to load the URL.

c. The WSDL should be loaded if the web service is operational.

Note: A WSDL is an XML file that describes the web service.

d. If the WSDL cannot be loaded, this indicates the web service had startup issues. See Section 7.2, "Troubleshooting Web Service Startup Errors" to diagnose the problem.

2. Verify that the WSDL for the SCOM Web Service can be accessed from the machine where the Enterprise Manager server is installed.

   a. Open a browser at the Enterprise Manager server and copy the URL from step 1-b above to the address window. The hostname for the URL will be localhost. Change localhost to the actual hostname or IP address of the machine where the SCOM Web Service is installed. If you specify a hostname, you must make sure that the hostname is recognized at the Enterprise Manager server machine. This can be done using the ping command.

   For example, if the Web Service is installed on the server with a hostname of SCOMServer01 and the URL listed in framework.log is:

   ```
   http://localhost:8080/services/SCOM2007/EventService
   ```

   The URL used at the Enterprise Manager server machine would be:

   ```
   http://SCOMServer01: 8080/services/SCOM2007/EventService
   ```

   b. Attempt to load the WSDL by appending ?WSDL at the end of the URL. If the WSDL cannot be loaded, either the hostname is not recognized at the Enterprise Manager machine or there is a connectivity issue between the two machines. If you specified a hostname, try using the IP address instead of the hostname in the URL. If it still will not load, you have a connectivity problem. You will need to consult with your IT department to resolve this issue.

3. Verify that the SCOM Connector specifies the correct URL for the createEvent and updateEvent operations.

   a. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator permissions.

   b. From the Enterprise Manager console, click **Setup**.

   c. Select **Extensibility** in the pane.

   d. Select **Management Connectors** in the left pane.

   e. The Management Connectors page appears, which shows the installed connectors.

   f. Click the **Configure** icon associated with the SCOM Connector. This invokes edit mode, enabling you to configure the connector.

   g. Verify that the URL identified in step 2-b is specified for the createEvent and updateEvent operations. Also verify the URLs listed for the setup, initialize, and uninitialize operations. The URL is basically the same as the URL listed above except it is for the SCOMService instead of the EventService. Listed below is the URL that would be used based on the previous example.

```
http://SCOMServer01: 8080/services/SCOM2007/SCOMService
```

h. If any of the operations are incorrect, change to the correct URL.

> **Note:** Do not append the WSDL to the end of the URL that is specified here.

i. Click **OK**.

4. If there were no errors found in the previous steps, this rules out connectivity issues between the connector and the SCOM Web Service. The problem must lie in the SCOM Web Service or the Oracle SCOM Agent. See Section 7.5, "Troubleshooting Web Service Operations Errors" for information on diagnosing errors.

7.2 Troubleshooting Web Service Startup Errors

To identify the cause of a startup failure, navigate to the adapters/log directory in the SCOM Web Service install directory and open the framework.log file in a text editor. Search for Exception to find any errors in the file. If the file does not exist, it indicates that there is a problem locating or executing the JVM. See Section 7.3, "JVM Errors" for information about resolving JVM issues.

Listed below are some possible Exceptions, an explanation of the root cause, and a description of the solution.

java.net.BindException: Address already in use: bind

This error indicates that the web service could not start because of a port conflict. There are two possible causes for this error.

1. Another application is using a port that the Web service is configured to use. If the web service is configured to use SSL, the port number is 8443. If it is not configured to use SSL, the port number is 8080.

   There are two possible solutions to this. You can change the other application to use a different port or you can change the SCOM Web Service to use a different port. To change the SCOM Web Service to use a different port, see Section A.2, "Changing Default Port Numbers".

2. There is an instance of the Web service already running. If this is the case then there is no change required. You should only run one instance of the Web service at a time.

org.springframework.beans.factory.BeanInitializationException: Could not load properties; nested exception is java.io.FileNotFoundException: … framework.properties (Permission denied)

This error indicates that the web service could not start because the permissions on the framework.properties file in the conf directory were not set correctly.

To solve the problem, change the permissions to give the account or group under which the SCOM Web Service runs read and execute permissions.

For any other startup errors, consult Oracle Support.

7.3 JVM Errors

The SCOM Web Service requires version 1.6 of the JVM. If there are multiple versions of the JVM installed on the machine, it is possible that an older version of the JVM is being executed whenever the web service starts.

On UNIX systems, the JAVA_HOME environment variable must be set to the directory where Java 1.6 is installed in the shell where the web service is started. To properly start the web service on a UNIX platform, perform the following:

1. Set the JAVA_HOME environment variable to the Java 1.6 install directory.
2. Navigate to the adapters/bin subdirectory in the Web service install directory.
3. Execute the ./service.sh start command.

On Windows systems, perform the following to insure that Java 1.6 is used when starting the web service.

1. Navigate to the adapters/bin subdirectory in the Web service install directory.
2. Run the iWaveAdaptersw.exe executable.
3. Click the Java tab.
4. Be sure the Use Default check box is not checked.
5. In the Java Virtual Machine box, specify the path to the jvm.dll file in the Java 1.6 install directory.
6. Click **OK**.

7.4 Using the Correct URL for SCOM Web Service Operations

Perform the following steps to identify and configure the connector to use the correct URL for SCOM Web Service operations.

1. Open a command terminal on the system where the SCOM web service is installed.
2. Change the working directory to the adapters/log directory in the SCOM web service installation directory.
3. Open the framework.log file in a text editor.
4. Go to the bottom of the file and search backwards for the string *Setting the server's publish address to be*. Continue searching backwards until you find the URL that contains EventService.

   You should specify the URL listed here for the createEvent and updateEvent operations, with one exception. You must replace the host name of localhost in the URL with the actual host name or IP address of the system where the SCOM Web Service is installed.
5. Go to the bottom of the file and search backwards for the string *Setting the server's publish address to be*. Continue searching backwards until you find the URL that contains SCOMService.

   You should specify the URL listed here for the setup, initialize and uninitialize and operations, with one exception.You must replace the host name of localhost in the URL with the actual host name or IP address of the system where the SCOM Web Service is installed.
6. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator permissions.

7. From the Enterprise Manager console, click **Setup**.
8. Select **Extensibility** in the pane.
9. Select **Management Connectors** in the left pane.
10. Click the name of the SCOM Connector.

    The General tab of the Configure Management Connector page appears.
11. Verify that the URL identified in step 4 is specified for the creatEvent and updateEvent operations.
12. Verify that the URL identified in step 5 is specified for the setup, initialize and uninitialize operations.
13. If any of the operations are incorrect, change to the correct URL and click **OK**.

7.5 Troubleshooting Web Service Operations Errors

Perform the following diagnostic steps if messages are not being generated or updated as expected in SCOM.

1. Verify that the event that was triggered is referenced in a rule that forwards events to the SCOM connector.
2. Verify that the OracleEnterpriseManager.Alert.Creator Management Pack has been imported into the SCOM server:

    a. Open the Ops Mgr 2007 console window and log in.

    b. In the Administration pane of the OpsMgr console, select **Administration**, then **Management Packs**.

    c. Verify that the "OracleEnterpriseManager Alert Creator" Management Pack is listed.

    d. If the pack is not listed, import it as specified in Section 2.4.4, "Installing the Alert Creator Management Pack".
3. Determine the error that Oracle Enterprise Manager has reported.

    a. To do this you need to examine the log file for errors. Perform the following steps to locate errors in the log file:

    1.) Open the emoms_pbs.trc file in a text editor. The file is located in the log directory at the following location:

    ```
    <EM_INSTANCE_BASE>/em/<OMS_NAME>/sysman/log/
    ```

    Where, <EM_INSTANCE_BASE> is the OMS Instance Base directory. By default, the

    OMS Instance Base directory is gc_inst, which is present under the parent directory of the Oracle Middleware Home.

    For example, if the Oracle Middleware Home is /u01/app/Oracle/Middleware,

    then the instance base directory is */u01/app/Oracle/Middleware/gc_inst*, and the log and trace files are available in */u01/app/Oracle/Middleware/gc_inst/em/EMGC_OMS1/sysman/log/* directory path.

    2.) Go to the bottom of the file and search backwards for *Caused by*. Lines that start with Caused by contain error information. The error information comes after the text in the line that reads *oracle.sysman.emSDK.webservices.outbound.WSInvokeException: caught WebServiceException :.*

4. Diagnose the problem based on the error information. See Section 7.6, "Errors from Oracle Enterprise Manager" for information on troubleshooting common error events.

7.6 Errors from Oracle Enterprise Manager

This section provides cause and solution information on troubleshooting common error messages. Find the error message in Table 7–1 that matches your error message, then refer to the corresponding section(s) indicated under Possible Cause for instructions to diagnose and correct the problem.

Table 7–1 Enterprise Manager Error Messages

Error Message	Possible Cause
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Bad response: 403 Forbidden from url …	Invalid Web Service Credentials
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Message send failed: sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target	SSL Not Configured in Enterprise Manager
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Message send failed: Connection refused	SCOM Web Service Down
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Message send failed: No route to host	Invalid IP Address
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Bad response: 404 Not Found from url …	Invalid Port Number or Invalid URL Path
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Message send failed: Connection timed out	Firewall Blocking Access
javax.xml.soap.SOAPException: javax.xml.soap.SOAPException: Message send failed: hostname	Unknown Host
javax.xml.transform.TransformerConfigurationException: Could not compile stylesheet	Invalid XML Format
Unable to reconnect to server after being disconnected	SCOM Server Not Operational
ERROR - Could not connect to the server <hostname> because it is not operational	SCOM Server Not Operational
ERROR - Could not login to the server because the account was invalid or has insufficient permissions	Invalid SCOM API Credentials or Wrong SCOM API Permissions
ERROR occurred invoking SCOM connector to insert event for null	SCOM Agent Not Operational or Alert Created Timeout

Table 7–1 (Cont.) Enterprise Manager Error Messages

Error Message	Possible Cause
javax.xml.ws.WebServiceException: org.apache.cxf.service.factory.ServiceConstructionException: Failed to create service	SCOM Agent Configuration
Request failed because the specified management pack could not be found	Management Pack Missing
&#13;&#10;Successfully inserted the event but timed out waiting for the alert to be created	Alert Created Timeout

Invalid Web Service Credentials

Cause

The user name or password for accessing the SCOM web service is incorrect.

Solution

1. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator privileges.
2. From the Enterprise Manager console, click **Setup**.
3. Select **Extensibility** in the pane.
4. Select **Management Connectors** from the menu. This will display the Management Connectors page.
5. Click the name of the SCOM Connector.

   This invokes edit mode, enabling you to configure the connector.
6. Correct the SCOM Web Service Username and SCOM Web Service Password fields, then click **OK**.

SSL Not Configured in Enterprise Manager

Cause

The SSL handshake between the Oracle Enterprise Manager Connector Framework and the SCOM web service failed. This failure occurs because Oracle Enterprise Manager is not configured correctly with the SSL certificate for the SCOM web service. The SSL certificate the SCOM web service uses must be imported into the Enterprise Manager key store. The certificate is either missing from the key store or does not match the SSL certificate provided by the SCOM web service.

Solution

Import the SSL certificate from the SCOM web service into the Enterprise Manager key store. See Section 6.1, "Configuring Enterprise Manager to use SSL" for details on setting up Oracle Enterprise Manager with the SCOM SSL certificate.

SCOM Web Service Down

Cause

The SCOM web service is down.

Solution

Perform the following steps to check the status of the web service and start it if necessary.

If the SCOM web service is installed on a Unix system:

1. Open a command terminal on the system where the SCOM web service is installed.
2. Change the working directory to the adapters/bin directory in the SCOM web service installation directory.
3. Enter the following command:

   ./service.sh status
4. If the command indicates that the service is not running, enter the following command:

   ./service.sh start

If the SCOM web service is installed on a Windows system:

1. Open a command terminal on the system where the SCOM web service is installed.
2. Change the working directory to the *adapters/log* directory in the SCOM web service installation directory.
3. Open the framework.log file in a text editor.
4. Go to the bottom of the file and search backwards for the string iWave Adapter Framework. If the last occurrence found is iWave Adapter Framework Started, this indicates that the web service is started.
5. If the web service is not started, start the web service as specified in Section 2.5.2.3, "Running the Web Service on Windows".

