# Table of Contents

<table>
<thead>
<tr>
<th>Preface</th>
<th>vii</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audience</td>
<td>vii</td>
</tr>
<tr>
<td>Organization</td>
<td>vii</td>
</tr>
<tr>
<td>Font conventions</td>
<td>vii</td>
</tr>
<tr>
<td>Related documents</td>
<td>viii</td>
</tr>
</tbody>
</table>

## Introduction

<table>
<thead>
<tr>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>About the Reverse Proxy Adapter</td>
</tr>
<tr>
<td>Example Process</td>
</tr>
</tbody>
</table>

## System Configuration

<table>
<thead>
<tr>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supported Operating Systems</td>
</tr>
<tr>
<td>System Requirements</td>
</tr>
<tr>
<td>Recommended</td>
</tr>
<tr>
<td>Minimum</td>
</tr>
<tr>
<td>Supported Databases</td>
</tr>
</tbody>
</table>

## Software Installation

<table>
<thead>
<tr>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
</tr>
<tr>
<td>Configuring SSL Communications</td>
</tr>
<tr>
<td>Installing a License File</td>
</tr>
<tr>
<td>Installing the Adapter Software - Windows</td>
</tr>
<tr>
<td>Before You Begin</td>
</tr>
<tr>
<td>Start the Installer</td>
</tr>
<tr>
<td>Installing the Adapter</td>
</tr>
<tr>
<td>Running the Adapter Windows Service</td>
</tr>
<tr>
<td>Shortcuts</td>
</tr>
<tr>
<td>Executable Location</td>
</tr>
<tr>
<td>Service Manager Application</td>
</tr>
<tr>
<td>Installing Adapter Software - UNIX</td>
</tr>
<tr>
<td>Before You Begin</td>
</tr>
<tr>
<td>Installing the Software</td>
</tr>
<tr>
<td>Running the Adapter UNIX Service</td>
</tr>
<tr>
<td>Manual Installation Procedures</td>
</tr>
<tr>
<td>Before You Begin</td>
</tr>
<tr>
<td>Install the Files</td>
</tr>
</tbody>
</table>
Framework Properties

Properties Editor Tool .................................................. 44
Starting the Tool .......................................................... 44
Using the Command ....................................................... 45
Parameters ....................................................................... 45
Global Framework Properties ........................................ 45
Acquisition monitoring ................................................... 47
Automated Alerts ............................................................. 47
Reverse Proxy Framework Properties ............................... 49

Custom Configuration

Client Acquisition Service .............................................. 51
Data Format ................................................................. 51
Object Format .............................................................. 51
RESTful Client Acquisition ........................................... 52
Calling the RESTful Client Acquisition from curl .............. 52
SOAP Client Acquisition Web Service .............................. 53
Running on another port ................................................ 53
Adapter High Availability ................................................ 55
Operations Services Scaling ......................................... 55
Acquisition Services Scaling ......................................... 55
Use the External Database for Multiple Adapters .............. 56
Adapter Failover Infrastructure ....................................... 56
Agent Scaling ............................................................... 56
Java to Adapter Integration ............................................ 57
Prerequisites .................................................................... 57
Persistence Database Management .................................. 59
Removing Transactions ................................................ 59
Reclaiming Unused Disk Space ....................................... 59
Connecting to the Database ............................................ 60
Changing the Database Port Number ............................... 60
Offloading the Acquisition Queue to an External Database . 60

Managing the Adapter

Prerequisites ..................................................................... 62
Connecting to the Adapter Java Process ............................ 62
Starting the Java Console ............................................... 63
Management Beans ....................................................... 63
com.iwave:name=Adapters ............................................ 63
com.iwave:name=AcquisitionQueue ................................ 64
Methods ........................................................................ 64
com.iwave:name=FrameworkAdmin ................................. 65

Maintenance and Troubleshooting

Logging ........................................................................... 67
Log Location .................................................................... 67
Preface

This manual includes instructions on installing, configuring, and troubleshooting the EMC Reverse Proxy Adapter software.

Audience

This publication is written for the System Administrator or Engineer responsible for installing and configuring the EMC Reverse Proxy Adapter software.

Organization

This publication consists of the following chapters:

- Introduction
- System Configuration
- Software Installation
- Framework Properties
- Custom Configuration
- Orchestrator Configuration
- Managing the Adapter
- Maintenance and Troubleshooting
- Index

Font conventions

- Bold format is used to identify dialog boxes and menu choices. For example: Start > Programs > IonixITO> adapters-2.4
- Courier fixed-width font is used for code elements (Java, HTML) as well as filenames and directories. For example:

  install_root refers to the directory where the Adapter software is installed, for example

  c:/Program files/install_root/Adapters
• Web addresses are displayed as hyperlinks. For example: http://www.EMC.com

Related documents

For more information on EMC software products, refer to the following manuals:

• **EMC Ionix IT Orchestrator Introduction and Planning Guide** - This manual provides an overview of the Orchestrator architecture and associated applications. It introduces the main components of the IT Orchestrator, Design Studio, Platform and applications.

• **EMC Ionix IT Orchestrator Installation and Configuration Guide** - This manual provides information on installing the EMC Ionix IT Orchestrator, Design Studio, Platform and associated plugins. Provides an overview on how to install the Orchestrator Studio on Windows and Linux platforms and describes the mechanism for installing the Orchestrator Platform in your managed environment.

• **EMC Ionix IT Orchestrator Administration and Operations Guide** - This manual identifies the administration tasks required to configure the Ionix IT Orchestrator Platform and describes the log files available for detailed analysis of running solutions.

• **EMC Ionix IT Orchestrator Developer Guide** - This manual provides an introduction to the Designer Studio and Lifecycle tools that enable you to design Services and Tasks in Orchestrator Studio. The manual identifies the procedures for building Services and Tasks and describes the configuration of the associated properties.
Introduction

About the Reverse Proxy Adapter

The Reverse Proxy Adapter was designed as a solution for customers that directly consume the web services of their Service Desk application from the Ionix IT Orchestrator application. With this approach and depending on their network setup, customers may need to allow access to their Service Desk application from the public Internet.

The Reverse Proxy Adapter provides a layer of indirection between the Ionix IT Orchestrator and the Service Desk application. The Reverse Proxy Adapter provides Security (HTTPS) and Authentication (HTTP Basic Authentication) up front so these configurations do not need to be set-up in the Service Desk application.

Example Process

The Reverse Proxy allows you to configure a service that will act like the target server. All requests are forwarded after security and authentication processes are performed.

