AlarmPoint Systems

HP Software Certified Application

AlarmPoint for HP Operations Manager software for Windows

Validation Date
November 24, 2010
Version 2.2.4

Contents

1. Introduction

Welcome to the AlarmPoint for HP Operations Manager software for Windows integration. This document defines software requirements, installation and configuration instructions, and integration demonstrations for using HP Operations Manager software for Windows (OM-W) and AlarmPoint. These integration notes are intended for experienced HP consultants, system administrators, and other technical readers.

1.1 Benefits

With the AlarmPoint integration, the appropriate technician to resolve an issue can be notified directly via voice, email, pager, BlackBerry, or other device. Information about the failure will be presented to the event resolver and when the recipient selects a response on their remote device, AlarmPoint updates the OM-W Message in real-time.

The benefit is that this process is immediate – significantly faster than the time required for staff to notice the failures or malfunctions, determine who is on call, and manually notify the right person. In addition, the ability to take simple actions on the event from any device gives the event resolver a quick way to deal with many issues and communicate to other team members the current status of the event.

During the process, every notification, response, and action is logged in AlarmPoint. In addition, AlarmPoint automatically annotates the OM-W Message with process information.

The AlarmPoint product features a self-service web user interface to allow accurate assignment of responsible personnel for each job. This integration also includes an enhanced Subscription panel that allows AlarmPoint Users to subscribe to OM-W Messages. This Subscription Panel queries the OM-W Server directly in real time to retrieve lists of Nodes and Node Groups, eliminating the need to create and maintain the lists manually.

1.2 Summary

This integration supports Message notifications (from OM-W to AlarmPoint) through the evaluation of OM-W policies. It also supports inbound actions (from AlarmPoint to OM-W) of Acknowledge, Own, Ignore, Change Severity and Add Annotation.

You will need to modify this configuration to suit your particular business requirements and adjust it to suit your expected loads. This example features extensive logging from AlarmPoint to HP Operations Manager for Windows; in a high-volume production system, this can significantly affect performance. Consider your expected volume of alerts and server capacity when designing your own integration with AlarmPoint.

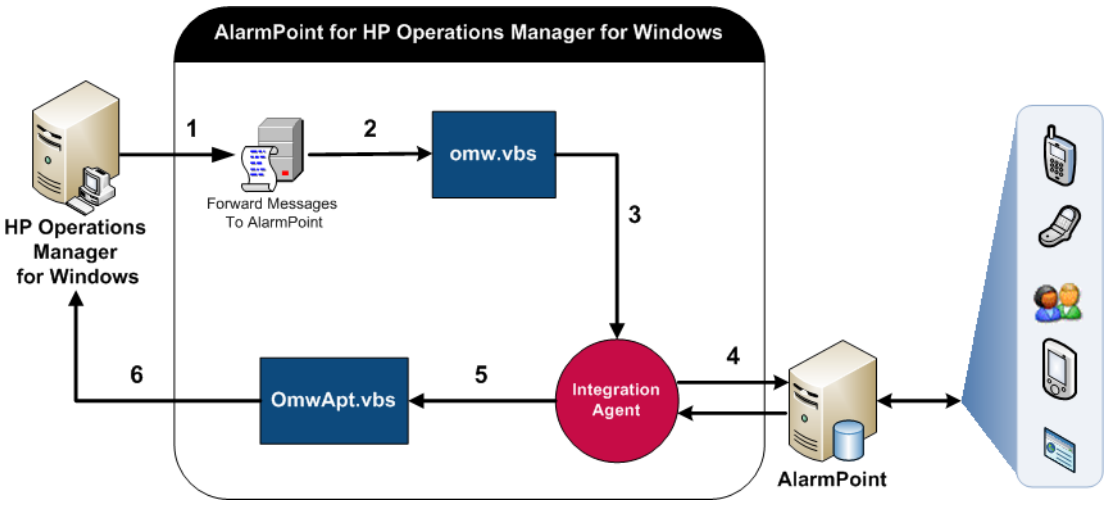
1.2.1 Overview

The basic steps of the integration process are as follows:

1. Install the AlarmPoint Integration Agent.
2. Install the configuration file and support scripts for the AlarmPoint Integration Agent.
3. Install the OM-W support files.
4. Import the Policies and Tools into OM-W.
5. Edit the OM-W Policy to reflect your unique business requirements.