Invalid IP Address

Cause

The IP address specified in the URL is invalid or the network is down.

Solution

Verify that the hostname/IP address configured for the connector is correct:

1. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator privileges.
2. From the Enterprise Manager console, click **Setup**.
3. Select **Extensibility** in the pane.
4. Select **Management Connectors** from the menu. This will display the Management Connectors page.
5. Click the name of the SCOM Connector.

   This invokes edit mode, enabling you to configure the connector.
6. Verify that the hostname/IP address specified in the URL for the createEvent and updateEvent operations are correct.
7. If the hostname/IP address is incorrect, provide the correct value, then click **OK**.

If the URLs specify a host name, make sure that the host name resolves to the correct IP address. To determine the IP address of the host name, issue the ping <hostname>

command, where <hostname> is the actual host name. This lists the IP address that was resolved for the host name. If this is incorrect, the system administrator needs to investigate why it is incorrect.

If the hostname/IP address appears to be correct, try to ping the system where the SCOM web service is installed using the hostname/IP address. If the ping fails, the system administrator needs to investigate why there is no connectivity.

Invalid Port Number

Cause

The port number specified in the URL is invalid.

Solution

Verify that the port number configured for the connector is correct:

1. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator privileges.
2. From the Enterprise Manager console, click **Setup**.
3. Select **Extensibility** in the pane.
4. Select **Management Connectors** from the menu. This will display the Management Connectors page.
5. Click the name of the SCOM Connector.

   This invokes edit mode, enabling you to configure the connector.
6. Verify that the port number specified in the URL for the createEvent, updateEvent, setup, initialize, and uninitialize operations are correct.
7. If the port number is incorrect, provide the correct value and click **OK**.

Invalid URL Path

Cause

The web service received the request and rejected it because an invalid path was specified in the URL.

Solution

Perform the following steps to test the URL the connector is using.

1. Log in to the Oracle Enterprise Manager console with an account that has Super Administrator privileges.
2. From the Enterprise Manager console, click **Setup**.
3. Select **Extensibility** in the pane.
4. Select **Management Connectors** from the menu. This will display the Management Connectors page.
5. Click the name of the SCOM Connector.

   This invokes edit mode, enabling you to configure the connector.
6. Select and copy the URL specified for the createEvent operation.
7. Open an internet browser on the system where the Oracle Enterprise Manager server is installed.
8. In the address window, enter the URL that was copied in step 6 above. Add *?wsdl* to the end of the URL. The URL should appear similar to the following example:

```
http://[Hostname]:8080/services/SCOM2007/EventService?wsdl
```

[Hostname] is the actual host name or IP address where the SCOM web service is installed.

If the WSDL is loaded, this confirms that the URL is correct. If it fails to load, there is a problem with the URL. Perform the steps specified in Section 7.4, "Using the Correct URL for SCOM Web Service Operations" to configure the connector to use the correct URL.

Firewall Blocking Access

Cause

A firewall is blocking access to the system where the SCOM Web Service is installed.

Solution

Contact your IT department to give Enterprise Manager access to the port used by the SCOM Web Service. Perform the steps specified in Section 7.4, "Using the Correct URL for SCOM Web Service Operations" to determine the URL used by the SCOM Web Service. The port number specified in the URL is the port number the IT department should open in the firewall.

Unknown Host

Cause

The system does not recognize the host name specified in the URL.

Solution

You can use the following options to address this issue:

- Coordinate with the system administrator to change the system configuration to recognize the host name.
- Specify the IP address in the URL instead of the host name. To do this, perform the following steps:
  1. Determine the IP address of the system where the SCOM web service is installed.
  2. Log in to the Oracle Enterprise Manager console by entering a user name with a ‘Super Administrator’ role, entering the appropriate password, then click **Login**.
  3. Click the **Setup** link at the top right part of the window. The Overview of Setup page appears.
  4. Click the **Management Connectors** link on the left side of the window. The Management Connectors page appears, which shows the installed connectors.
  5. Click on the **Configure** icon associated with the Microsoft SCOM Connector. This invokes edit mode, enabling you to configure the connector.
  6. Change the host name to the IP address in the URL specified for the createEvent and updateEvent operations.
  7. Click **OK**.

Invalid XML Format

Cause

The connector framework could not process the request because the XSL file was formatted incorrectly. This problem should not occur unless the connector has been customized.

Solution

Examine any changes made to the XSL template files for mistakes that could have caused the problem. If you can't find the problem manually, load the XSL in a utility that performs XML validation.

SCOM Server Not Operational

Cause

The SCOM Agent could not insert the alert into SCOM because the wrong host name is configured for SCOM or the SCOM server is down.

Solution

Perform the following steps to determine and correct the root cause of the problem:

1. Verify that the host name or IP address listed in the error message is correct for the RMS system. If the host name or IP address are incorrect, perform the following steps to correct the configuration:
   a. Open Windows Explorer on the system where the SCOM Agent is located.
   b. Navigate to the bin directory in the SCOM Agent installation directory.
   c. Run the SCOMAgentConfig.exe utility to start the SCOM Agent Configuration Tool.
   d. Click **Load** to open a directory navigation window.
   e. Navigate to the SCOM Agent installation directory and open the SCOMAgent.cfg file.
   f. Click the **Management Groups** tab, then click **Edit** to display the Edit Management Group window.
   g. Correct the hostname/IP address in the Server field, then click **Update**.
   h. Click **Save** to save the changes to the configuration file.
   i. Click **Exit** to exit the utility.
   j. Stop and restart the SCOM Agent in IIS.
2. Verify that the following OpsMgr services are running based on the version of SCOM:

   preR2
   OpsMgr Config Service
   OpsMgr Health Service
   OpsMgr SDK Service
   R2
   System Center Data Access
   System Center Management
   System Center Management Configuration

Invalid SCOM API Credentials

Cause

The SCOM Agent could not send the alert to the SCOM server, because the credentials configured for accessing the SCOM API are invalid.

Solution

Perform the following steps to change the credentials for accessing the SCOM API:

1. Open Windows Explorer on the system where the SCOM Agent is located.
2. Navigate to the bin directory in the SCOM Agent installation directory.
3. Run the SCOMAgentConfig.exe utility to start the SCOM Agent Configuration Tool.
4. Click **Load** to open a directory navigation window.
5. Navigate to the SCOM Agent installation directory and open the SCOMAgent.cfg file.
6. Click the **Management Groups** tab, then click **Edit** to display the Edit Management Group window.
7. Correct the credential information in the Domain, Username, and Password fields, then click **Update**.
8. Click **Save** to save the changes to the configuration file.
9. Click **Exit** to exit the utility.
10. Stop and restart the SCOM Agent in IIS.

Wrong SCOM API Permissions

Cause

The SCOM Agent could not send the alert to the SCOM server, because the credentials configured for accessing the SCOM API do not have sufficient permissions.

Solution

Refer to Section 2.4.2, "Setting Up the Agent Account". This section provides the steps required to set up the account for accessing the SCOM API.

SCOM Agent Not Operational

Cause

The web service could not create an alert in SCOM because the SCOM Agent is not operational.

Solution

Open IIS Manager on the system where the SCOM Agent was installed, and start the web site for the Agent.

SCOM Agent Configuration

Cause

The web service could not connect to the SCOM Agent because the web service has an invalid configuration parameter. Either the URL for the SCOM Agent is incorrect or the credentials for accessing the SCOM Agent are invalid.

Solution

1. Verify that the URL for the SCOM Agent is correct. You should specify the URL that was provided at the end of the SCOM Agent installation. Note that if the host name in the URL is localhost and you are accessing it from another system, you need to replace localhost with the host name or IP address of the SCOM Agent installation machine.

   If you do not know the URL, you can determine it as follows:

   - If the SCOM Agent was installed as a web site, the address is:

     ```
     http://<IP>:<port>/Service.asmx
     ```

     ... where <IP> is the IP address, and <port> is the port number specified when installing the Agent.

   - If the SCOM Agent was installed as a virtual directory, the address is:

     ```
     http://<IP>:<port>/<vdir>/Service.asmx
     ```

     ... where <IP> is the IP address, <port> is the port number for the web service where the agent was installed, and <vdir> is the virtual directory name specified for the Agent.

2. Select a user name and password that are valid on the system where the SCOM Agent was installed.

3. Open a command window and change the working directory to adapters\endpoints\SCOM2007 in the SCOM web service installation directory.

4. Rerun the SCOM Web Service installer using the URL and credentials from the preceding steps. See Section 2.5.1.2, "Installing the Web Service on Unix" or Section 2.5.2.2, "Installing the Web Service on Windows", depending on your platform, for the procedure.

Management Pack Missing

Cause

The web service could not create an alert in SCOM because the OracleEnterpriseManager.Alert.Creator management pack has not been imported into SCOM.

Solution

Refer to Section 2.4.4, "Installing the Alert Creator Management Pack" for the steps required to import the management pack into SCOM.

Alert Create Timeout

Cause

The web service was able to insert an event in SCOM, but an alert was not created within the timeout period. This likely indicates that an error occurred in the alert generating rule and it was unloaded by SCOM. Whenever this occurs, the Ops Mgr Health Service generates an error followed by a warning in the Operations Manager log. The error entry begins with the following message:

```
A module reported an error 0x80070057 from a callback which was running as part of
rule "Create.Default.Alert" running for instance "OracleEnterpriseManager Event
Source" with id ...
```

The warning entry begins with the following message:

```
Summary: 1 rule(s)/monitor(s) failed and got unloaded, 1 of them reached the
failure limit that prevents automatic reload ...
```

Note: This situation should not occur if the default SCOM connector configuration files are used. The only known way this can occur is if the SCOM Agent web service is directly accessed and an invalid value is passed for the Priority or Severity fields.

Solution

Restart the Windows service named “Ops Mgr Health Service” on the RMS system.

A

Customizing Microsoft SCOM

This appendix describes the SCOM mappings and various other configuration defaults and contains the following sections:

- XML Format of Microsoft SCOM Alerts
- Changing Default Port Numbers
- Changing the Default Custom Field
- Changing SCOM API Connection Parameters

A.1 XML Format of Microsoft SCOM Alerts

Example A–1 represents the format that the Microsoft SCOM web service expects for creating new alerts in Microsoft SCOM. Example A–2 represents the format that the Microsoft SCOM web service expects for updating alerts in Microsoft SCOM.

Example A–1 Sample Create Format for Microsoft SCOM Web Service

```
<iwaveaf:create xmlns:iwaveaf="http://iwavesoftware.com/services/
  adapter-framework">
  <event>
    <summary></summary>
    <description></description>
    <severity></severity>
    <priority></priority>
    <extended-fields>
      <string-field name="CustomField1"></string-field>
      ...
      <string-field name="CustomField10"></string-field>
    </extended-fields>
  </event>
</iwaveaf:create>
```

Example A–2 Sample Update Format for Microsoft SCOM Web Service

```
<iwaveaf:update xmlns:iwaveaf="http://iwavesoftware.com/services/
adapter-framework">
  <event>
    <identifier></identifier>
    <status></status>
    <extended-fields>
      <string-field name="AlertHistory"></string-field>
      <string-field name="CustomField1"></string-field>
      …
      <string-field name="CustomField10"></string-field>
    </extended-fields>
  </event>
</iwaveaf:update>
```

A.1.1 Mappings Between XML Format and Alert Field Names

Table A–1 identifies the mappings between the Microsoft SCOM alert field names and the XML format that the Microsoft SCOM web services uses when creating an alert in SCOM. Table A–2 identifies the mappings between the Microsoft SCOM alert field names and the XML format that the Microsoft SCOM web services uses when updating an alert in SCOM.

The XML document presented to the Microsoft SCOM web service must have the corresponding fields set. Fields denoted with an asterisk (*) are optional. This must be handled in the appropriate translation file identified in Table A–1.