For example:

```
reverseproxy.target.url.scheme = http
reverseproxy.target.url.port = 9080
reverseproxy.target.url.server = hostxyz
```

When the user makes a request to the proxy such as using the following URL:

```
https://adapter-host:9443/rproxy/services/SomeWebServiceEndpoint
```

This will be forwarded to the target by the Reverse Proxy as the following URL therefore providing HTTPS and Basic Authentication to the adapter:

```
http://hostxyz:9080/services/SomeWebServiceEndpoint
```

In this scenario, only access to adapter-host on port 9443 is required to access the web service on hostxyz.
EXAMPLE PROCESS
Chapter 4

System Configuration

This chapter identifies the requirements for configuring the EMC Adapter solution and discusses the following topics:

- Supported Operating Systems
- System Requirements
- Supported Databases

Supported Operating Systems

The Adapter software can be installed on the following operating systems:

- **Windows** - NT4, 2000, XP, 2003 Server, 2008 Server
- **UNIX** - Solaris 10 (Sparc), HP-UX 11 (Itanium), AIX
- **Linux** - x86, x64, Kernel 2.6 (or higher)

System Requirements

Before you install the Adapter software, your Adapter server must meet the following system requirements.

**Recommended**

The following are the recommended system requirements for installing the EMC Adapter software:

- **CPU**: Dual Core Processor running 2GHz or higher
- **RAM**: 2 GB
- **Disk Space**: 5 GB free disk space
- **Java Runtime Environment**: JRE 6.0 Update 16 (or above) is recommended. This is bundled with the Windows installer.
**Minimum**

The following are the minimum system requirements for installing the EMC Adapter software:

- **CPU**: Pentium Class or Equivalent Processor or higher
- **RAM**: 512 MB Free
- **Disk Space**: 2 GB free disk space
- **Java Runtime Environment**: JRE 6.0 Update 16 (or above) is recommended. This is bundled with the Windows installer.

**Supported Databases**

The EMC Adapter uses a relational Apache Derby database for the Persistence Queue. In addition to the Apache Derby database, the EMC Adapter supports the following external databases:

- Oracle
- DB2
- Microsoft SQL Server
Chapter 5

Software Installation

Topics covered in this section:

- Overview
- Configuring SSL Communications
- Installing a License File
- Installing the Adapter Software - Windows
- Running the Adapter Windows Service
- Installing Adapter Software - UNIX
- Running the Adapter UNIX Service
- Manual Installation Procedures
Overview

Based on your EMC configuration, the following components may be installed (see Figure 5-1):

- **EMC ITIL Adapters** – includes the Adapter software and license file.
- **EMC ITIL Adapter Agent** (Optional) – includes the Adapter URL Agent software or custom Agent. The Agent may be installed with the Adapter on the same server or on a separate server.
- **License files** – are provided by EMC Customer Support and must be installed in the appropriate directory.
- **EMC Ionix IT Orchestrator** (Optional) – includes the Ionix IT Orchestrator software and license file.

Figure 5-1 Installation Components
Configuring SSL Communications

To configure SSL communications between the Ionix IT Orchestrator and Adapter, complete the following steps.

1. From the Adapter server, obtain a copy of the Adapter SSL Certificate.
2. Verify that there is not an existing or incorrect certificate already installed in the Ionix IT Orchestrator keystore.
3. Copy the hostname.cer file to the Ionix IT Orchestrator server.
4. Confirm that the JAVA_HOME is set to Java 6. Once the JAVA_HOME is verified, execute the following command to delete the existing certificate:

   ```
   "%JAVA_HOME%\bin\keytool" -delete -alias iwave -storepass changeit -keystore "%JAVA_HOME%\lib\security\cacerts"
   ```

5. Install the certificate into the Ionix IT Orchestrator keystore.
6. Confirm that the JAVA_HOME is set to Java 6. Once the JAVA_HOME is verified, execute the following command to store the certificate:

   ```
   "%JAVA_HOME%\bin\keytool" -import -alias iwave -v -file iwave.cer -keypass iwave -storepass changeit -keystore "%JAVA_HOME%\lib\security\cacerts"
   ```

Figure 5-2  SSL Configuration
INSTALLING A LICENSE FILE

Installing a License File

When the Adapter software is purchased, a software license file (in XML format) is provided by EMC Customer Support. The license file identifies the following information for each Adapter server included in the EMC software configuration:

- the types of Adapters that can be installed on the server
- the Adapter software version
- the expiry date for the Adapter software
- the license key

During Adapter software installation, the user identifies the location of the license file. The installer then copies the license file from the original location to the following EMC Adapter installation directory:

```
install_root/adapters-2.4/conf
```
Installing the Adapter Software - Windows

This section identifies the steps required for installing the EMC Adapter software.

Before You Begin

Before you install the EMC Adapter software, complete the following tasks:

- Identify the location of the Adapter license file.
- Close all Windows applications.
- Determine the type of Agent you are installing (if any).
- Determine which Adapter will be installed.
- Determine on which server the Adapter and Agent software will be installed.

Start the Installer

To install the EMC URL Agent and/or Adapter software, complete the following steps:

1. Access the location of the Adapter installer executable. Double-click the `setup-Adapter-winxx_2.4.exe` file.
2. The Open File - Security Warning window displays. Click Run.
3. The InstallAnywhere progress screen displays.

Figure 5-3 Installer Progress Screen
4. The progress screen displays.

Figure 5-4  Progress Screen

5. After the InstallAnywhere Wizard finishes loading, the EMC Adapters Installer Introduction screen displays. From this screen, select the Next button to continue.

Figure 5-5  Introduction Screen
6. The **Choose Install Set** screen displays. From this screen, select the appropriate installation option.
   - For instructions on installing your Adapter, refer to "Installing the Adapter".

**Figure 5-6** Choose Install Set
Installing the Adapter

1. From the Choose Install Set screen, select the **Reverse Proxy Adapter** option. Click **Next**.

*Figure 5-7  Choose Install Set Screen*
2. The **Choose Install Folder** screen displays with the default installation folder. To change the install folder, click the **Choose** button and navigate to the appropriate installation directory. Click **Next**.

**Figure 5-8** Choose Install Folder Screen
3. The **Choose Shortcut Folder** screen displays. From this screen, specify the areas where the software shortcut icons will be created. Click the **Next** button to continue.

   - **In a new Program Group** – creates a new program group with the specified name and includes a shortcut in the Windows All Programs list.
   - **In an existing Program Group** – includes a shortcut in an existing program group.
   - **In the Start Menu** – includes the Adapter shortcut in the Start menu.
   - **On the Desktop** – creates an Adapter shortcut on the desktop.
   - **In the Quick Launch Bar** – includes the Adapter shortcut in the Quick Launch menu bar.
   - **Other** – specify a custom directory location by selecting the **Choose** button.
   - **Don’t create icons** – Select this option if you do not want to create an Adapter shortcut.
   - **Create Icons for All Users** – Select this option to create shortcuts for all users configured on the server. The shortcuts are only added to the Start Menu and Desktop.

**Figure 5-9** Choose Shortcut Folder Screen
Note: If you choose not to create a menu shortcut, you can start the Adapter framework using the `startAdapters.bat` file. This file is located in the `install_root\bin` directory.

4. The **Choose License File** screen displays. From this screen, verify that the license file location identified in the **Please Choose License File** field, is correct. To change the license file location, select the **Choose** button and select the appropriate directory. Click **Next**.

Figure 5-10  Choose License File Screen
5. The **Enable Security** screen displays. On this screen, the **Use HTTPS (port 9443)** check box is selected by default. This option uses SSL for secure Web-based communications. To use standard unsecured HTTP Web-based communications, un-check this box. Click **Next**.