1.2.2 Architecture

The following diagram provides a high-level overview of the major components for this integration:



1. When an event occurs on a system monitored by HP Operations Manager, OM-W creates a message which triggers the Forward Messages to AlarmPoint policy.
2. If the Policy's conditions are true (the default setting is for any message of "Critical" severity), the message details are sent to omw.vbs.
3. The enriched details are then sent to the AlarmPoint Integration Agent using an AlarmPoint AddEvent via HTTP POST.
4. The Integration Agent sends the event to AlarmPoint, which in turn notifies the Group defined in the originating message's Message Group field.
5. The notification response returns from AlarmPoint to the Integration Agent, which sends an OM-W Request to OmwApt.vbs via Response Action Scripting.
6. OmwApt.vbs sends a remote WMI action to OM-W to update the message.

1.3 System Requirements

The following products are used in this integration:

- HP Operations Manager software for Windows, version 8.10
- AlarmPoint 4.0.0 (patch 004 or later)
- AlarmPoint Integration Agent 4.0.0 (patch 001 or later)

Note: *The AlarmPoint Integration Agent must be installed on a different computer than the one on which HP OM-W is installed, because an API call is required for the integration to accurately track user activity in the system. The call cannot be made from the same local system as OM-W due to security limitations in the API, so the Integration Agent uses a remote WMI call which allows AlarmPoint to associate the correct user with the activity.*

This document assumes that the above products are installed and running prior to installing the integration. To download the latest patches, or for more information, product notes, knowledge base articles, and links to more AlarmPoint products, visit the AlarmPoint Customer Support site at http://connect.alarmpoint.com.

1.3.1 Supported Operating Systems

This integration supports the following operating systems:

- Microsoft Windows 2003 (validated)

1.4 Conventions & Terminology

This section describes how styles are used in this document, and provides a list of definitions.

1.4.1 Conventions

Some instructions appear in the following format: **MENU > OPTION**; for example, **File > Open** means click the **File** menu, and then click the **Open** menu option.

Words in **bold** typically reference text that appears on the screen.

Words in `monospace` font represent the following:

- text that must be typed into the computer
- directory and file names

Directory placeholders

The default AlarmPoint installation folder is `C:\Program Files\AlarmPointSystems\AlarmPoint`; this folder, or its equivalent on your deployment, is referred to throughout this document as `$AP_HOME`.

- For example, the `C:\Program Files\AlarmPointSystems\AlarmPoint\webserver\webapps\cocoon\jsp` folder in a default AlarmPoint installation would be referred to within this document as `$AP_HOME\webserver\webapps\cocoon\jsp`.

The location to which you install the AlarmPoint Integration Agent is referred to as `$APIA_HOME`.

The location on the HP OM-W server containing the AlarmPoint integration components (by default `C:\Program Files\HP\HP BTO Software\install\`) is referred to as `$APOMW`.

1.4.2 Terminology

With respect to the AlarmPoint System, the following definitions apply:

Term	Meaning
AlarmPoint System	Umbrella term for all AlarmPoint software components
AlarmPoint Web User Interface	Browser-accessible interface for controlling AlarmPoint components and information
Management System	A synonym for HP Operation Management software for Windows
Event	Item of interest that typically generates a notification for a recipient
Device	Medium through which a User is contacted (e-mail, phone, BlackBerry, pager, etc.)
Recipient	User or Group to be notified of the event

2. Installation

This chapter provides information about installing the AlarmPoint for HP Operations Manager software for Windows integration.

Note: *Before installing this integration, ensure that you have installed AlarmPoint, the AlarmPoint Developer IDE, the AlarmPoint Integration Agent, and HP OM-W. For instructions, refer to the appropriate user documentation for each product.*

2.1 Installing the Integration

To install the integration, extract the `AP-HP-OM-W.zip` integration archive to your local machine. The following shows the notable files and folders in the archive:

```
.-- components
| |-- alarmpoint
| | |-- scripts
| | | '-- AP-HP-OM-W-v2.21.aps
| | |-- sub_panel
| | | |-- jsp
| | | | |-- OMSearchNode.jsp
| | | | |-- OMSearchNodeGroups.jsp
| | | | |-- OMSearchObject.jsp
| | | | |-- OMSearchPolicy.jsp
| | | | '-- OMWSubscriptionForm.jsp
| | | '-- lib
| | '-- vox
| |-- alarmpoint-integration-agent
| | '-- hpomw
| |   |-- hp_operations_manager_win.xml
| |   |-- hpomw.js
| |   '-- OmwApt.vbs
| '-- hp-omw
|   |-- policies
|   |-- omw.vbs
|   '-- omw-forwarded.vbs
'-- documentation
  '-- AP40-HP-OM-W.pdf
```

2.1.1 Component Description

The following table describes notable integration components and folders:

Component Name	Description
hp_operations_manager_windows.xml	This xml file contains the parameter mapping for messages injected for the hp_operations_manager_win Event Domain., and the Response Action Script (RAS) which handles updating the Management System according to the AlarmPoint response messages injected.