Table A–1 Create Alert Attributes and XML Path Mappings

Microsoft SCOM Slot Names	XML Path	Attribute Name	Attribute Value
Name	/create/event/summary		
Description	/create/event/severity		
Priority	/create/event/priority		
* Custom Field 1	/create/event/extended-fields/string-field	name	CustomField1
* Custom Field 2	/create/event/extended-fields/string-field	name	CustomField2
* Custom Field 3	/create/event/extended-fields/string-field	name	CustomField3
* Custom Field 4	/create/event/extended-fields/string-field	name	CustomField4
* Custom Field 5	/create/event/extended-fields/string-field	name	CustomField5
* Custom Field 6	/create/event/extended-fields/string-field	name	CustomField6
* Custom Field 7	/create/event/extended-fields/string-field	name	CustomField7
* Custom Field 8	/create/event/extended-fields/string-field	name	CustomField8
* Custom Field 9	/create/event/extended-fields/string-field	name	CustomField9
* Custom Field 10	/create/event/extended-fields/string-field	name	CustomField10

Table A–2 Update Alert Attributes and XML Path Mappings

Microsoft SCOM Slot Names	XML Path	Attribute Name	Attribute Value
Id	/update/event/identifier		
Resolution State	/update/event/status		
* Custom Field 1	/update/event/extended-fields/string-field	name	CustomField1

Table A–2 (Cont.) Update Alert Attributes and XML Path Mappings

Microsoft SCOM Slot Names	XML Path	Attribute Name	Attribute Value
* Custom Field 2	/update/event/extended-fields/string-field	name	CustomField2
* Custom Field 3	/update/event/extended-fields/string-field	name	CustomField3
* Custom Field 4	/update/event/extended-fields/string-field	name	CustomField4
* Custom Field 5	/update/event/extended-fields/string-field	name	CustomField5
* Custom Field 6	/update/event/extended-fields/string-field	name	CustomField6
* Custom Field 7	/update/event/extended-fields/string-field	name	CustomField7
* Custom Field 8	/update/event/extended-fields/string-field	name	CustomField8
* Custom Field 9	/update/event/extended-fields/string-field	name	CustomField9
* Custom Field 10	/update/event/extended-fields/string-field	name	CustomField10
* Alert History	/update/event/extended-fields/string-field	name	AlertHistory

A.1.2 Extended Fields

Extended fields pass information for fields that are not mapped. An extended field is defined as a <string-field/> element that is a child of the extended-fields node. The name of the field is specified in the name attribute, and the value of the field is specified as the element value.

A.2 Changing Default Port Numbers

In most cases, you can use the default port numbers that the Microsoft SCOM web service uses. However, if there are any conflicts with existing applications, you need to change the port numbers.

8080 is the default port number for HTTP communication, and 8443 is the default port for HTTPS communication. To change the port number, perform the steps in the appropriate subsection.

A.2.1 Changing the Web Service Port (8080)

The SCOM web service uses port 8080 as the default port for communication when the web service is configured at installation to use HTTP (no SSL). To change the port number to a different value, perform the following steps at the system where the SCOM web services are installed. Replace <SCOM_INSTALL> with the directory where the SCOM web services are installed.

1. Open a command prompt window and change the working directory to:

   <SCOMWS_INSTALL>/adapters/conf

2. Make a back-up copy of the framework.properties file, then open the file with a text editor.
3. Replace all references to the old port number with the new port number, then save the file.
4. Restart the web service as instructed in Section 2.5.1.3, "Running the Web Service on Unix" and Section 2.5.2.3, "Running the Web Service on Windows".

Perform the following steps to change the URL the SCOM connector is using:

1. From the Enterprise Manager console, click **Setup**.
2. Select **Extensibility** in the pane.
3. Select **Management Connectors** in the left pane.
4. Click the name of the SCOM Connector.

   The General tab of the Configure Management Connector page appears.
5. Change the URLs listed in the Web Service End Points section to use the new port number.
6. Click **OK** to save your changes.

A.2.2 Changing the Web Service Port (8443)

Contact Oracle support for assistance in switching the default SSL port 8443 to a different port.

A.3 Changing the Default Custom Field

The SCOM Agent requires that a custom field be set aside for its exclusive use. By default, alerts that the SCOM Agent creates will have custom field 10 set to the Guid of the event that triggered the alert. If this presents a problem, perform the following steps to change the SCOM Agent to use a different custom field.

1. Change the OracleEnterpriseManager.Alert.Creator.xml management pack file to place the event Guid in a different custom field:

   a. On the system where the SCOM Agent is installed, navigate to the directory where the OracleEnterpriseManager.Alert.Creator.xml file is located.

   b. Open the file with a text editor and search for Custom10. You should find the following line in the file:

   ```
   <Custom10>$Data/EventOriginId$</Custom10>
   ```

   c. Change “$Data/EventOriginId$” to “$Data/Params/Param[20]$”.

   d. Locate the tag for the custom field that the SCOM Agent will use, and change the contents to “$Data/EventOriginId$”.

   e. Save the file and exit the editor.
2. Import the updated management pack into SCOM as described in Section 2.4.4, "Installing the Alert Creator Management Pack".
3. Change the SCOM Web Service configuration to search for the event Guid in a different custom field:

   a. On the system where the SCOM Web Service is installed, navigate to the adapters/conf directory in the SCOM Web Service installation directory.

   b. Open the framework.properties file with a text editor and search for the scom2007.connector.response.query property.

   c. Change CustomField10 in the search string to the custom field number the management pack is setting.

   d. Save the file and exit the editor.

4. Stop and start the SCOM Web Service.

   Refer to the appropriate section in Chapter 2 based on the platform where the SCOM Web Service is installed. For Unix platforms, see Running the Web Service on Unix. For Windows platforms, see Running the Web Service on Windows.

A.4 Changing SCOM API Connection Parameters

When installing the SCOM Agent, you must configure the connection parameters used for accessing the SCOM API. The installation includes the SCOMAgentConfig utility you can use to change the Agent configuration parameters.

Perform the following steps to change the connection parameters:

1. Open Windows Explorer on the system where the SCOM Agent is installed.
2. Navigate to the bin directory in the SCOM Agent installation directory.
3. Run the SCOMAgentConfig.exe utility. This action starts the SCOM Agent Configuration Tool.
4. Click **Load**. This action opens a directory navigation window.
5. Navigate to the SCOM Agent installation directory and open the SCOMAgent.cfg file.
6. Click the **Management Groups** tab and click **Edit**. This action displays the Edit Management Group window.
7. Change the desired information in the following fields listed, then click **Update**:

   Name
   Server
   Domain
   Username
   Password

8. Click **Save** to save the changes to the configuration file.
9. Click **Exit** to exit the utility.
10. Stop and restart the SCOM Agent in IIS.

B

Enterprise Manager Event Samples

This appendix contains samples that show the XML format of events from Enterprise Manager that conform to the schemas defined in Appendix C, "Enterprise Manager Event Format". These samples can be used to help identify the fields to map when customizing the event templates. They can also be used to test your customizations once they have been built.

B.1 Create Sample

Listed below is a sample of the XML format generated by Enterprise Manager for SCOM create operations.

```
<EMEvent xmlns="http://xmlns.oracle.com/sysman/connector">
    <ConnectorGUID>2C08969483F141F1B7482F376935C0AC</ConnectorGUID>
    <NotificationRuleOwner>SYSMAN</NotificationRuleOwner>
    <NotificationRuleName>SCOM R2 Rule</NotificationRuleName>
    <ConnectorVariable>
        <VariableName>REGISTRATION_ID</VariableName>
        <VariableValue>Dummy</VariableValue>
    </ConnectorVariable>
    <Property>
        <Name>Notification_Method_Name</Name>
        <Value>SCOM R2</Value>
    </Property>
    <Property>
        <Name>AuthenticationType</Name>
        <Value>HTTPBasicAuthentication</Value>
    </Property>
    <Property>
        <Name>Registered</Name>
        <Value>yes</Value>
    </Property>
    <Property>
        <Name>IsNewTargetType</Name>
        <Value>yes</Value>
    </Property>
    <Property>
        <Name>Initialized</Name>
        <Value>yes</Value>
    </Property>
    <SystemAttributes>
        <EventClass>metric_alert</EventClass>
        <EventID>B69EC6D3F8B6A48EE040020A78016095</EventID>
        <SequenceID>B69EC6D3F8B7A48EE040020A78016095</SequenceID>
        <ReportedDate>2012-01-17T17:02:08.000-06:00</ReportedDate>
        <DisplayTZ>America/Chicago</DisplayTZ>
        <EventName>Load:cpuUtil</EventName>
        <Severity>Critical</Severity>
        <SeverityCode>CRITICAL</SeverityCode>
        <SourceInfo>
            <SourceObjInfo>
                <ObjID>D93A2E4E1622013D74D9534ECB090944</ObjID>
                <ObjName>orcsvr1.iwavesoftware.com</ObjName>
                <ObjOwner>SYSMAN</ObjOwner>
                <SourceObjType>TARGET</SourceObjType>
                <SourceObjSubType>host</SourceObjSubType>
            </SourceObjInfo>
            <TargetInfo>
                <TargetGUID>D93A2E4E1622013D74D9534ECB090944</TargetGUID>
                <TargetName>orcsvr1.iwavesoftware.com</TargetName>
                <TargetType>host</TargetType>
                <TargetTypeLabel>Host</TargetTypeLabel>

<TargetURL>https://orcsvr1.iwavesoftware.com:7801/em/redirect?pageType=TARGET_
HOMEPAGE&amp;targetName=orcsvr1.iwavesoftware.com&amp;targetType=host</TargetURL>
                <TargetProperty>
                    <Name>Target_Host</Name>
                    <Value>orcsvr1.iwavesoftware.com</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Target Version</Name>
                    <Value>5.5.0.0.0</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Line of Business</Name>
                    <Value>Software</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Location</Name>
                    <Value>Frisco, TX</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Comment</Name>
                    <Value>Testing target properties</Value>
                </TargetProperty>
            </TargetInfo>
        </SourceInfo>
        <Message>CPU Utilization is 7.777%, crossed warning (0.1) or critical
(0.2)
            threshold.</Message>

<EventURL>https://orcsvr1.iwavesoftware.com:7801/em/redirect?pageType=sdk-core-eve
nt-console-detailEvent&amp;issueID=B69EC6D3F8B7A48EE040020A78016095</EventURL>
        <AutoClose>true</AutoClose>
        <EventCategory>Capacity</EventCategory>
    </SystemAttributes>
    <EventClassSpecificAttributes>
        <StringAttribute>
            <Name>is_thresholdable</Name>
            <Value>1</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>coll_name</Name>
            <Value>LoadLinux</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_metric_extension</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>cycle_guid</Name>
            <Value>B69EC6D3F8B3A48EE040020A78016095</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_remote</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_type</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_guid</Name>
            <Value>0C71A1AFAC2D7199013837DA35522C08</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>num_keys</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>key_value</Name>
            <Value> </Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description_nlsid</Name>
        </StringAttribute>
        <StringAttribute>
            <Name>value</Name>
            <Value>7.777</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_long_running</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group</Name>
            <Value>Load</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_udm</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description</Name>
```

```
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column_nlsid</Name>
            <Value>host_load_cpuIdle</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column</Name>
            <Value>cpuUtil</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit_nlsid</Name>
            <Value>em__sys__standard_percent</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit</Name>
            <Value>%</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group_nlsid</Name>
            <Value>host_load</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>severity_guid</Name>
            <Value>B69EC6D3F8B3A48EE040020A78016095</Value>
        </StringAttribute>
    </EventClassSpecificAttributes>
</EMEvent>
```