**Figure 5-11** Enable Security screen
6. The Web Service Authentication screen displays. From this screen, enter the **Username** and **Password** you want to use for authentication. You must enter the password twice for confirmation purposes. Click **Next**.

**Figure 5-12** Web Services Authentication
7. The Monitoring Server Configuration screen displays. Based on your Adapter configuration, determine the appropriate step to complete:

- If you are monitoring the Adapter with the Monitoring Server and the Monitoring Server is on a different host from the Adapter, select the **Specify host for Monitoring Server** checkbox. Identify the following information and click **Next**:
  - **URL** - identifies the URL location of the Monitoring Server.
  - **Username** - identifies the username for the Monitoring Server.
  - **Password** - identifies the password associated with the username.
- If you are not using a Monitoring Server or your Monitoring Server is running on the same server as the Adapter, leave the checkbox unchecked. Click **Next**.

**Figure 5-13 Monitoring Server Configuration**
8. The Pre-Installation Summary screen displays. Using the scroll bar, review the information before continuing the installation process.
   — If the information is correct, click Next.
   — If the information is incorrect, click Previous and change the configurations on the previous screen. Click Next when done.

Figure 5-14 Pre-Installation Summary
9. The Adapters Configuration screen displays. Click **Next**.

**Figure 5-15** Adapters Configuration
10. The Reverse Proxy Adapter Configuration screen displays. From this screen, specify the following information and click **Next**.

- **URL Path** - This is the base URL to be removed when requests are made to the target. This should always be `/rproxy` unless the name of the Reverse Proxy web application has been changed (by changing the directory name).

- **Authentication Enabled** - Indicates if HTTP Basic Authentication should be enabled to connect to the proxy.
  - **Username** - the core adapter framework username. If specified, this username should be used with the password below as HTTP Basic Authentication to the target server.
  - **Password** - the core adapter framework password. If the username is specified, this is the password used for HTTP Basic Authentication.

- **Scheme** - The scheme or protocol to be used to get to the target server. This should be HTTP or HTTPS.

- **Port** - The port number to reach the target server. Example: 9080 for default HTTP.

- **Server** - The server that all requests should be forwarded to that come to the proxy.

*Figure 5-16 Reverse Proxy Adapter Configuration*
11. The Preparing to Install screen displays. Click **Install**.

**Figure 5-17** Preparing to Install
12. The Installing Adapters screen displays. This screen displays the progress of the installation process. During this step, the Windows Service is installed and a keystore file is generated for SSL.

Figure 5-18 Installing the Adapter
13. The **Start Service** screen displays. From this screen, select the **Start the Adapter Service** checkbox if you want to start the Adapter service during the installation process.

If you do not start the Adapter Service during installation, the Adapter service can be started manually. For more information on starting and stopping the Windows Service, refer to "Running the Adapter Windows Service" on page 39. Click **Next**.

**Figure 5-19** Start Service
14. If you selected the **Start the Adapter Service** option, the following screen displays.

**Figure 5-20** Starting the Adapters Service
15. The **Install Complete** screen displays. Click **Done** to exit the installer.

**Figure 5-21** Install Complete
Running the Adapter Windows Service

The Adapters Windows Service is automatically installed during the installation process and if selected, is automatically started, as well.

**Shortcuts**

Based on which option was selected during installation, shortcuts may be provided from the Windows Start menu for starting and stopping the Windows Service.

- To start the Adapters Service, select **Start>All Programs>install_root>adapters-2.4>Start Adapters Service.**
- To stop the Adapters Service, select **Start>All Programs>install_root>adapters-2.4>Stop Adapters Service.**

**Executable Location**

Two executable files are provided for interacting with the Windows Service:

- **adapters.exe**: The main executable that launches the service.
- **adaptersw.exe**: The executable for starting the Service Manager application.

These files are located in the installation `bin` directory: `install_root/bin`

**Service Manager Application**

The Service Manager application is an application that allows you to configure options of the service such as turning on debugging. The service can also be started and stopped from this console.

To start the Manager application, double-click on the **adaptersw.exe** file. The Adapters Properties screen displays (see Figure 5-22).
**Figure 5-22** Adapters Properties - Services Manager Console
Installing Adapter Software - UNIX

Before You Begin

Before installing the Adapters on the UNIX platform, you must complete the following tasks:

- For UNIX platform:
  - Install the Java Runtime Environment (JRE) 6 Update 11.
  - Java 6 must be installed and run on the local computer. If multiple Java platforms are present on the local computer, make sure that %JAVA_HOME% points to Java 6 when running the Adapter.

- For both UNIX and Linux (optional):
  - Create an “admin” user for running the framework.

Installing the Software

To install the Adapter software on a UNIX platform, run the appropriate setup application from the Linux or UNIX prompt.

- **Linux Installer**: setup-Adapter-linux_2.4.bin
- **Other UNIX Platforms**: setup-Adapter-unix_2.4.bin

Running the Adapter UNIX Service

The bin/service.sh script starts and stops the framework as a service. A special file service.PID is created when the service is started. This is used to find the PID of the running service. Do not delete this file manually.

- **Starting the service**: ./service.sh start
- **Stopping the service**: ./service.sh stop
- **Restarting the service**: ./service.sh restart
- **Service Status**: ./service.sh status

If you wish the service to start automatically at boot time, the following is a sample **init.d** script you can use to call the service.sh script:

```bash
#!/bin/sh
IWAVE_USER=admin
IWAVE_SERVICE=/usr/IonixITO/adapters-2.4/bin/service.sh
su - $(IWAVE_USER) -c "$IWAVE_SERVICE" $1
```
Manual Installation Procedures

If an installation is being performed on an unsupported operating system, the EMC Adapter can be installed manually without the installer.

Before You Begin

Before manually installing the Adapter software, you must complete the following tasks:

- Ensure that you are installing on either a Windows or UNIX platform.
- Java 6 must be installed and run on the local computer. If multiple Java platforms are present on the local computer, make sure that %JAVA_HOME% points to Java 6 when running the EMC Adapter.
- An Adapter license file (license.xml) must be copied to the install_root\conf directory. If you do not have a license file for your Adapter, contact your EMC Sales Representative.

Install the Files

To install the files manually on a Windows or UNIX platform, complete the appropriate steps.

For Windows:

1. Contact EMC Customer Support and request a zipped copy of the Adapter installer to be placed on the EMC FTP site.
2. From the Adapter server, download the zipped file.
3. Unzip the file and place the extracted file(s) in any directory on any machine with network access to the computer running the endpoint application.

Note: For Windows installations, there are no registry keys to edit. The location of these files is referred to as %IonixITO_HOME%.

For UNIX and Linux:

1. Contact EMC Customer Support and request a tarred/zipped copy of the Adapter installer to be placed on the EMC FTP site.
2. From the Adapter server, download the tarred/zipped file in binary format. Place the .zip file in the directory where you want the software installed. For example: /opt/EMC
3. Extract the files included in the zip file. This will create an adapters directory with several sub directories.

   jar xfv adapter-2.4.zip
4. Change to the adapters/conf directory.