Component Name	Description
omw.vbs	This script enriches the data coming from the OM-W policy to include the events primary node and node groups, along with a textual representation for severity and properly encoded xml characters. It is also responsible for configuring the logging level of AP-OMW-Inject.log, which logs when events are sent from OMW to AlarmPoint. All other enrichment should be performed by the Input Action Scripting to limit the scope of this script. This script handles event injection only and identifies the message by its GUID.
omw-forwarded.vbs	This script contains logic to identify OM-W Events that have been forwarded to AlarmPoint for notification, so that they can be deleted within AlarmPoint when they are Acknowledged in the OM-W console. It is also responsible for configuring the logging level of AP-OMW-Forwarded.log, which logs when del events are sent from OM-W to terminate AlarmPoint events when they are acknowledged from OM-W. This script is called from a WMI Policy that is triggered when OM-W Events have been Acknowledged. Events are identified through their Message GUID.
hp-omw	Contains OM-W tools, policies, test applications, and the Visual Basic application that updates OM-W with AlarmPoint responses.
OmwApt.vbs	This script is responsible for handling the responses generated by Users' Devices within AlarmPoint. It is also responsible for configuring the logging level of AP-OMW-Responses.log, which logs user responses and message delivery annotations from AlarmPoint to OM-W. The script creates a remote WMI connection and uses WMI actions to facilitate the response in OM-W.

2.1.2 Installing Integration Components

This section describes how to install and configure the integration components required by AlarmPoint, the AlarmPoint Integration Agent, and HP OM-W.

2.1.2.1 AlarmPoint Integration Agent

To install and configure the Integration Agent components:

1. Copy the `AP-HP-OM-W\components\alarmpoint-integration-agent\hpomw` folder from the extracted integration archive to `$APIA_HOME\integrationservices\`.
2. Open the `$APIA_HOME\conf\IAConfig.xml` file and add the following line to the `service-configs` section:

```
<path>hpomw/hp_operations_manager_windows.xml</path>
```

3. Open the `$APIA_HOME\integrationservices\hpomw\OmwApt.vbs` file; locate and change the following setting to specify the IP address of the OM-W host machine:

```
# owmHost should be set to the host which is running HP Operations Manager Windows
omwHost = "localhost"
```

2.1.2.2 HP Operations Manager

The integration includes a set of host files for use on OM-W Server computers, located in the `hp-omw` folder in the extracted integration archive.

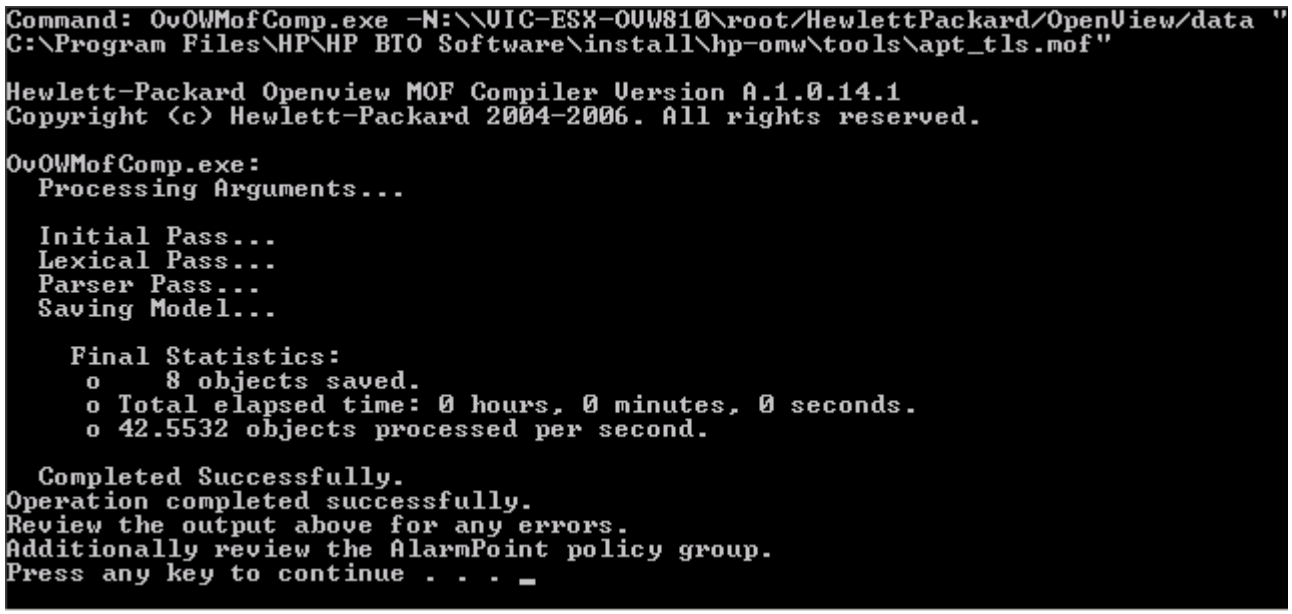
To install and configure the AlarmPoint Host Files:

1. Copy the `AP-HP-OM-W\components\hp-omw` folder from the extracted integration archive to the `C:\Program Files\HP\HP BTO Software\install\` directory on the Management Server.
2. Create an environment variable named APOMW, and set the value to the location of the `hp-omw` directory (e.g., `C:\Program Files\HP\HP BTO Software\install\hp-omw`). Add the environment variable to the "path" environment variable.

Note: *You must use the 8.3 filename convention used by DOS, as it supplies a short name for the folders to eliminate spaces. To determine the 8.3 filename of a folder, type* `dir /x` *in a command line. For example the default hp-omw directory in the 8.3 convention is* `C:\PROGRA~1\HP\HPBTOS~1\install\hp-omw`

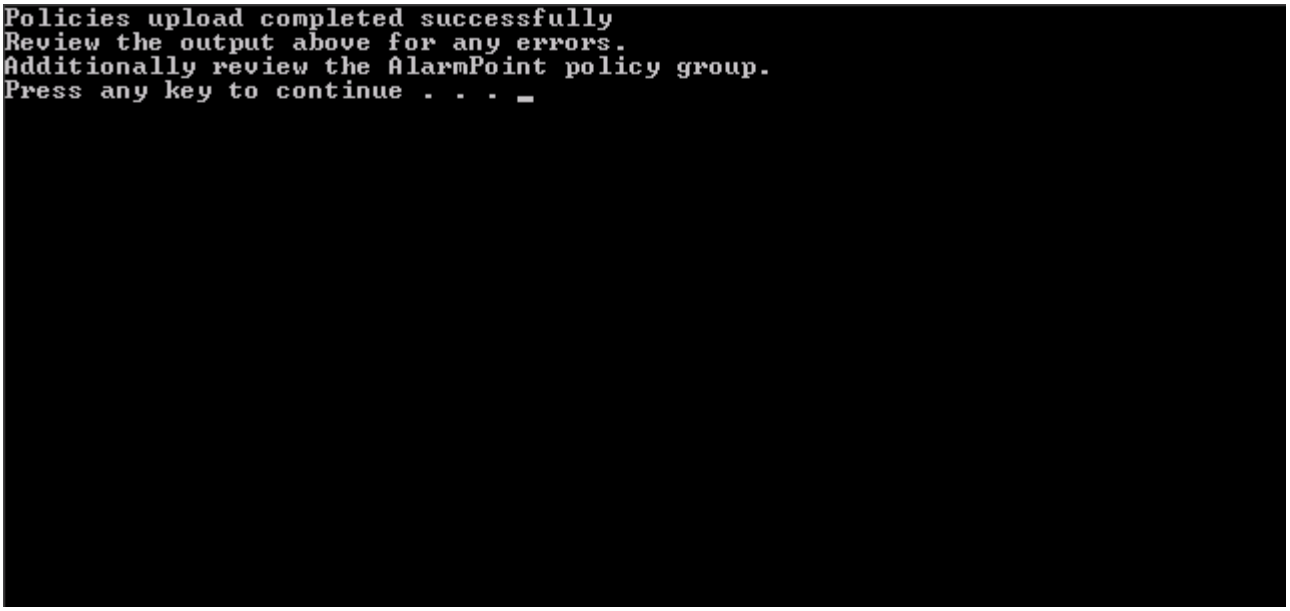
---

3. Navigate to the `C:\Program Files\HP\HP BTO Software\install\hp-omw` and open the `omw.vbs` file in a text editor.
4. Locate the `Const integrationAgentIP = "localhost"` line and change the value within quotes to the IP address of the machine on which you installed the AlarmPoint Integration Agent.
5. Save and close the `omw.vbs` file.
6. Repeat steps 3 to 5 for the `omw-forwarded.vbs` file.
7. From the `$APOMW\tools` directory, execute the `ap_tools.bat` file to import the test tool into the AlarmPoint tools folder within OM-W. (This tool will be used to verify the integration.)
   - A successful execution results in output similar to the following figure:



8. To import the AlarmPoint policies into OM-W, execute the `ap_policies.bat` file in the `$\policies` directory.

- A successful import results in output similar to the following figure:



At this point, all the files necessary for the default integration are in their required locations on the Management server.