B.2 Update Sample

Listed below is a sample of the XML format generated by Enterprise Manager for SCOM update operations.

```
<EMEvent xmlns="http://xmlns.oracle.com/sysman/connector">
    <ConnectorGUID>2C08969483F141F1B7482F376935C0AC</ConnectorGUID>
    <ExternalEventID>368d9bea-a659-4086-b239-0f7cf650d9df</ExternalEventID>
    <NotificationRuleOwner>SYSMAN</NotificationRuleOwner>
    <NotificationRuleName>SCOM R2 Rule</NotificationRuleName>
    <ConnectorVariable>
        <VariableName>REGISTRATION_ID</VariableName>
        <VariableValue>Dummy</VariableValue>
    </ConnectorVariable>
    <Property>
        <Name>Notification_Method_Name</Name>
        <Value>SCOM R2</Value>
    </Property>
    <Property>
        <Name>AuthenticationType</Name>
        <Value>HTTPBasicAuthentication</Value>
    </Property>
    <Property>
        <Name>Registered</Name>
        <Value>yes</Value>
    </Property>
    <Property>
        <Name>IsNewTargetType</Name>
        <Value>yes</Value>
    </Property>
    <Property>
        <Name>Initialized</Name>
        <Value>yes</Value>
    </Property>
    <SystemAttributes>
        <EventClass>metric_alert</EventClass>
        <EventID>B6A5393A91D232A0E040020A7801445B</EventID>
        <SequenceID>B69EC6D3F8B7A48EE040020A78016095</SequenceID>
        <ReportedDate>2012-01-17T17:03:49.000-06:00</ReportedDate>
        <DisplayTZ>America/Chicago</DisplayTZ>
        <EventName>Load:cpuUtil</EventName>
        <Severity>Warning</Severity>
        <SeverityCode>WARNING</SeverityCode>
        <SourceInfo>
            <SourceObjInfo>
                <ObjID>D93A2E4E1622013D74D9534ECB090944</ObjID>
                <ObjName>orcsvr1.iwavesoftware.com</ObjName>
                <ObjOwner>SYSMAN</ObjOwner>
                <SourceObjType>TARGET</SourceObjType>
                <SourceObjSubType>host</SourceObjSubType>
            </SourceObjInfo>
            <TargetInfo>
                <TargetGUID>D93A2E4E1622013D74D9534ECB090944</TargetGUID>
                <TargetName>orcsvr1.iwavesoftware.com</TargetName>
                <TargetType>host</TargetType>
                <TargetTypeLabel>Host</TargetTypeLabel>

<TargetURL>https://orcsvr1.iwavesoftware.com:7801/em/redirect?pageType=TARGET_
HOMEPAGE&amp;targetName=orcsvr1.iwavesoftware.com&amp;targetType=host</TargetURL>
                <TargetProperty>
                    <Name>Target_Host</Name>
                    <Value>orcsvr1.iwavesoftware.com</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Target Version</Name>
                    <Value>5.5.0.0.0</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Line of Business</Name>
                    <Value>Software</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Location</Name>
                    <Value>Frisco, TX</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Comment</Name>
                    <Value>Testing target properties</Value>
                </TargetProperty>
            </TargetInfo>
        </SourceInfo>
        <Message>CPU Utilization is 9.68%, crossed warning (0.1) or critical (99)
            threshold.</Message>

<EventURL>https://orcsvr1.iwavesoftware.com:7801/em/redirect?pageType=sdk-core-eve
nt-console-detailEvent&amp;issueID=B69EC6D3F8B7A48EE040020A78016095</EventURL>
        <AutoClose>true</AutoClose>
        <EventCategory>Capacity</EventCategory>
```

```
</SystemAttributes>
<EventClassSpecificAttributes>
    <StringAttribute>
        <Name>is_thresholdable</Name>
        <Value>1</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>coll_name</Name>
        <Value>LoadLinux</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>is_metric_extension</Name>
        <Value>0</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>metric_column_resbundle</Name>
        <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>metric_description_resbundle</Name>
        <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>unit_resbundle</Name>
        <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>cycle_guid</Name>
        <Value>B69EC6D3F8B3A48EE040020A78016095</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>is_remote</Name>
        <Value>0</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>metric_type</Name>
        <Value>0</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>metric_guid</Name>
        <Value>0C71A1AFAC2D7199013837DA35522C08</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>num_keys</Name>
        <Value>0</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>key_value</Name>
        <Value> </Value>
    </StringAttribute>
    <StringAttribute>
        <Name>metric_description_nlsid</Name>
    </StringAttribute>
    <StringAttribute>
        <Name>value</Name>
        <Value>9.68</Value>
    </StringAttribute>
    <StringAttribute>
        <Name>is_long_running</Name>
        <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group</Name>
            <Value>Load</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_udm</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description</Name>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column_nlsid</Name>
            <Value>host_load_cpuIdle</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column</Name>
            <Value>cpuUtil</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit_nlsid</Name>
            <Value>em__sys__standard_percent</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit</Name>
            <Value>%</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group_nlsid</Name>
            <Value>host_load</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>severity_guid</Name>
            <Value>B6A5393A91CF32A0E040020A7801445B</Value>
        </StringAttribute>
    </EventClassSpecificAttributes>
</EMEvent>
```

B.3 Clear Sample

Listed below is a sample of the XML format generated by Enterprise Manager for SCOM clear operations.

```
<EMEvent xmlns="http://xmlns.oracle.com/sysman/connector">
    <ConnectorGUID>2C08969483F141F1B7482F376935C0AC</ConnectorGUID>
    <ExternalEventID>368d9bea-a659-4086-b239-0f7cf650d9df</ExternalEventID>
    <NotificationRuleOwner>SYSMAN</NotificationRuleOwner>
    <NotificationRuleName>SCOM R2 Rule</NotificationRuleName>
    <ConnectorVariable>
        <VariableName>REGISTRATION_ID</VariableName>
        <VariableValue>Dummy</VariableValue>
    </ConnectorVariable>
    <Property>
        <Name>Notification_Method_Name</Name>
        <Value>SCOM R2</Value>
    </Property>
    <Property>
        <Name>AuthenticationType</Name>
        <Value>HTTPBasicAuthentication</Value>
    </Property>
    <Property>
        <Name>Registered</Name>
        <Value>yes</Value>
    </Property>
    <Property>
        <Name>IsNewTargetType</Name>
        <Value>yes</Value>
    </Property>
    <Property>
        <Name>Initialized</Name>
        <Value>yes</Value>
    </Property>
    <SystemAttributes>
        <EventClass>metric_alert</EventClass>
        <EventID>B69EDA076C5C8256E040020A780165F6</EventID>
        <SequenceID>B69EC6D3F8B7A48EE040020A78016095</SequenceID>
        <ReportedDate>2012-01-17T17:04:45.000-06:00</ReportedDate>
        <DisplayTZ>America/Chicago</DisplayTZ>
        <EventName>Load:cpuUtil</EventName>
        <Severity>Clear</Severity>
        <SeverityCode>CLEAR</SeverityCode>
        <SourceInfo>
            <SourceObjInfo>
                <ObjID>D93A2E4E1622013D74D9534ECB090944</ObjID>
                <ObjName>orcsvr1.iwavesoftware.com</ObjName>
                <ObjOwner>SYSMAN</ObjOwner>
                <SourceObjType>TARGET</SourceObjType>
                <SourceObjSubType>host</SourceObjSubType>
            </SourceObjInfo>
            <TargetInfo>
                <TargetGUID>D93A2E4E1622013D74D9534ECB090944</TargetGUID>
                <TargetName>orcsvr1.iwavesoftware.com</TargetName>
                <TargetType>host</TargetType>
                <TargetTypeLabel>Host</TargetTypeLabel>

<TargetURL>https://orcsvr1.iwavesoftware.com:7801/em/redirect?pageType=TARGET_
HOMEPAGE&amp;targetName=orcsvr1.iwavesoftware.com&amp;targetType=host</TargetURL>
                <TargetProperty>
                    <Name>Target_Host</Name>
                    <Value>orcsvr1.iwavesoftware.com</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Target Version</Name>
                    <Value>5.5.0.0.0</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Line of Business</Name>
                    <Value>Software</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Location</Name>
                    <Value>Frisco, TX</Value>
                </TargetProperty>
                <TargetProperty>
                    <Name>Comment</Name>
                    <Value>Testing target properties</Value>
                </TargetProperty>
            </TargetInfo>
        </SourceInfo>
        <Message>CPU Utilization is 3.75%, fallen below warning (98) and critical
(99)
            thresholds.</Message>

<EventURL>https://orcsvr1.iwavesoftware.com:7801/em/redirect?pageType=sdk-core-eve
nt-console-detailEvent&amp;issueID=B69EC6D3F8B7A48EE040020A78016095</EventURL>
        <AutoClose>true</AutoClose>
        <EventCategory>Capacity</EventCategory>
    </SystemAttributes>
    <EventClassSpecificAttributes>
        <StringAttribute>
            <Name>is_thresholdable</Name>
            <Value>1</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>coll_name</Name>
            <Value>LoadLinux</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_metric_extension</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>cycle_guid</Name>
            <Value>B69EC6D3F8B3A48EE040020A78016095</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_remote</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_type</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_guid</Name>
            <Value>0C71A1AFAC2D7199013837DA35522C08</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>num_keys</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>key_value</Name>
            <Value> </Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description_nlsid</Name>
        </StringAttribute>
        <StringAttribute>
            <Name>value</Name>
            <Value>3.75</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_long_running</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group</Name>
            <Value>Load</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>is_udm</Name>
            <Value>0</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_description</Name>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column_nlsid</Name>
            <Value>host_load_cpuIdle</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_column</Name>
            <Value>cpuUtil</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit_nlsid</Name>
            <Value>em__sys__standard_percent</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>unit</Name>
            <Value>%</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group_nlsid</Name>
            <Value>host_load</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>metric_group_resbundle</Name>
            <Value>oracle.sysman.eml.rsc.gen.hostMsg</Value>
        </StringAttribute>
        <StringAttribute>
            <Name>severity_guid</Name>
            <Value>B69EDA076C5A8256E040020A780165F6</Value>
        </StringAttribute>
    </EventClassSpecificAttributes>
</EMEvent>
```

C

Enterprise Manager Event Format

Templates translate the XML format of Enterprise Manager events to the XML format required for the endpoint application's web service. To customize your templates, you need to understand the format of both ends so that you know how to map the data. This appendix provides detailed information on the format of the XML document that contains event information from Enterprise Manager.

A schema file defines the structure of an XML document. The EMEvent.xsd schema file defines the main structure of the event data. This file also imports structure information from other schema files. The following sections provide information from the schema files that define the EMEvent XML format.

C.1 EMEvent.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            xmlns="http://xmlns.oracle.com/sysman/connector"
            targetNamespace="http://xmlns.oracle.com/sysman/connector"
            elementFormDefault="qualified"
xmlns:jaxb="http://java.sun.com/xml/ns/jaxb"
  jaxb:version="2.0"
  xmlns:xjc="http://java.sun.com/xml/ns/jaxb/xjc"
  jaxb:extensionBindingPrefixes="xjc">

  <xsd:annotation>
    <xsd:appinfo>
      <jaxb:globalBindings>
        <xjc:simple />
      </jaxb:globalBindings>
    </xsd:appinfo>
  </xsd:annotation>
   <xsd:include schemaLocation="connectorCommon.xsd"/>
   <xsd:element name="EMEvent" type="EMEventType" />
   <xsd:complexType name="EMEventType">
         <xsd:sequence>
            <xsd:element name="ConnectorGUID" type="xsd:string"/>
            <xsd:element name="ExternalEventID" type="xsd:string"
                         minOccurs="0"/>
            <xsd:element name="NotificationRuleOwner" type="xsd:string"/>
            <xsd:element name="NotificationRuleName" type="xsd:string"/>
            <xsd:element name="ConnectorVariable" type="VariableType"
                         minOccurs="0" maxOccurs="50"/>
            <xsd:element name="Property" type="PropertyType" minOccurs="0"
                         maxOccurs="50"/>
            <xsd:element name="SystemAttributes"
```