5. Copy your license file to the conf directory. Ensure that the license file is named license.xml.

6. Change to the adapters/endpoints directory. Access the directory for your endpoint adapter.

7. Copy the endpoint-spring.xml file to the adapters/conf directory.

8. Using the propertiesEditor.bat utility, modify the following parameters in the framework.properties file.
   - framework.username
   - framework.password - use the -h option when setting the password.
   - endpoint.datasource.jdbc.url
   - endpoint.datasource.jdbc.username
   - endpoint.datasource.jdbc.password

9. Change to the adapters/bin directory

10. Run the setup.bat script.
    ./setup.bat

11. Install the Adapter as a service.
    ./service.bat install

12. Start the adapters service.

13. After a few seconds, change to the adapters/log directory and view the framework.log file. At the bottom of the file, the log should indicate that the adapter framework has started.
Framework Properties

During installation, the framework.properties file is installed in the conf directory. The framework.properties file includes configuration parameters for the Adapter Framework. Any parameters specified in this file are automatically recognized by the external (framework.xml) and internal (inside the framework jars) spring configurations. These parameters allow the base configuration to be used and only a few parameters should be changed.

The framework.properties file is in the standard Java Properties format. The file can be edited using a text editor or the properties in the file can be edited using the PropertiesEditor command.

The properties in the framework.properties file that are encrypted are shown in the format: "ENC;XXXXX" where XXXXX is the encrypted data.

Note: The Properties Editor tool is only capable of creating these properties and not reading them.

Properties Editor Tool

The Properties Editor is a tool used to edit configuration properties in the framework.properties file. This command allows encrypted parameters to be specified.

Starting the Tool

To start the Properties Editor tool, execute the following batch file or shell script from the command line:

- For Windows: bin/propertiesEditor.bat
- For Unix: bin/propertiesEditor.sh
**Using the Command**

When entering the `PropertiesEditor` command, multiple options can be specified in a single invocation of the command.

**Usage**

```
PropertiesEditor [options] <target-file>
```

**Options**

- `-p <key>=<value>` Sets a property with the key and value specified in plain text.
- `-e <key>=<value>` Sets a property with the key and value specified in encrypted data.
- `-h <key>=<value>` Sets a property with the key and value specified in SHA-1 Hashed format. This is used for framework passwords.
- `-m <file>` Merges multiple properties from the file specified to the target file.

**Examples**

The following are examples of using the Properties Editor tool:

- To set the framework username and password for authentication, enter the following command:
  ```
  bin/propertiesEditor.bat -p framework.username=admin -h framework.password="Hashed String" conf/framework.properties
  ```
  
- To set the Adapter hostname, username and password, enter the following command:
  ```
  bin/propertiesEditor.bat -p adapter.host=hostname -p adapter.username=Demo -e adapter.password=password conf/framework.properties
  ```

**Parameters**

The `framework.properties` file includes two types of properties:

- global framework properties
- Adapter-specific properties

**Global Framework Properties**

These global framework properties are automatically configured for all Adapters.
### Framework Authentication

The following properties are used for the Web Services HTTP authentication:

- **framework.username** - Username to log into the provided web services.
- **framework.password** - Password to log into the provided web services.
- **framework.admin.password** - Administrator password to restart the adapters over JMX.
- **framework.auth.enabled** - If this is set to true, authentication will be required. (Default: true)

### Default base URLs to Exposed Services

The following properties configure the base URLs for the Adapter services:

- **services.url** - Base URL for all adapter operations services (https://${hostname}:9443/services).
- **client.url** - Base URL for client acquisition service (https://${hostname}:9443/services).
- **notification.url** - Base URL for notifications (http://${hostname}:9080/services/notification).
- **acquisitionevent.url** - Base URL for acquisition events (http://${hostname}:9080/services/acquisition-event).

### DataSource Properties

The following properties can be modified to enable support for another database instead of the default (Derby) Persistence database:

- **datasource.jdbc.driver** – identifies the database driver (i.e., com.microsoft.sqlserver.jdbc.SQLServerDriver).
- **datasource.jdbc.url** – identifies URL to the database (i.e., jdbc:sqlserver://hostname:1433;DatabaseName=testdb).
- **datasource.jdbc.username** - identifies the username to log into the database.
- **datasource.jdbc.password** – identifies the password associated with the database username. If required, the password can be encrypted.

### Database Monitoring

The following properties can be modified to enable monitoring tasks on the database. Only one Adapter per database can have this enabled.

- **datasource.monitor.enabled** - identifies if database monitoring is enabled. (Default: true)
Acquisition monitoring

The Adapters automatically send the `adapter.acquisition.queuesize` and `adapter.acquisition.queuewait` events to the Monitoring Server based on the health of the Adapter queue. The following parameters are used to configure Acquisition monitoring.

- `acquisition.monitor.enabled` - identifies if Acquisition monitoring is enabled. (Default: true)
- `acquisition.monitor.queue-threshold` - identifies the number of items allowed in the queue before the `queuesize` event is sent. (Default: 500)
- `acquisition.monitor.queue-wait-hours` - identifies the number of hours an item must be on the queue before the `queuewait` event is sent. (Default: 24)
- `acquisition.bufferqueue.limit` - identifies the maximum number of items that are allowed in the buffer queue. (Default: 500)

Automated Alerts

If the Monitoring Server is on a different host from the Adapter, the following configuration are updated to include the connection settings for the Monitoring Server. These parameters identify the location to send automated alerts. If you are using a Monitoring Server and it is installed on the same server as your Adapter, leave these parameters blank.

- `framework.events.url` - identifies the Monitoring Server URL location to send events.
- `framework.events.username` - identifies the Monitoring Server username for basic authentication.
- `framework.events.password` - identifies the password associated with the Monitoring Server username. The password can be encrypted.
PARAMETERS
Reverse Proxy Framework Properties

The `framework.properties` file includes the following properties for the Reverse Proxy Adapter:

- **reverseproxy.request.url.path** - This is the base URL to be removed when requests are made to the target. This should always be `/rproxy` unless the name of the reverse proxy web app has been changed (by changing the directory name).
- **reverseproxy.auth.enabled** - Indicates if HTTP Basic Authentication should be enabled to connect to the proxy. The user/password used are the core adapter framework username and password.
- **reverseproxy.target.auth.username** - If specified, this username should be used with the password below as HTTP Basic Authentication to the target server.
- **reverseproxy.target.auth.password** - If the username is specified, this is the password used for HTTP Basic Authentication.
- **reverseproxy.target.url.scheme** - This is the scheme or protocol to be used to get to the target server. This should be `http` or `https`.
- **reverseproxy.target.url.port** - This is the port number to reach the target server. Example: 80 for default HTTP.
- **reverseproxy.target.url.server** - This is the server that all requests should be forwarded to that come to the proxy.

**Figure 1-1  Example Framework Properties file - Reverse Proxy**

```yaml
## Reverse Proxy
reverseproxy.request.url.path = /rproxy
reverseproxy.auth.enabled = false
reverseproxy.target.auth.username =
reverseproxy.target.auth.password =
reverseproxy.target.url.scheme = http
reverseproxy.target.url.port = 8080
reverseproxy.target.url.server = localhost
```
Chapter 7

Custom Configuration

Topics covered in this section:

- Client Acquisition Service
- Adapter High Availability
- Java to Adapter Integration
- Persistence Database Management

Client Acquisition Service

The Client Acquisition service allows customers to leverage the EMC Acquisition Services with their data. Customer data can be added to the existing EMC Acquisition system. Processes in Ionix IT Orchestrator can be configured to feed off of their data similar to the existing Adapter workflows. Two services are provided for getting data into the Acquisition Queue. Both provide the exact same features, but allow customers to choose the easiest integration point.

Data Format

The data must be provided in the following XML format:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<client-object>
  <!-- Any client XML can go in here -->
</client-object>
```

Object Format

Schema for the client object (client-object.xsd):

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<x:schema version="1.0" xmlns:x="http://www.w3.org/2001/XMLSchema">
  <x:s:element name="client-object" type="clientDefinedObject"/>
</x:schema>
```
To provide data to the acquisition queue, customers must always specify the serviceId and objectType along with the object data.

- **serviceId** - Unique identifier of the service that placed this data here. This is used to differentiate between different applications that push data into the queue.

- **objectType** - Unique type of the object. This should not be one of the built in types (Incident, Event, CI, Change, Problem, ITObject). It can be any unique name to identify the customer's data.

The return for the service provides the ID of the transaction back to the client. This can be used to correlate the call with the transaction that appears in the queue.

**RESTful Client Acquisition**

For the Client Acquisition endpoint, HTTP POST to the following address:

https://localhost:9443/services/client-acquisition/{serviceId}/{objectType}:

Replace `{serviceId}` and `{objectType}` with the desired values.

- HTTP Basic Authentication is expected to authenticate the user.
- If a problem occurs, and HTTP 500 (Internal Error) response code is returned, the content of the response contains the error message. A HTTP 200 code indicates the request was processed normally. Any HTTP 4xx codes indicate some sort of client error.

**Note:** If you have a form with a ":" colon in it, use the ASCII Hex decimal with a "%" sign prefix to escape it in the Restful service URL.

**Calling the RESTful Client Acquisition from curl**

In order to call the RESTful Client Acquisition from curl, the URL Agent must be installed. Once the URL Agent is installed, complete the following tasks:

1. Create an XML file with the root node of the client-object.

   ```xml
   <?xml version="1.0" encoding="UTF-8"?>
   <client-object>
     <!-- Any client XML can go in here -->
   </client-object>
   ```

2. From the command line, call the following command:
python agent.py -l -v -x {filename} https://iwave:password@localhost:9443/services/client-acquisition/{serviceId}/{objectType}

3. An HTTP/1.1 200 OK response is sent and the data should be in the Acquisition Queue.

**SOAP Client Acquisition Web Service**

The following is the location of the SOAP Client Acquisition Service WSDL:
https://localhost:9443/services/ClientAcquisitionService?wsdl

If an error occurs, a web service fault is thrown matching the framework exception com.iwave.acquisition.client.AcquisitionException. This exception contains a message with the details of the problem. HTTP Basic Authentication is expected to authenticate the user.

**Running on another port**

To run the Client Acquisition service from another port, you must modify the client-acquisition-spring.xml file. Figure 7-1 identifies the spring configuration file for the client acquisition service. In this file, perform the following modifications:

1. Replace the \$\{services.url\} with the required URL and port number (i.e., 9080).

   **Note:** If you specify port 9443, you must use HTTPS in the URL.

2. If you need to disable authentication, delete the following three lines in the two places they occur.
   
   \[
   \begin{align*}
   \text{<jaxws:inInterceptors>}
   \text{<ref bean="securityInterceptor"/>}
   \text{</jaxws:inInterceptors>}
   \end{align*}
   \]

3. To start a second copy of the Acquisition Service, copy the modified file to the conf directory.
Figure 7-1  Spring Configuration

```xml
<beans xmlns="http://www.springframework.org/schema/beans"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:jaxws="http://cxf.apache.org/jaxws"
      xmlns:jaxrs="http://cxf.apache.org/jaxrs"
      xmlns:lang="http://www.springframework.org/schema/lang"
      xsi:schemaLocation="
        http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans.xsd
        http://cxf.apache.org/jaxws
        http://cxf.apache.org/schemas/jaxws.xsd
        http://cxf.apache.org/jaxrs
        http://cxf.apache.org/schemas/jaxrs.xsd
        http://www.springframework.org/schema/lang
        http://www.springframework.org/schema/lang/spring-lang-2.0.xsd">

  <bean id="clientAcquisitionService2"
        class="com.iwave.acquisition.client.ClientAcquisitionService">
    <property name="queue" ref="acquisitionQueue"/>
  </bean>

  <!-- Restful exception handler -->
  <bean id="acquisitionExceptionProvider2"
        class="com.iwave.acquisition.client.AcquisitionExceptionMapper"/>

  <!-- Endpoints -->
  <jaxws:endpoint
    id="clientAcquisitionServiceEndpoint2"
    implementor="#clientAcquisitionService2"
    implementorClass="com.iwave.acquisition.client.ClientAcquisitionService"
    address="${services.url}/ClientAcquisitionService2">
    <jaxws:inInterceptors>
      <ref bean="securityInterceptor"/>
    </jaxws:inInterceptors>
  </jaxws:endpoint>

  <jaxrs:server id="clientAcquisitionServiceRest2"
    address="${services.url}/client-acquisition2">
    <jaxrs:inInterceptors>
      <ref bean="securityInterceptor"/>
    </jaxrs:inInterceptors>
    <jaxrs:serviceBeans>
      <ref bean="clientAcquisitionService2"/>
    </jaxrs:serviceBeans>
    <jaxrs:providers>
      <ref bean="acquisitionExceptionProvider2"/>
    </jaxrs:providers>
  </jaxrs:server>
</beans>
```
Adapter High Availability

The EMC Ionix IT Orchestrator Adapter solution may be scaled to allow more Adapters in the network configuration. Adapter scaling may be performed for the following reasons:
• to increase the number of transactions being processed
• to provide high-availability (i.e., redundancy)
• to handle additional simultaneous requests

Figure 7-2 provides an illustration of a network configuration with multiple Adapters utilizing an external Acquisition database.

Figure 7-2  Adapter Scaled Network Configuration

Operations Services Scaling

The Operations Web Services can be scaled easily. The operations are essentially stateless, allowing a second adapter to be installed on another system that can handle the operations. If the primary Adapter fails, workflows can be updated to switch to use a secondary Adapter’s operations services.

Acquisition Services Scaling

The Acquisition Services component of the Adapter architecture can be scaled by completing the following steps:
1. Offload the Acquisition Queue to an external database.
2. Use the external database with multiple Adapters.
Use the External Database for Multiple Adapters

Each Acquisition Service is pulling from the same shared queue. A second adapter install can be used to handle additional acquisitions. Workflows can be updated to switch to the second adapter if the acquisition queue operations fail on the primary server.