```
                         type="EventSystemAttributesType"/>
            <xsd:element name="EventClassSpecificAttributes">
               <xsd:complexType>
                  <xsd:sequence>
                     <xsd:choice minOccurs="0" maxOccurs="200">
                     <xsd:element name="StringAttribute" type="StringValueType"/>
                     <xsd:element name="NumberAttribute" type="StringValueType"/>
                     <xsd:element name="RawAttribute" type="StringValueType"/>
                     <xsd:element name="DateAttribute" type="DateValueType"/>
                     </xsd:choice>
                  </xsd:sequence>
               </xsd:complexType>
            </xsd:element>
            <xsd:element name="EventContextAttributes">
               <xsd:complexType>
                  <xsd:sequence>
                     <xsd:choice minOccurs="0" maxOccurs="200">
                     <xsd:element name="StringAttribute" type="StringValueType"/>
                     <xsd:element name="NumberAttribute" type="StringValueType"/>
                     </xsd:choice>
                  </xsd:sequence>
               </xsd:complexType>
            </xsd:element>
         </xsd:sequence>
      </xsd:complexType>
   <xsd:complexType name="EventSystemAttributesType">
      <xsd:sequence>
         <xsd:element name="EventClass" type="xsd:string"/>
         <xsd:element name="EventID" type="xsd:string"/>
         <xsd:element name="SequenceID" type="xsd:string"/>
         <xsd:element name="OccurredDate" type="xsd:dateTime" minOccurs="0"/>
         <xsd:element name="ReportedDate" type="xsd:dateTime"/>
         <xsd:element name="DisplayTZ" type="xsd:string"/>
         <xsd:element name="EventName" type="xsd:string"/>
         <xsd:element name="Severity" type="xsd:string"/>
         <xsd:element name="SeverityCode" type="xsd:string"/>
         <xsd:element name="SourceInfo" type="SourceInfoType"/>
         <xsd:element name="Message" type="xsd:string" minOccurs="0"/>
         <xsd:element name="ActionMessage" type="xsd:string" minOccurs="0"/>
         <xsd:element name="EventURL" type="xsd:string"/>
         <xsd:element name="AutoClose" type="xsd:boolean"/>
         <xsd:element name="EventCategory" type="xsd:string" minOccurs="0"
                      maxOccurs="50"/>
      </xsd:sequence>
   </xsd:complexType>
   <xsd:complexType name="StringValueType">
      <xsd:sequence>
         <xsd:element name="Name" type="xsd:string"/>
         <xsd:element name="Value" type="xsd:string"/>
      </xsd:sequence>
   </xsd:complexType>
   <xsd:complexType name="DateValueType">
      <xsd:sequence>
         <xsd:element name="Name" type="xsd:string"/>
         <xsd:element name="Value" type="xsd:dateTime"/>
      </xsd:sequence>
   </xsd:complexType>
</xsd:schema>
```

C.2 connectorCommon.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            xmlns="http://xmlns.oracle.com/sysman/connector"
            targetNamespace="http://xmlns.oracle.com/sysman/connector"
            elementFormDefault="qualified">
  <xsd:include schemaLocation="externalEvent.xsd"/>
  <xsd:complexType name="SourceInfoType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for Source Information.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
   <xsd:element name="SourceObjInfo" type="SourceObjInfoType" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              This element defines the data structure for the source object, the
EM
              subsystem or component, that raises an EM event or an incident.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="TargetInfo" type="TargetInfoType" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              The element defines the data structure for an EM target as related
              to the connector framework.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="SourceObjInfoType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for Source Object Information.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ObjID" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              The unique ID to identify the source object.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="ObjName" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              The name of the source object.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
     <xsd:element name="ObjOwner" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              The owner of the source object.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="SourceObjType" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              The type of the source object.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="SourceObjSubType" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              The subtype of the source object.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="TargetInfoType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for target information.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="TargetGUID" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              A unique GUID for the target.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="TargetName" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              Name of the target.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="TargetType" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              Type of the target.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="TargetTypeLabel" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              The display label of the target type.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="TargetURL" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              The URL of the target.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="TargetProperty" type="PropertyType" minOccurs="0"
                   maxOccurs="50">
        <xsd:annotation>
            <xsd:documentation>
              An optional list of properties for the target.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="PropertyType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for a property attribute.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="Name" type="xsd:string">
          <xsd:annotation>
              <xsd:documentation>
                A string name defining a property attribute.
              </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
     <xsd:element name="Value" type="xsd:string" nillable="true">
          <xsd:annotation>
              <xsd:documentation>
                A non-null string value.
              </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="VariableType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for a general variable.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="VariableName" type="StringStrictT32">
          <xsd:annotation>
              <xsd:documentation>
                Name of the variable. It has to be a string containing 1 or upto
                32 upper case or lower case letters or numbers.
              </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      <xsd:element name="VariableValue" type="StringT2048">
          <xsd:annotation>
              <xsd:documentation>
                Value of the variable. It has to be a string containing 1 or upto
                2048 characters.
              </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="GetAlertsResponse">
```

```
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for responses to a getAlerts
request.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="Alert" minOccurs="0" maxOccurs="200">
          <xsd:annotation>
              <xsd:documentation>
                The individual alerts contained in the response. A response may
have
                upto 200 alerts.
              </xsd:documentation>
          </xsd:annotation>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element ref="ExternalEvent">
              <xsd:annotation>
                  <xsd:documentation>
                    Details of the external event in the alert, as defined in
                    ExternalEvent.xsd.
                  </xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="InstanceVariable" type="VariableType"
                         minOccurs="0" maxOccurs="50">
              <xsd:annotation>
                  <xsd:documentation>
                    A list of instance variables for the alert.
                  </xsd:documentation>
              </xsd:annotation>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="ConnectorVariablesType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for connector variables. An element
          of type ConnectorVariablesType may have up to 50 connector variables, as
          defined next.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ConnectorVariable" type="VariableType" minOccurs="0"
                   maxOccurs="50">
        <xsd:annotation>
            <xsd:documentation>
              A connector varialbe as a name/value pair.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:simpleType name="StringT64">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          64 bytes.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="64"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringT128">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          128 bytes.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="128"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringT256">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          256 bytes.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="256"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringT512">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          512 bytes.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="512"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringT2048">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          2048 bytes.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="2048"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringStrictT16">
    <xsd:annotation>
        <xsd:documentation>
```

```
          This section defines a simple type for a String with maximum length of
          16 bytes. The String can only contain lower or upper case letters,
numbers,
          and the underscore characters.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="16"/>
      <xsd:pattern value="([a-zA-Z0-9_])*"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringStrictT32">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          32 bytes. The String can only contain lower or upper case letters,
numbers,
          and the underscore characters.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="32"/>
      <xsd:pattern value="([a-zA-Z0-9_])*"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringStrictT64">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          64 bytes. The String can only contain lower or upper case letters,
numbers,
          and the underscore characters.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="64"/>
      <xsd:pattern value="([a-zA-Z0-9_])*"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringStrictT128">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          128 bytes. The String can only contain lower or upper case letters,
numbers,
          and the underscore characters.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="128"/>
      <xsd:pattern value="([a-zA-Z0-9_])*"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="StringStrictT256">
    <xsd:annotation>
        <xsd:documentation>
```

```
          This section defines a simple type for a String with maximum length of
          256 bytes. The String can only contain lower or upper case letters,
numbers,
          and the underscore characters.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="256"/>
      <xsd:pattern value="([a-zA-Z0-9_])*"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="VersionT">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a simple type for a String with maximum length of
          20 bytes. The String can only contain numbers and the period characters.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="1"/>
      <xsd:maxLength value="20"/>
      <xsd:pattern value="([0-9.])*"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

C.3 externalEvent.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            xmlns="http://xmlns.oracle.com/sysman/connector"
            targetNamespace="http://xmlns.oracle.com/sysman/connector"
            elementFormDefault="qualified">
  <xsd:element name="ExternalEvent">
    <xsd:annotation>
        <xsd:documentation>
          This section defines the attribute requirement of an extenral event
          for the connector framework to process it.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="SystemAttributes"
type="ExternalEventSystemAttributesType">
          <xsd:annotation>
              <xsd:documentation>
                Attributes to capture general information about the external event
                system. These attributes are system-specific, with all events from
                the same external system sharing the same system attributes.
              </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="EventClassAttributes"
                     type="ExternalEventClassAttributesType">
          <xsd:annotation>
              <xsd:documentation>
                Attributes to capture specific information required for the event
                as defined in the event class.
              </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:complexType name="ExternalEventSystemAttributesType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complex type for system attributes required for
          all external events.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
     <xsd:element name="eventName" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              Name of the event.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
  <xsd:element name="category" type="xsd:string" minOccurs="0" maxOccurs="50">
        <xsd:annotation>
            <xsd:documentation>
              The event category to which the event belongs.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="targetName" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              Name of the target on which event was generated. It refers
              to an entity in external systems simillar to an EM target.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="targetType" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              The type of the target. Target types defined for event connectors
              are used. See connectorDeploy.xsd.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="occurrenceDate" type="xsd:dateTime">
        <xsd:annotation>
            <xsd:documentation>
              Date when the event occured.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="detectedDate" type="xsd:dateTime">
        <xsd:annotation>
            <xsd:documentation>
              Date when the event was last detected.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="autoClose" type="xsd:boolean">
        <xsd:annotation>
            <xsd:documentation>
```

```
              A flag indicating if an event is auto closed by the system, or it
              has to be manually closed by users.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="message" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              A description of the event.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="severity" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              Severity level of the event.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="ExternalEventClassAttributesType">
    <xsd:annotation>
        <xsd:documentation>
          This section defines a complext type for class specific attributes
required
          for all external events in the class.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="external_event_id" type="xsd:string">
        <xsd:annotation>
            <xsd:documentation>
              ID used in external system to identify the event.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="external_rule_id" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              Optional rule ID that delivered the event in the external system.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="external_host" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              Optional host information from extrernal system where event was
generated.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="external_source" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              Optional source information from the external system.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="external_severity" type="xsd:string">
```

```
        <xsd:annotation>
            <xsd:documentation>
              Severity level of the event on external system.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="external_status" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              Status of the event on extenral system.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="custom_field1" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              An optional field.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="custom_field2" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              An optional field.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="custom_field3" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              An optional field.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="custom_field4" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              An optional field.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    <xsd:element name="custom_field5" type="xsd:string" minOccurs="0">
        <xsd:annotation>
            <xsd:documentation>
              An optional field.
            </xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

D

SCOM Web Service WSDL Format

Templates translate the XML format of Enterprise Manager events to the XML format required for the endpoint application's web service. To customize your templates, you need to understand the format of both ends so that you know how to map the data. This appendix provides detailed information on the format of the XML document that is required by the SCOM web service.

A WSDL file defines the interface that must be used to call a web service. The WSDL defines the structure of the XML document that it accepts and how the web service is to be called. The contents of the WSDL file are listed in the section entitled "EventService WSDL".

D.1 EventService WSDL