Adapter Failover Infrastructure

Since it is difficult to add logic for Adapter failover within workflows, specific infrastructure must be implemented to handle Adapter failover. The following are options for implementing high-availability for the EMC Adapters:

- **Round-Robin DNS Server** - A Round-Robin DNS can be used to allow a single hostname to provide access to multiple IP addresses representing the multiple Adapter environments installed. With this configuration, if the primary Adapter server goes out-of-service, the DNS server can point users to the secondary Adapter server that is in-service.

- **Highly Available HTTP Proxy Server** - A highly available HTTP Proxy server can be used to forward requests between Adapters. When the primary Adapter server goes out-of-service, the HTTP proxy server should be able to forward requests to a secondary Adapter server that is in-service.

Agent Scaling

It is assumed that if the endpoint application is up and running, it should be able to invoke the Agent script running within that environment. The agent needs to only notify one acquisition adapter that is running. The reason is that the message only needs to be added to the queue once.

There are two options for this:

- If the customer has a Round-Robin DNS or proxy set up that allows access to any of the installed adapters that are running, this can be used from the agent. The agent would point to the single hostname. One of the adapters receives the request and add it only once to the queue.

- If multiple URLs are present on the agent command line, it makes the request to only one of them, the first available one. Store-forwarding is still provided but is only used if none of the services can be reached.
Java to Adapter Integration

If a Java application must be used to push information into any of the Adapter RESTful web services, specific Java code must be used. Figure 7-3 includes a Java code sample used to perform an HTTP POST or GET with the EMC Adapter.

Prerequisites

In order to use this code, you need to download the Apache Commons HTTP Client version 3.1. The Commons HTTP Client can be downloaded from the following location:

http://hc.apache.org/downloads.cgi

From the Apache Http Components Downloads page, scroll down and select to download the Commons HttpClient 3.1. To use this client, you also need the following jar files:

- commons-httpclient-3.1.jar
- commons-codec-1.3.jar
- commons-logging-1.1.1.jar
package com.iwave.net;
import java.io.IOException;
import org.apache.commons.httpclient.HttpClient;
import org.apache.commons.httpclient.HttpException;
import org.apache.commons.httpclient.HttpStatus;
import org.apache.commons.httpclient.methods.GetMethod;
import org.apache.commons.httpclient.methods.PostMethod;
import org.apache.commons.httpclient.methods.StringRequestEntity;
/**
 * Example of making an HTTP GET or POST in Java.
 * *
 */
public class SimpleHTTPClient {

    public String post(String url, String xml) throws HttpException, IOException {
        HttpClient client = new HttpClient();
        PostMethod method = new PostMethod(url);
        try {
            if (xml != null) {
                method.setRequestEntity(new StringRequestEntity(xml, "text/xml", "UTF-8"));
            }
            // Execute the method.
            int statusCode = client.executeMethod(method);
            if (statusCode != HttpStatus.SC_OK) {
                throw new IOException("HTTP POST method failed: " + method.getStatusLine());
            }
            return method.getResponseBodyAsString();
        }
        finally {
            method.releaseConnection();
        }
    }

    public String get(String url) throws HttpException, IOException {
        HttpClient client = new HttpClient();
        GetMethod method = new GetMethod(url);
        try {
            // Execute the method.
            int statusCode = client.executeMethod(method);
            if (statusCode != HttpStatus.SC_OK) {
                throw new IOException("HTTP GET method failed: " + method.getStatusLine());
            }
            return method.getResponseBodyAsString();
        }
    }
}
Persistence Database Management

This section discusses the following maintenance tasks that can be performed on the Persistence database.

- Removing Transactions
- Reclaiming Unused Disk Space
- Connecting to the Database
- Changing the Database Port Number
- Offloading the Acquisition Queue to an External Database

Removing Transactions

The pruning service removes all transactions that are older than 7 days and are in the “Error” state from the Persistence database. The service runs every night at 3:00 a.m.

Reclaiming Unused Disk Space

After large amounts of data have been deleted or updated on the Derby database, a Derby table or index can contain unused space. By default, Derby does not return unused space to the operating system. However, Derby does provide a way to reclaim unused space in tables and associated indexes.

If you determine that a table and its indexes have a significant amount of unused space, use one of the following procedures to reclaim the space:

- SYCS_UTIL.SYSCS_COMPRESS_TABLE - is guaranteed to recover the maximum amount of free space, at the cost of temporarily creating new tables and indexes before the statement in committed.
- SYCS_UTIL.SYSCS_INPLACE_COMPRESS_TABLE - attempts to reclaim space within the same table, but cannot guarantee it will recover all available space. This procedure uses no temporary files and moves rows around within the same table or index.

For more information, refer to the Apache Derby Administration Guide.

To reclaim space in the table and recreate the indexes, execute the following script from adapters/bin directory:

runScript.bat derbyMaint
Connecting to the Database

To connect to the Adapter Derby database over the network, use the Java driver \texttt{derbyclient-10.4.2.0.jar}. The following URL will connect to the Derby database running in the Adapter.

\texttt{jdbc:derby://HOST:1528/persistence-db}

Changing the Database Port Number

To change the port number of the database, modify the \texttt{derby.drpa.portNumber} in the \texttt{system.properties} file located in the \texttt{conf} directory.

Offloading the Acquisition Queue to an External Database

In addition to the Apache Derby database, the Acquisition Queue can be offloaded to an external database. To enable support for another database, complete the following steps:

1. Download the JDBC driver for the required platform and place the file in the \texttt{lib/adapters} directory in the adapter install directory.
2. Modify the following properties in the Adapter \texttt{framework.properties} file:
   - \texttt{datasource.jdbc.driver}: identifies the database driver name (i.e., \texttt{com.microsoft.sqlserver.jdbc.SQLServerDriver}).
   - \texttt{datasource.jdbc.url}: identifies the URL to the database (i.e., \texttt{jdbc:sqlserver://hostname:1433;DatabaseName=testdb}).
   - \texttt{datasource.jdbc.username}: Username
   - \texttt{datasource.jdbc.password}: Password. This can be encrypted if desired.
3. Restart the Adapter.

For the database to be highly available, database clustering technologies should be used. Refer to the appropriate database documentation for more information.
Managing the Adapter

In order to manage the EMC Adapters, the Java Console application may be installed on your workstation. The Java Console graphical user interface is a management tool that complies to the Java Management Extensions (JMX) specification. Java Console uses the Java Virtual Machine (Java VM) to provide information about the performance and resource consumption of the Adapters installed in your network.

This application allows you to perform the following functions:
• list the names of the Adapters that are licensed and loaded
• provide the status of the endpoint applications
• restart all Adapters
• provide a status of items in the Acquisition Queue

**Note:** For information on monitoring Events that are generated on the EMC Adapter and Ionix IT Orchestrator, refer to the “EMC Monitoring Server Installation and Configuration Guide”.
Prerequisites

In order to use the Java console to manage the Adapter services, the following tasks must be completed:

- The Adapter software must be installed on your computer.
- The following Java Development Kit (JDK) package must be installed on your computer:
  
  **Java JDK1.6.0**

Connecting to the Adapter Java Process

There are two ways you can connect to the Adapter Java process:

- Connect to the local JMX instance. By default, JMX is not available for remote connection. Java allows you to connect to a Java process running as the same user using the jconsole.
- JMX can be enabled in the process itself for remote access. Remote access can make use of SSL and authentication. `setEnv` – Enable JMX monitoring in
setEnv20 identifies the Java properties that can be set to enable JMX without any security or authentication. These properties are included in the setEnv.bat/.sh file, which is located in the install_root\adapters-2.4\bin directory. The JMX monitoring properties are disabled by default. To enable JMX monitoring, delete the rem comment from the setEnv file. PORT can be replaced with any port name that is desired.

- `-Dcom.sun.management.jmxremote.port=PORT`
- `-Dcom.sun.management.jmxremote.authenticate=false`
- `-Dcom.sun.management.jmxremote.ssl=false`

**Figure 8-2** setEnv – Enable JMX monitoring in setEnv

---

### Starting the Java Console

The Java Console is started by the `jconsole` executable. This executable can be found in the `JDK_HOME/bin` directory, where `JDK_HOME` is the directory in which the JDK software is installed. If this directory is in your system path, you can start the Java Console by typing `jconsole` in a command (shell) prompt. Otherwise, you have to type the full path to the executable file.

---

### Management Beans

The following Adapter management beans can be used through JMX to manage the Adapter:

- `com.iwave:name=Adapters`
- `com.iwave:name=AcquisitionQueue`
- `com.iwave:name=FrameworkAdmin`

**com.iwave:name=Adapters**

The Adapters Management bean allows the user to see which Adapters are loaded and the status of the endpoint applications.

### Methods

The following Methods are supported for the Adapter Management bean:

- `listAdapterNames()` - Gives a list of the names of the adapters that are licensed and loaded.
Management Beans

- getEndpointStatus(endpointName) - Gives the status of an adapter. The supplied parameter endpointName should be one of the values returned from listAdapterNames(). Possible values for endpoint status are:
  - Unknown - Indicates that this adapter does not support the status check. Currently the only adapters that support this operation are BMC Remedy and CA-USD.
  - Available - Indicates that the endpoint application is running and responds to a simple command.
  - NotAvailable - Indicates that the endpoint application cannot be reached or fails to respond to a simple command.

com.iwave:name=AcquisitionQueue

The Acquisition Management bean allows the user to see the status of the Acquisition Queue without directly querying the database.

Methods

The following Methods are supported for the Acquisition Queue bean:

- countErrors(serviceId, objectType) - Gives a count of the items on the queue in the Error state. This is the same as the acquisition service countErrors() method.
- countPoll(serviceId, objectType) - Gives a count of the items on the queue in the Poll state. This is the same as the acquisition service count() method.
- countBlocked(serviceId, objectType) - Gives a count of the items on the queue in the Blocked state.
- countTotal(serviceId, objectType) - Gives a count of all of the items on the queue regardless of state.
- queueWaitingTime() - Gets the queue waiting time for the top item on the persistence queue. The date of the first transaction is returned. This can give an idea of when the last transaction was processed and if the queue processor is stalled.

Each of the count methods take parameters of the serviceId and objectType. These must be valid serviceId and objectType or part of a word. % is a valid wildcard. If no objectType and serviceId are specified, the default will be to query all.

Note: The Acquisition Management Service is also exposed as a web service. The following is the WSDL for the Acquisition Management Service:

https://localhost:9443/services/AcquisitionManagementService?wsdl
com.iwave:name=FrameworkAdmin

The Framework Administration bean allows administration tasks to be performed on the Adapters.

**Methods**

The following Methods are supported for the Adapter Management bean:

- **restart (password)** - Restarts the adapters. All services are shutdown, configuration files reloaded and the services restarted. A parameter of the admin password is required.

- **framework.admin.password** - This is the new setting in framework.properties that contains the admin password hash. The default value for this password is adminpw. If the JMX service will be enabled, this password should be changed from the default.
This chapter covers the following topics pertaining to the maintenance and troubleshooting of the EMC Adapter software.

- **Logging**
- **Error Types and Meanings**
- **Re-Installing or Uninstalling the Windows Service**
- **Starting and Stopping the Adapter**
- **Uninstalling the Adapter Software - Windows**

## Logging

Logging is provided for both the Adapter Framework and the URL Agent. Before logs can be generated, logging must be turned on and the amount of details specified.

### Log Location

Based on the operating system, the EMC Adapter stores logs in following directories:

- **For Windows:** `C:\install_root\adapters\log`
- **For UNIX:** `/opt/install_root/adapters/log`

### Turning on Adapter Logging (Windows)

To turn on Adapter Windows logging, you must edit the `log4j.properties` file.

1. **From Windows Explorer, navigate to the `conf` directory:**
   
   `C:\install_root\Adapters\conf`

2. **Right-click `log4j.properties` file and click Open on the pop-up menu. If this is the first time you have opened the properties file, Windows displays the following dialog:**
Figure 9-1  Windows Unknown File Type

3. Click **Select the program from a list** option and click **OK**.
4. From the Open With List, select **Notepad**. Select Always use the selected program to open this kind of file and click **OK** to open Notepad.
5. In the **log4j.properties** file, find the Production Environment Configuration section and the line:
   