```
<wsdl:definitions name="EventService"
    targetNamespace="http://iwavesoftware.com/services/adapter-framework"
    xmlns:ns1="http://schemas.xmlsoap.org/soap/http"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:tns="http://iwavesoftware.com/services/adapter-framework"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <wsdl:types>
        <xs:schema attributeFormDefault="unqualified"
elementFormDefault="unqualified"
            targetNamespace="http://iwavesoftware.com/services/adapter-framework"
            xmlns:tns="http://iwavesoftware.com/services/adapter-framework"
            xmlns:xs="http://www.w3.org/2001/XMLSchema">
            <xs:element name="Class" type="tns:eventClass"/>
            <xs:element name="ITObject" type="tns:itObject"/>
            <xs:element name="Object" type="tns:eventObject"/>
            <xs:element name="acknowledge" type="tns:acknowledge"/>
            <xs:element name="acknowledgeResponse"
type="tns:acknowledgeResponse"/>
            <xs:element name="assignment" type="tns:assignment"/>
            <xs:element name="categorization" type="tns:categorization"/>
            <xs:element name="change" type="tns:change"/>
            <xs:element name="ci" type="tns:ci"/>
            <xs:element name="ciRef" type="tns:ciRef"/>
            <xs:element name="ciRelationship" type="tns:ciRelationship"/>
            <xs:element name="classification" type="tns:classification"/>
            <xs:element name="clear" type="tns:clear"/>
            <xs:element name="clearResponse" type="tns:clearResponse"/>
            <xs:element name="client-object" type="tns:clientDefinedObject"/>
            <xs:element name="complex-field" type="tns:complexField"/>
            <xs:element name="compliance" type="tns:compliance"/>
            <xs:element name="contact" type="tns:contact"/>
            <xs:element name="container" type="tns:container"/>
            <xs:element name="create" type="tns:create"/>
            <xs:element name="createResponse" type="tns:createResponse"/>
            <xs:element name="data" type="tns:data"/>
            <xs:element name="date-field" type="tns:dateField"/>
            <xs:element name="diary-list-field" type="tns:diaryListField"/>
            <xs:element name="diary-value" type="tns:diaryValue"/>
            <xs:element name="escalation" type="tns:escalation"/>
            <xs:element name="event" type="tns:event"/>
            <xs:element name="find" type="tns:find"/>
            <xs:element name="findResponse" type="tns:findResponse"/>
            <xs:element name="get" type="tns:get"/>
            <xs:element name="getResponse" type="tns:getResponse"/>
            <xs:element name="group" type="tns:group"/>
            <xs:element name="incident" type="tns:incident"/>
            <xs:element name="knowledge" type="tns:knowledge"/>
            <xs:element name="log" type="tns:log"/>
            <xs:element name="metric" type="tns:metric"/>
            <xs:element name="number-field" type="tns:numberField"/>
            <xs:element name="objectReference" type="tns:objectReference"/>
            <xs:element name="problem" type="tns:problem"/>
            <xs:element name="product" type="tns:product"/>
            <xs:element name="resolution" type="tns:resolution"/>
            <xs:element name="source" type="tns:source"/>
            <xs:element name="string-field" type="tns:stringField"/>
            <xs:element name="update" type="tns:update"/>
            <xs:element name="updateResponse" type="tns:updateResponse"/>
            <xs:element name="vendor" type="tns:vendor"/>
            <xs:complexType name="clear">
                <xs:sequence>
                    <xs:element minOccurs="0" name="destination"
type="xs:string"/>
                    <xs:element minOccurs="0" name="event" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="event">
                <xs:complexContent>
                    <xs:extension base="tns:itObject">
                        <xs:sequence>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="createDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="description"
type="xs:string"/>
                            <xs:element minOccurs="0" name="escalation"
type="tns:escalation"/>
                            <xs:element minOccurs="0" name="eventClass"
type="tns:eventClass"/>
                            <xs:element minOccurs="0"
name="externalEventIdentifier"
                                type="xs:string"/>
                            <xs:element minOccurs="0"
name="externalIncidentIdentifier"
                                type="xs:string"/>
                            <xs:element minOccurs="0" name="group"
type="tns:group"/>
                            <xs:element minOccurs="0" name="impact"
type="xs:string"/>
                            <xs:element minOccurs="0" name="knowledgeRoot">
```

```
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0"
                                            name="knowledge"
type="tns:knowledge"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="lastModified" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="lastModifiedBy"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="lastModifiedNonConnector"
type="xs:dateTime"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="lastRepeatDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="logs">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0" name="log"
                                            type="tns:log"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="maintenanceModeLastModified"
type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="metric"
type="tns:metric"/>
                            <xs:element name="monitorEvent" type="xs:boolean"/>
                            <xs:element minOccurs="0" name="object"
type="tns:eventObject"/>
                            <xs:element minOccurs="0" name="owner"
type="xs:string"/>
                            <xs:element minOccurs="0" name="priority"
type="xs:string"/>
                            <xs:element default="-2147483648" minOccurs="0"
name="repeatCount"
                                type="xs:int"/>
                            <xs:element minOccurs="0" name="resolvedBy"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="resolvedDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="severity"
type="xs:string"/>
                            <xs:element minOccurs="0" name="source"
type="tns:source"/>
                            <xs:element minOccurs="0" name="status"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="statusLastModified" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="summary"
type="xs:string"/>
                            <xs:element minOccurs="0" name="type"
type="xs:string"/>
                            <xs:element minOccurs="0" name="urgency"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="itObject">
                <xs:sequence>
                    <xs:element minOccurs="0" name="extended-fields">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:choice maxOccurs="unbounded" minOccurs="0">
                                    <xs:element name="string-field"
type="tns:stringField"/>
                                    <xs:element name="date-field"
type="tns:dateField"/>
                                    <xs:element name="number-field"
type="tns:numberField"/>
                                    <xs:element name="diary-list-field"
type="tns:diaryListField"/>
                                    <xs:element name="complex-field"
type="tns:complexField"/>
                                </xs:choice>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="escalation">
                <xs:sequence>
                    <xs:element name="escalated" type="xs:boolean"/>
                    <xs:element minOccurs="0" name="escalatedBy"
type="xs:string"/>
                    <xs:element minOccurs="0" name="server" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="eventClass">
                <xs:sequence>
                    <xs:element minOccurs="0" name="container"
type="tns:container"/>
                    <xs:element minOccurs="0" name="description"
type="xs:string"/>
                    <xs:element minOccurs="0" name="displayName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="container">
                <xs:sequence>
                    <xs:element minOccurs="0" name="description"
type="xs:string"/>
                    <xs:element minOccurs="0" name="displayName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="group">
                <xs:sequence>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="version" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="knowledge">
                <xs:sequence>
                    <xs:element minOccurs="0" name="content" type="xs:string"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="dateCreated" type="xs:dateTime"/>
                    <xs:element minOccurs="0" name="description"
type="xs:string"/>
                    <xs:element minOccurs="0" name="displayName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="languageCode"
type="xs:string"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="lastModified" type="xs:dateTime"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="status" type="xs:string"/>
                    <xs:element minOccurs="0" name="summary" type="xs:string"/>
                    <xs:element minOccurs="0" name="type" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="log">
                <xs:sequence>
                    <xs:element minOccurs="0" name="extended-fields">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:choice maxOccurs="unbounded" minOccurs="0">
                                    <xs:element name="string-field"
type="tns:stringField"/>
                                    <xs:element name="date-field"
type="tns:dateField"/>
                                    <xs:element name="number-field"
type="tns:numberField"/>
                                    <xs:element name="diary-list-field"
type="tns:diaryListField"/>
                                    <xs:element name="complex-field"
type="tns:complexField"/>
                                </xs:choice>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:element minOccurs="0" name="description"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="source" type="xs:string"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="submitDate" type="xs:dateTime"/>
                    <xs:element minOccurs="0" name="submitter" type="xs:string"/>
                    <xs:element minOccurs="0" name="summary" type="xs:string"/>
                    <xs:element minOccurs="0" name="type" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="stringField">
                <xs:simpleContent>
                    <xs:extension base="xs:string">
                        <xs:attribute name="name" type="xs:string"
use="required"/>
                    </xs:extension>
                </xs:simpleContent>
            </xs:complexType>
            <xs:complexType name="dateField">
                <xs:simpleContent>
                    <xs:extension base="xs:dateTime">
                        <xs:attribute name="name" type="xs:string"
use="required"/>
                    </xs:extension>
                </xs:simpleContent>
            </xs:complexType>
            <xs:complexType name="numberField">
                <xs:simpleContent>
                    <xs:extension base="xs:decimal">
                        <xs:attribute name="name" type="xs:string"
use="required"/>
                    </xs:extension>
                </xs:simpleContent>
            </xs:complexType>
            <xs:complexType name="diaryListField">
                <xs:sequence>
                    <xs:element maxOccurs="unbounded" minOccurs="0"
name="diary-value"
                        type="tns:diaryValue"/>
                </xs:sequence>
                <xs:attribute name="name" type="xs:string" use="required"/>
            </xs:complexType>
            <xs:complexType name="diaryValue">
                <xs:simpleContent>
                    <xs:extension base="xs:string">
                        <xs:attribute name="author" type="xs:string"/>
                        <xs:attribute name="timestamp" type="xs:dateTime"/>
                    </xs:extension>
                </xs:simpleContent>
            </xs:complexType>
            <xs:complexType name="complexField">
                <xs:sequence>
                    <xs:choice maxOccurs="unbounded" minOccurs="0">
                        <xs:element name="string-field" type="tns:stringField"/>
                        <xs:element name="date-field" type="tns:dateField"/>
                        <xs:element name="number-field" type="tns:numberField"/>
                        <xs:element name="diary-list-field"
type="tns:diaryListField"/>
                        <xs:element name="complex-field" type="tns:complexField"/>
                    </xs:choice>
                </xs:sequence>
                <xs:attribute name="name" type="xs:string" use="required"/>
            </xs:complexType>
            <xs:complexType name="metric">
                <xs:sequence>
                    <xs:element minOccurs="0" name="category" type="xs:string"/>
                    <xs:element minOccurs="0" name="keyValues">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element maxOccurs="unbounded" minOccurs="0"
name="string"
                                    type="xs:string"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:element minOccurs="0" name="subCategory"
type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="eventObject">
                <xs:sequence>
                    <xs:element minOccurs="0" name="computerName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="displayName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="inMaintenanceMode"
type="xs:boolean"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="state" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="source">
                <xs:sequence>
                    <xs:element minOccurs="0" name="computerName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="container"
type="tns:container"/>
                    <xs:element minOccurs="0" name="description"
type="xs:string"/>
                    <xs:element minOccurs="0" name="displayName"
type="xs:string"/>
                    <xs:element minOccurs="0" name="domain" type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="type" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="clearResponse">
                <xs:sequence/>
            </xs:complexType>
            <xs:complexType name="update">
                <xs:sequence>
                    <xs:element minOccurs="0" name="destination"
type="xs:string"/>
                    <xs:element minOccurs="0" name="event" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="updateResponse">
                <xs:sequence>
                    <xs:element minOccurs="0" name="return" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="acknowledge">
                <xs:sequence>
                    <xs:element minOccurs="0" name="destination"
```

```
type="xs:string"/>
                    <xs:element minOccurs="0" name="event" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="acknowledgeResponse">
                <xs:sequence>
                    <xs:element minOccurs="0" name="return" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="get">
                <xs:sequence>
                    <xs:element minOccurs="0" name="destination"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="getResponse">
                <xs:sequence>
                    <xs:element minOccurs="0" name="return" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="find">
                <xs:sequence>
                    <xs:element minOccurs="0" name="destination"
type="xs:string"/>
                    <xs:element minOccurs="0" name="query" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="findResponse">
                <xs:sequence>
                    <xs:element maxOccurs="unbounded" minOccurs="0" name="return"
type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="create">
                <xs:sequence>
                    <xs:element minOccurs="0" name="destination"
type="xs:string"/>
                    <xs:element minOccurs="0" name="event" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="createResponse">
                <xs:sequence>
                    <xs:element minOccurs="0" name="return" type="tns:event"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType mixed="true" name="clientDefinedObject">
                <xs:sequence>
                    <xs:any maxOccurs="unbounded" minOccurs="0"
namespace="##other"
                        processContents="skip"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="incident">
                <xs:complexContent>
                    <xs:extension base="tns:itObject">
                        <xs:sequence>
                            <xs:element minOccurs="0" name="assignee"
type="tns:assignment"/>
                            <xs:element minOccurs="0" name="cis">
```

```
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0" name="ci"
                                            type="tns:ciRef"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="classification"
                                type="tns:classification"/>
                            <xs:element minOccurs="0" name="customer"
type="tns:contact"/>
                            <xs:element minOccurs="0" name="description"
type="xs:string"/>
                            <xs:element minOccurs="0" name="direct"
type="tns:contact"/>
                            <xs:element minOccurs="0" name="impact"
type="xs:string"/>
                            <xs:element minOccurs="0" name="log-data">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0" name="log"
                                            type="tns:log"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="logs">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0" name="log"
                                            type="tns:objectReference"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="owner"
type="tns:assignment"/>
                            <xs:element minOccurs="0" name="priority"
type="xs:string"/>
                            <xs:element minOccurs="0" name="relationships">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0"
                                            name="relationship"
type="tns:objectReference"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="resolution"
type="tns:resolution"/>
                            <xs:element minOccurs="0" name="severity"
type="xs:string"/>
                            <xs:element minOccurs="0" name="status"
type="xs:string"/>
                            <xs:element minOccurs="0" name="summary"
type="xs:string"/>
                            <xs:element minOccurs="0" name="urgency"
type="xs:string"/>
```