   ```
   log4j.rootLogger=INFO,File,ErrorAppender
   ```
   
   Replace the word **INFO** with **DEBUG**.
7. Find the line:
   ```
   log4j.logger.com.iwave=INFO
   ```
8. Replace the word **INFO** with **DEBUG**. The **log4j.properties** file should look like the file identified in **Figure 9-2**.
9. From the menu, select **File > Save** to save the changes.
10. From the menu, select **File > Exit** to close the file.
Figure 9-2  log4j.properties File

```
# Root logger. Indicates which Appenders
# Production Environment Configuration
log4j.rootLogger=INFO,File,ErrorAppender
#log4j.rootLogger=DEBUG,Console,File,ErrorAppender

# Console Appender
log4j.appender.Console=org.apache.log4j.ConsoleAppender
log4j.appender.Console.layout.ConversionPattern=%d [%t] %-5p [%c] %m%n

# File Appender
log4j.appender.File=org.apache.log4j.RollingFileAppender
log4j.appender.File.File=${iwave.home}/log/framework.log
log4j.appender.File.MAXFileSize=5MB
log4j.appender.File.Append=true
log4j.appender.File.layout=org.apache.log4j.PatternLayout
log4j.appender.File.layout.ConversionPattern=%d [%t] %-5p %c - %m (%F:%L) %n

# Debugging pattern with more information but slower
log4j.appender.File.layout=org.apache.log4j.PatternLayout
log4j.appender.File.layout.ConversionPattern=%d [%t] %-5p %c - %m (%F:%L) %n

# Error File Appender
log4j.appender.ErrorAppender=org.apache.log4j.RollingFileAppender
log4j.appender.ErrorAppender.File=${iwave.home}/log/framework-errors.log
log4j.appender.ErrorAppender.MAXFileSize=5MB
log4j.appender.ErrorAppender.MAXBackupIndex=5
log4j.appender.ErrorAppender.layout=org.apache.log4j.PatternLayout
log4j.appender.ErrorAppender.layout.ConversionPattern=%d [%t] %-5p %c - %m (%F:%L) %n

# IMS Events Appender (if enabled)
log4j.appender.Events=org.apache.log4j.RollingFileAppender
log4j.appender.Events.File=${iwave.home}/log/events.log
log4j.appender.Events.MAXFileSize=5MB
log4j.appender.Events.MAXBackupIndex=5
log4j.appender.Events.Append=true
log4j.appender.Events.layout=org.apache.log4j.PatternLayout
log4j.appender.Events.layout.ConversionPattern=%d [%t] %-5p %c - %m (%F:%L) %n

# IMS Logger settings
log4j.additivity.IMS=false
log4j.logger.IMS=INFO, Events

# Restrictions on the log level of certain packages
log4j.logger.org.springframework=WARN
log4j.logger.org.quartz=WARN
log4j.logger.org.mortbay=WARN
log4j.logger.org.eclipse.session=ERROR
log4j.logger.oracle.toplink=ERROR
log4j.logger.com.lmc=WARN
log4j.logger Sabha.INFO
log4j.logger好莱.INFO

# To enable logging for the adapter, this category should be set to DEBUG
log4j.logger.com.iwave=DEBUG
```
Error Types and Meanings

The URL Agent and EMC Adapter generate their own errors. Several types of errors can occur when several complex software applications interact. This complexity is expanded when considering that the applications may be communicating over vast distances (via the Internet) and that they are typically asynchronous.

The following types of errors that may occur when using the EMC Adapter:

- Recoverable
- Permanent
- Global Errors

Recoverable Errors

Recoverable errors go away in time without involving any particular change to the EMC configuration or the endpoint application. For instance, if the EMC URL Agent attempts to send an event notification to the EMC Adapter and the connection is down, the Agent continues sending the notification until the connection is restored. In this case, there is an error message in the log, but no action needs to be performed to return the Adapter to service. The Adapter is returned to service automatically.

Permanent Errors

Permanent errors cannot be recovered automatically and require some form or corrective interaction. For example, if the EMC user for the endpoint application does not have a high enough privilege to open a new incident, the attempt fails. No matter how many times the EMC user attempts to open a new Incident, the action continues to fail until the user account is modified to allow the action.

Global Errors

An error can thrown from either the Adapter, Ionix IT Orchestrator or Integration workflow. Some of these errors are results of errant configurations such as an incorrect workflow assumption, incorrect URL, or a bad SSL configuration. Other errors could result from an error in the software itself.
**Adapter Errors**

The table below indicates errors that can occur in the EMC Adapter. These errors can be generated by either the Acquisition or Operations Services.

<table>
<thead>
<tr>
<th>Error</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcquisitionException</td>
<td>A problem occurred while trying to retrieve a transaction from the adapter.</td>
</tr>
<tr>
<td>ObjectNotFoundException</td>
<td>Occurs when the object requested on a get, update, or remove operation is not found</td>
</tr>
<tr>
<td>UnsupportedOperationException</td>
<td>The method used on the adapter is not supported by the adapter.</td>
</tr>
<tr>
<td>OperationsException</td>
<td>Typically a recoverable error such as connectivity issues</td>
</tr>
<tr>
<td>NonRecoverableOperationsException</td>
<td>Permanent error condition.</td>
</tr>
</tbody>
</table>
Re-Installing or Uninstalling the Windows Service

If the Windows Service needs to be re-installed or un-installed, complete the following steps:

Re-installing the Windows Service

To re-install the Windows Service:

1. Access the bin directory:
   
   \install_root\adapters-2.4\bin

2. Run the installation script.
   
   service.bat install

Uninstalling the Windows Service

To uninstall the Windows Service:

1. Access the bin directory:
   
   \install_root\adapters-2.4\bin

2. Run the un-install script.
   
   service.bat uninstall

Starting and Stopping the Adapter

If the Windows Service has been uninstalled and the Adapter needs to be started, the startAdapters.bat file is included in the following directory:


   \install_root\adapters-2.4\bin\startAdapters.bat

This command starts the Adapter framework with a command window. To stop the Adapter framework, close the command window.
Uninstalling the Adapter Software - Windows

To uninstall the EMC Adapter software, complete the following tasks:
1. From the Start menu, select the **Uninstall Adapters** option.
2. The Uninstall EMC Adapters Introduction screen displays. From this screen, click **Uninstall**.

Figure 9-3  Uninstall Introduction
3. The Uninstaller begins to remove features associated with the installed Adapter. It then uninstalls the EMC Adapter files, shortcut, LaunchAnywhere features, folders, and registry.

Figure 9-4 Uninstall In Progress
4. The Uninstallation Complete screen displays. The files that were not installed may be displayed on the screen. These files must be removed manually. Click **Done**.

**Figure 9-5** Uninstallation complete
Index

A
acquisition
  monitoring 47
acquisition service
  scaling 55
adapter
  failover infrastructure 56
  management 61
Adapter Framework
  running as a Windows service 39
  starting 72
  stopping 72
agent
  scaling 56
alerts
  automated 47
  Apache Derby 16

B
base URLs 46

C
client acquisition service
  running on another port 53

D
database
  monitoring 46
databases
  enabling support for another database 46
  external
    user for multiple adapters 56
  supported 16
datasource properties 46

E
error messages
  types 70
  error types
    permanent 70
    recoverable 70
  errors 70
    Adapter 71

F
framework HTTP authentication 46
framework properties 44
  for the global framework 45

G
  global framework properties 45

H
  high availability 55

I
install folder 25
install set 23, 24
installer
  Linux 41
  starting 21
  Unix 41

J
Java console
  starting 63
Java Management Extensions 61
Java to Adapter integration 57
JMX 61
  management beans 63

L
license file 27
directory location 20
installing 20
log
location
for UNIX 67
for Windows 67
log4j.properties file 67
logging 67
turning on
for Windows 67

O
operating systems supported 15
operations service scaling 55

P
password
iMS 30
web service authentication 29
permanent errors 70
persistence database 59
changing the port number 60
connecting to 60
removing transactions 59
properties editor 44
command 45
starting 44

R
recoverable errors 70
RESTful Client Acquisition 52

S
security 28
shortcut folder 26
SOAP client acquisition service 53
software
uninstall
Windows 73
software installation 17
for Windows 21
manual 42
UNIX 42
Windows 42
system configuration 15
system requirements 15

troubleshooting 67
error types and meanings 70

U
URL
iMS 30
username
iMS 30
web service authentication 29

W
web service authentication 29
Windows service
executable location 39
manager application 39
manual install or uninstall 72
re-installing 72
running 39
uninstalling 72