```
                            <xs:element minOccurs="0" name="vendor"
type="tns:vendor"/>
                            <xs:element minOccurs="0" name="weight"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="assignment">
                <xs:sequence>
                    <xs:element minOccurs="0" name="company" type="xs:string"/>
                    <xs:element minOccurs="0" name="organization"
type="xs:string"/>
                    <xs:element minOccurs="0" name="owner" type="xs:string"/>
                    <xs:element minOccurs="0" name="primaryGroup"
type="xs:string"/>
                    <xs:element minOccurs="0" name="secondaryGroup"
type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="ciRef">
                <xs:sequence>
                    <xs:element minOccurs="0" name="extended-fields">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:choice maxOccurs="unbounded" minOccurs="0">
                                    <xs:element name="string-field"
type="tns:stringField"/>
                                    <xs:element name="date-field"
type="tns:dateField"/>
                                    <xs:element name="number-field"
type="tns:numberField"/>
                                    <xs:element name="diary-list-field"
type="tns:diaryListField"/>
                                    <xs:element name="complex-field"
type="tns:complexField"/>
                                </xs:choice>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:element minOccurs="0" name="description"
type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="status" type="xs:string"/>
                    <xs:element minOccurs="0" name="type" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="classification">
                <xs:sequence>
                    <xs:element minOccurs="0" name="company" type="xs:string"/>
                    <xs:element minOccurs="0" name="operational"
type="tns:categorization"/>
                    <xs:element minOccurs="0" name="product" type="tns:product"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="reportedDate" type="xs:dateTime"/>
                    <xs:element minOccurs="0" name="reportedSource"
type="xs:string"/>
                    <xs:element minOccurs="0" name="serviceType"
```

```
type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="categorization">
                <xs:sequence>
                    <xs:element minOccurs="0" name="tier1" type="xs:string"/>
                    <xs:element minOccurs="0" name="tier2" type="xs:string"/>
                    <xs:element minOccurs="0" name="tier3" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="product">
                <xs:complexContent>
                    <xs:extension base="tns:categorization">
                        <xs:sequence>
                            <xs:element minOccurs="0" name="manufacturer"
type="xs:string"/>
                            <xs:element minOccurs="0" name="modelVersion"
type="xs:string"/>
                            <xs:element minOccurs="0" name="name"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="contact">
                <xs:sequence>
                    <xs:element minOccurs="0" name="company" type="xs:string"/>
                    <xs:element minOccurs="0" name="department" type="xs:string"/>
                    <xs:element minOccurs="0" name="emailAddress"
type="xs:string"/>
                    <xs:element minOccurs="0" name="firstName" type="xs:string"/>
                    <xs:element minOccurs="0" name="lastName" type="xs:string"/>
                    <xs:element minOccurs="0" name="middleName" type="xs:string"/>
                    <xs:element minOccurs="0" name="organization"
type="xs:string"/>
                    <xs:element minOccurs="0" name="phoneNumber"
type="xs:string"/>
                    <xs:element minOccurs="0" name="site" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="objectReference">
                <xs:sequence>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="role" type="xs:string"/>
                    <xs:element minOccurs="0" name="type" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="resolution">
                <xs:complexContent>
                    <xs:extension base="tns:categorization">
                        <xs:sequence>
                            <xs:element minOccurs="0" name="cause"
type="xs:string"/>
                            <xs:element minOccurs="0" name="closureSource"
type="xs:string"/>
                            <xs:element minOccurs="0" name="detail"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
```

```
                                name="forecastResolutionDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="method"
type="xs:string"/>
                            <xs:element minOccurs="0" name="product"
type="tns:product"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="requiredDate" type="xs:dateTime"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="resolutionDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="satisfactionRating"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="vendor">
                <xs:sequence>
                    <xs:element minOccurs="0" name="assignmentStatus"
type="xs:string"/>
                    <xs:element minOccurs="0" name="contact" type="tns:contact"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="reportedToVendorDate" type="xs:dateTime"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="resolvedDate" type="xs:dateTime"/>
                    <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                        name="respondedToVendorDate" type="xs:dateTime"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="ci">
                <xs:complexContent>
                    <xs:extension base="tns:itObject">
                        <xs:sequence>
                            <xs:element minOccurs="0" name="assignee"
type="xs:string"/>
                            <xs:element minOccurs="0" name="category"
type="xs:string"/>
                            <xs:element minOccurs="0" name="className"
type="xs:string"/>
                            <xs:element minOccurs="0" name="classNamespace"
type="xs:string"/>
                            <xs:element minOccurs="0" name="compliance"
type="tns:compliance"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="createDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="datasetId"
type="xs:string"/>
                            <xs:element minOccurs="0" name="description"
type="xs:string"/>
                            <xs:element minOccurs="0" name="hostName"
type="xs:string"/>
                            <xs:element minOccurs="0" name="item"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
```

```
minOccurs="0"
                                name="lastModified" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="lastModifiedBy"
type="xs:string"/>
                            <xs:element minOccurs="0" name="manufacturer"
type="xs:string"/>
                            <xs:element minOccurs="0" name="model"
type="xs:string"/>
                            <xs:element minOccurs="0" name="name"
type="xs:string"/>
                            <xs:element minOccurs="0" name="relationships">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0"
                                            name="relationship"
type="tns:ciRelationship"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="serialNumber"
type="xs:string"/>
                            <xs:element minOccurs="0" name="status"
type="xs:string"/>
                            <xs:element minOccurs="0" name="submitter"
type="xs:string"/>
                            <xs:element minOccurs="0" name="summary"
type="xs:string"/>
                            <xs:element minOccurs="0" name="type"
type="xs:string"/>
                            <xs:element minOccurs="0" name="version"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="compliance">
                <xs:sequence>
                    <xs:element minOccurs="0" name="policyIdentifier"
type="xs:string"/>
                    <xs:element minOccurs="0" name="policyName" type="xs:string"/>
                    <xs:element minOccurs="0" name="status" type="xs:string"/>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="ciRelationship">
                <xs:sequence>
                    <xs:element minOccurs="0" name="className" type="xs:string"/>
                    <xs:element minOccurs="0" name="classNamespace"
type="xs:string"/>
                    <xs:element minOccurs="0" name="datasetId" type="xs:string"/>
                    <xs:element minOccurs="0" name="identifier" type="xs:string"/>
                    <xs:element minOccurs="0" name="name" type="xs:string"/>
                    <xs:element minOccurs="0" name="sourceRole" type="xs:string"/>
                    <xs:element minOccurs="0" name="targets">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element maxOccurs="unbounded" minOccurs="0"
name="target"
                                    type="tns:objectReference"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
            <xs:complexType name="change">
                <xs:complexContent>
                    <xs:extension base="tns:itObject">
                        <xs:sequence>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="actualEnd" type="xs:dateTime"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="actualStart" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="businessJustification"
type="xs:string"/>
                            <xs:element minOccurs="0" name="categorization"
                                type="tns:categorization"/>
                            <xs:element minOccurs="0" name="company"
type="xs:string"/>
                            <xs:element minOccurs="0" name="createDate"
type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="description"
type="xs:string"/>
                            <xs:element minOccurs="0" name="impact"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="lastModified" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="lastModifiedBy"
type="xs:string"/>
                            <xs:element minOccurs="0" name="logs">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0" name="log"
                                            type="tns:objectReference"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="priority"
type="xs:string"/>
                            <xs:element minOccurs="0" name="product"
type="tns:product"/>
                            <xs:element minOccurs="0" name="reason"
type="xs:string"/>
                            <xs:element minOccurs="0" name="relationships">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0"
                                            name="relationship"
type="tns:objectReference"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="requester"
type="tns:contact"/>
                            <xs:element minOccurs="0" name="riskLevel"
type="xs:string"/>
```

```
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="scheduledEnd" type="xs:dateTime"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="scheduledStart" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="status"
type="xs:string"/>
                            <xs:element minOccurs="0" name="summary"
type="xs:string"/>
                            <xs:element minOccurs="0" name="timing"
type="xs:string"/>
                            <xs:element minOccurs="0" name="type"
type="xs:string"/>
                            <xs:element minOccurs="0" name="urgency"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="problem">
                <xs:complexContent>
                    <xs:extension base="tns:itObject">
                        <xs:sequence>
                            <xs:element minOccurs="0" name="categorization"
                                type="tns:categorization"/>
                            <xs:element minOccurs="0" name="company"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="createDate" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="description"
type="xs:string"/>
                            <xs:element minOccurs="0"
name="externalProblemIdentifier"
                                type="xs:string"/>
                            <xs:element minOccurs="0" name="impact"
type="xs:string"/>
                            <xs:element minOccurs="0" name="investigationDriver"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="lastModified" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="lastModifiedBy"
type="xs:string"/>
                            <xs:element minOccurs="0" name="logs">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0" name="log"
                                            type="tns:objectReference"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="priority"
type="xs:string"/>
                            <xs:element minOccurs="0" name="product"
type="tns:product"/>
                            <xs:element minOccurs="0" name="relationships">
                                <xs:complexType>
```

```
                                    <xs:sequence>
                                        <xs:element maxOccurs="unbounded"
minOccurs="0"
                                            name="relationship"
type="tns:objectReference"/>
                                    </xs:sequence>
                                </xs:complexType>
                            </xs:element>
                            <xs:element minOccurs="0" name="requester"
type="tns:contact"/>
                            <xs:element minOccurs="0" name="rootCause"
type="xs:string"/>
                            <xs:element minOccurs="0" name="severity"
type="xs:string"/>
                            <xs:element minOccurs="0" name="status"
type="xs:string"/>
                            <xs:element minOccurs="0" name="summary"
type="xs:string"/>
                            <xs:element default="1970-01-01T00:00:00.0-00:00"
minOccurs="0"
                                name="targetResolution" type="xs:dateTime"/>
                            <xs:element minOccurs="0" name="urgency"
type="xs:string"/>
                            <xs:element minOccurs="0" name="weight"
type="xs:string"/>
                            <xs:element minOccurs="0" name="workaround"
type="xs:string"/>
                        </xs:sequence>
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            <xs:complexType name="data">
                <xs:sequence>
                    <xs:choice maxOccurs="unbounded" minOccurs="0">
                        <xs:element name="string-field" type="tns:stringField"/>
                        <xs:element name="date-field" type="tns:dateField"/>
                        <xs:element name="number-field" type="tns:numberField"/>
                        <xs:element name="diary-list-field"
type="tns:diaryListField"/>
                        <xs:element name="complex-field" type="tns:complexField"/>
                    </xs:choice>
                </xs:sequence>
            </xs:complexType>
            <xs:element name="UnsupportedOperationException"
                type="tns:UnsupportedOperationException"/>
            <xs:complexType name="UnsupportedOperationException">
                <xs:sequence/>
            </xs:complexType>
            <xs:element name="OperationsException"
type="tns:OperationsException"/>
            <xs:complexType name="OperationsException">
                <xs:sequence/>
            </xs:complexType>
            <xs:element name="ObjectNotFoundException"
type="tns:ObjectNotFoundException"/>
            <xs:complexType name="ObjectNotFoundException">
                <xs:sequence/>
            </xs:complexType>
            <xs:element name="NonRecoverableOperationsException"
                type="tns:NonRecoverableOperationsException"/>
            <xs:complexType name="NonRecoverableOperationsException">
                <xs:sequence/>
            </xs:complexType>
        </xs:schema>
    </wsdl:types>
    <wsdl:message name="getResponse">
        <wsdl:part element="tns:getResponse" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="create">
        <wsdl:part element="tns:create" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="clearResponse">
        <wsdl:part element="tns:clearResponse" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="NonRecoverableOperationsException">
        <wsdl:part element="tns:NonRecoverableOperationsException"
            name="NonRecoverableOperationsException"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="acknowledgeResponse">
        <wsdl:part element="tns:acknowledgeResponse" name="parameters">
</wsdl:part>
    </wsdl:message>
    <wsdl:message name="update">
        <wsdl:part element="tns:update" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="ObjectNotFoundException">
        <wsdl:part element="tns:ObjectNotFoundException"
name="ObjectNotFoundException"
        > </wsdl:part>
    </wsdl:message>
    <wsdl:message name="acknowledge">
        <wsdl:part element="tns:acknowledge" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="get">
        <wsdl:part element="tns:get" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="createResponse">
        <wsdl:part element="tns:createResponse" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="findResponse">
        <wsdl:part element="tns:findResponse" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="OperationsException">
        <wsdl:part element="tns:OperationsException" name="OperationsException">
</wsdl:part>
    </wsdl:message>
    <wsdl:message name="find">
        <wsdl:part element="tns:find" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="clear">
        <wsdl:part element="tns:clear" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:message name="UnsupportedOperationException">
        <wsdl:part element="tns:UnsupportedOperationException"
name="UnsupportedOperationException"
        > </wsdl:part>
    </wsdl:message>
    <wsdl:message name="updateResponse">
        <wsdl:part element="tns:updateResponse" name="parameters"> </wsdl:part>
    </wsdl:message>
    <wsdl:portType name="IEventServiceDestination">
        <wsdl:operation name="clear">
            <wsdl:input message="tns:clear" name="clear"> </wsdl:input>
            <wsdl:output message="tns:clearResponse" name="clearResponse">
</wsdl:output>
            <wsdl:fault message="tns:ObjectNotFoundException"
name="ObjectNotFoundException"> </wsdl:fault>
            <wsdl:fault message="tns:UnsupportedOperationException"
                name="UnsupportedOperationException"> </wsdl:fault>
            <wsdl:fault message="tns:OperationsException"
name="OperationsException"> </wsdl:fault>
            <wsdl:fault message="tns:NonRecoverableOperationsException"
                name="NonRecoverableOperationsException"> </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="update">
            <wsdl:input message="tns:update" name="update"> </wsdl:input>
            <wsdl:output message="tns:updateResponse" name="updateResponse">
</wsdl:output>
            <wsdl:fault message="tns:ObjectNotFoundException"
name="ObjectNotFoundException"> </wsdl:fault>
            <wsdl:fault message="tns:OperationsException"
name="OperationsException"> </wsdl:fault>
            <wsdl:fault message="tns:NonRecoverableOperationsException"
                name="NonRecoverableOperationsException"> </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="acknowledge">
            <wsdl:input message="tns:acknowledge" name="acknowledge">
</wsdl:input>
            <wsdl:output message="tns:acknowledgeResponse"
name="acknowledgeResponse"> </wsdl:output>
            <wsdl:fault message="tns:ObjectNotFoundException"
name="ObjectNotFoundException"> </wsdl:fault>
            <wsdl:fault message="tns:UnsupportedOperationException"
                name="UnsupportedOperationException"> </wsdl:fault>
            <wsdl:fault message="tns:OperationsException"
name="OperationsException"> </wsdl:fault>
            <wsdl:fault message="tns:NonRecoverableOperationsException"
                name="NonRecoverableOperationsException"> </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="get">
            <wsdl:input message="tns:get" name="get"> </wsdl:input>
            <wsdl:output message="tns:getResponse" name="getResponse">
</wsdl:output>
            <wsdl:fault message="tns:ObjectNotFoundException"
name="ObjectNotFoundException"> </wsdl:fault>
            <wsdl:fault message="tns:UnsupportedOperationException"
                name="UnsupportedOperationException"> </wsdl:fault>
            <wsdl:fault message="tns:OperationsException"
name="OperationsException"> </wsdl:fault>
            <wsdl:fault message="tns:NonRecoverableOperationsException"
                name="NonRecoverableOperationsException"> </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="find">
            <wsdl:input message="tns:find" name="find"> </wsdl:input>
            <wsdl:output message="tns:findResponse" name="findResponse">
</wsdl:output>
            <wsdl:fault message="tns:UnsupportedOperationException"
                name="UnsupportedOperationException"> </wsdl:fault>
            <wsdl:fault message="tns:OperationsException"
name="OperationsException"> </wsdl:fault>
```

```
            <wsdl:fault message="tns:NonRecoverableOperationsException"
                name="NonRecoverableOperationsException"> </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="create">
            <wsdl:input message="tns:create" name="create"> </wsdl:input>
            <wsdl:output message="tns:createResponse" name="createResponse">
</wsdl:output>
            <wsdl:fault message="tns:OperationsException"
name="OperationsException"> </wsdl:fault>
            <wsdl:fault message="tns:NonRecoverableOperationsException"
                name="NonRecoverableOperationsException"> </wsdl:fault>
        </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="EventServiceSoapBinding"
type="tns:IEventServiceDestination">
        <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
        <wsdl:operation name="clear">
            <soap:operation soapAction="" style="document"/>
            <wsdl:input name="clear">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="clearResponse">
                <soap:body use="literal"/>
            </wsdl:output>
            <wsdl:fault name="ObjectNotFoundException">
                <soap:fault name="ObjectNotFoundException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="UnsupportedOperationException">
                <soap:fault name="UnsupportedOperationException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="OperationsException">
                <soap:fault name="OperationsException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="NonRecoverableOperationsException">
                <soap:fault name="NonRecoverableOperationsException"
use="literal"/>
            </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="update">
            <soap:operation soapAction="" style="document"/>
            <wsdl:input name="update">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="updateResponse">
                <soap:body use="literal"/>
            </wsdl:output>
            <wsdl:fault name="ObjectNotFoundException">
                <soap:fault name="ObjectNotFoundException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="OperationsException">
                <soap:fault name="OperationsException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="NonRecoverableOperationsException">
                <soap:fault name="NonRecoverableOperationsException"
use="literal"/>
            </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="acknowledge">
            <soap:operation soapAction="" style="document"/>
            <wsdl:input name="acknowledge">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="acknowledgeResponse">
                <soap:body use="literal"/>
            </wsdl:output>
            <wsdl:fault name="ObjectNotFoundException">
                <soap:fault name="ObjectNotFoundException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="UnsupportedOperationException">
                <soap:fault name="UnsupportedOperationException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="OperationsException">
                <soap:fault name="OperationsException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="NonRecoverableOperationsException">
                <soap:fault name="NonRecoverableOperationsException"
use="literal"/>
            </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="get">
            <soap:operation soapAction="" style="document"/>
            <wsdl:input name="get">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="getResponse">
                <soap:body use="literal"/>
            </wsdl:output>
            <wsdl:fault name="ObjectNotFoundException">
                <soap:fault name="ObjectNotFoundException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="UnsupportedOperationException">
                <soap:fault name="UnsupportedOperationException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="OperationsException">
                <soap:fault name="OperationsException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="NonRecoverableOperationsException">
                <soap:fault name="NonRecoverableOperationsException"
use="literal"/>
            </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="find">
            <soap:operation soapAction="" style="document"/>
            <wsdl:input name="find">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="findResponse">
                <soap:body use="literal"/>
            </wsdl:output>
            <wsdl:fault name="UnsupportedOperationException">
                <soap:fault name="UnsupportedOperationException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="OperationsException">
                <soap:fault name="OperationsException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="NonRecoverableOperationsException">
                <soap:fault name="NonRecoverableOperationsException"
use="literal"/>
            </wsdl:fault>
        </wsdl:operation>
        <wsdl:operation name="create">
            <soap:operation soapAction="" style="document"/>
            <wsdl:input name="create">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="createResponse">
                <soap:body use="literal"/>
            </wsdl:output>
            <wsdl:fault name="OperationsException">
                <soap:fault name="OperationsException" use="literal"/>
            </wsdl:fault>
            <wsdl:fault name="NonRecoverableOperationsException">
                <soap:fault name="NonRecoverableOperationsException"
use="literal"/>
            </wsdl:fault>
        </wsdl:operation>
    </wsdl:binding>
    <wsdl:service name="EventService">
        <wsdl:port binding="tns:EventServiceSoapBinding" name="EventServicePort">
            <soap:address
location="http://localhost:8080/services/SCOM2007/EventService"/>
        </wsdl:port>
    </wsdl:service>
</wsdl:definitions>
```

D.2 SCOMService WSDL

```
<?xml version='1.0' encoding='UTF-8'?>
<wsdl:definitions name="SCOMService"
  targetNamespace="http://iwavesoftware.com/services/adapter-framework"
  xmlns:ns1="http://schemas.xmlsoap.org/soap/http"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://iwavesoftware.com/services/adapter-framework"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <xs:schema attributeFormDefault="unqualified" elementFormDefault="unqualified"
      targetNamespace="http://iwavesoftware.com/services/adapter-framework"
      xmlns:tns="http://iwavesoftware.com/services/adapter-framework"
      xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="deregister" type="tns:deregister"/>
      <xs:element name="deregisterResponse" type="tns:deregisterResponse"/>
      <xs:element name="register" type="tns:register"/>
      <xs:element name="registerResponse" type="tns:registerResponse"/>
      <xs:complexType name="deregister">
        <xs:sequence>
          <xs:element minOccurs="0" name="destination" type="xs:string"/>
          <xs:element minOccurs="0" name="deleteSubscriptions" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="deregisterResponse">
        <xs:sequence/>
      </xs:complexType>
      <xs:complexType name="register">
        <xs:sequence>
          <xs:element minOccurs="0" name="destination" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="registerResponse">
```

```
        <xs:sequence/>
      </xs:complexType>
      <xs:element name="OperationsException" type="tns:OperationsException"/>
      <xs:complexType name="OperationsException">
        <xs:sequence/>
      </xs:complexType>
      <xs:element name="NonRecoverableOperationsException"
        type="tns:NonRecoverableOperationsException"/>
      <xs:complexType name="NonRecoverableOperationsException">
        <xs:sequence/>
      </xs:complexType>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="OperationsException">
    <wsdl:part element="tns:OperationsException" name="OperationsException">
</wsdl:part>
  </wsdl:message>
  <wsdl:message name="registerResponse">
    <wsdl:part element="tns:registerResponse" name="parameters"> </wsdl:part>
  </wsdl:message>
  <wsdl:message name="NonRecoverableOperationsException">
    <wsdl:part element="tns:NonRecoverableOperationsException"
      name="NonRecoverableOperationsException"> </wsdl:part>
  </wsdl:message>
  <wsdl:message name="deregister">
    <wsdl:part element="tns:deregister" name="parameters"> </wsdl:part>
  </wsdl:message>
  <wsdl:message name="register">
    <wsdl:part element="tns:register" name="parameters"> </wsdl:part>
  </wsdl:message>
  <wsdl:message name="deregisterResponse">
    <wsdl:part element="tns:deregisterResponse" name="parameters"> </wsdl:part>
  </wsdl:message>
  <wsdl:portType name="ISCOMService">
    <wsdl:operation name="deregister">
      <wsdl:input message="tns:deregister" name="deregister"> </wsdl:input>
      <wsdl:output message="tns:deregisterResponse" name="deregisterResponse">
</wsdl:output>
      <wsdl:fault message="tns:OperationsException" name="OperationsException">
</wsdl:fault>
      <wsdl:fault message="tns:NonRecoverableOperationsException"
        name="NonRecoverableOperationsException"> </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="register">
      <wsdl:input message="tns:register" name="register"> </wsdl:input>
      <wsdl:output message="tns:registerResponse" name="registerResponse">
</wsdl:output>
      <wsdl:fault message="tns:OperationsException" name="OperationsException">
</wsdl:fault>
      <wsdl:fault message="tns:NonRecoverableOperationsException"
        name="NonRecoverableOperationsException"> </wsdl:fault>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="SCOMServiceSoapBinding" type="tns:ISCOMService">
    <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="deregister">
      <soap:operation soapAction="" style="document"/>
      <wsdl:input name="deregister">
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output name="deregisterResponse">
        <soap:body use="literal"/>
      </wsdl:output>
      <wsdl:fault name="OperationsException">
        <soap:fault name="OperationsException" use="literal"/>
      </wsdl:fault>
      <wsdl:fault name="NonRecoverableOperationsException">
        <soap:fault name="NonRecoverableOperationsException" use="literal"/>
      </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="register">
      <soap:operation soapAction="" style="document"/>
      <wsdl:input name="register">
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output name="registerResponse">
        <soap:body use="literal"/>
      </wsdl:output>
      <wsdl:fault name="OperationsException">
        <soap:fault name="OperationsException" use="literal"/>
      </wsdl:fault>
      <wsdl:fault name="NonRecoverableOperationsException">
        <soap:fault name="NonRecoverableOperationsException" use="literal"/>
      </wsdl:fault>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="SCOMService">
    <wsdl:port binding="tns:SCOMServiceSoapBinding" name="SCOMServicePort">
      <soap:address
location="http://localhost:8080/services/SCOM2007/SCOMService"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Index