3. Configuration

Before using the integration, you must set up an AlarmPoint OM-W Windows User, and configure AlarmPoint and HP Operations Manager for Windows. This section explains the configuration processes required for each component.

3.1 Configuring an AlarmPoint OM-W Windows User

The AlarmPoint OM-W Windows User is used throughout the integration as the default user for WMI interaction to and from HP Operations Manager for Windows. You will need to create this user in Windows on the HP OM-W Server, and then associate the user with the Windows HP OM-W group.

To configure the AlarmPoint OM-W Windows user:

1. In the Windows Control Panel, open **Administrative Tools > Computer Management**.
2. In the Computer Management window, expand System Tools > Local Users and Groups.
3. Right-click Users and select New User.
4. In the New User dialog box, type the following information:
   - **User name**: AlarmPointOMW
   - **Full name**: AlarmPoint Integration User
   - **Description**: This user is required for WMI interaction between AlarmPoint and HP OM-W
   - **Password**: AlarmPointOMW
   - **Confirm password**: AlarmPointOMW

Note: *AlarmPointOMW is the default user name and password used by the integration. If you want to specify a different user name or password, ensure that you use the correct combination when configuring the remaining components in the integration.*

5. Clear the **User must change password at next logon** check box, and then click **Create**.
6. In the Users pane, double-click the new **AlarmPointOMW** user.
7. In the AlarmPointOMW Properties dialog box, click the **Member Of** tab, and then click **Add**.
8. In the Select Groups dialog box, in the **Enter the object names to select** field, type `HP-OVE-ADMINS`, and then click **Check Names**.

Note: *You can also use the HP-OVE-OPERATORS group, but this may limit some actions within the integration; it is recommended that the user be added to the Admins group.*

9. Once the group is fully qualified, click **OK**.
10. Click **Apply**, and then click **OK**.

3.1.1 User References

This user is referenced by the following parts of the integration:

- **AlarmPoint Action Scripts**: for more information, see “Importing the AlarmPoint script package” on page 9.
- **AlarmPoint Subscription Panels**: for more information, see “Configuring the Subscription JSPs” on page 13.

3.2 Configuring AlarmPoint

Configuring AlarmPoint requires the following steps:

- Import the AlarmPoint script package.
- Install the voice files.
- Define an Event Domain.
- Configure the default recipient (set up a target Group which contains a User with a text phone).
- Configure the Subscription Panel and create a fail-safe Group (optional)

3.2.1 Importing the AlarmPoint script package

The integration includes a script package containing example functionality and a starting point for integrating AlarmPoint with OM-W. To use the script package, you must import it into the AlarmPoint Database, and set the message ownership variables. AlarmPoint uses the values in the message ownership variables when processing the responses through the OmwApt.vbs script, which uses WMI actions to update the message in OM-W.

Note: *This step requires the AlarmPoint Developer IDE.*

To import the AlarmPoint Script Package:

1. Launch the AlarmPoint Developer IDE, and configure the database connection.
   - Refer to the AlarmPoint Developer IDE Help or the *AlarmPoint Developer's Guide & Scripting Reference* for details.
2. Click **Workspace > Import**.
3. In the File dialog box, locate the `AP-HP-OM-W\components\alarmpoint\scripts\AP-HP-OM-W-v2.21.aps` file in the extracted integration archive, and then click **Open**.
4. In the Select Company dialog box, select **Default Company**, and then click **OK**.
5. Once the script is imported, click **OK** again.
6. In the left pane of the IDE, expand **Default Company > HP Operations Manager for Windows (Business) > r## 2.2 (PRODUCTION) > PROCESS**.
7. Double-click the **initial** script.

8. In the right pane, locate the following variables and set their values as described in the following table:

Variable	Description
$main.hp_omw_owner_type = "default"	Specifies who should own the OM-W message for Own, Ack, or Change Severity responses. Possible values are: • **default**: sets the owner to value specified by the $main.hp_omw_owner variable, which should be set to a hostname\OM-W user account. • **custom-field**: Sets the owner to the value specified in the AlarmPoint Custom Fields for the responding AlarmPoint User; must be a valid HP OM-W Windows account in the HOSTNAME\USER format. To use the "custom-field" setting, you must create two custom text fields in AlarmPoint named "HP OMW User" and "HP OMW Password" and specify a valid HOSTNAME\USER account and password for each AlarmPoint User that will respond to OM-W notifications. If this variable is set to "custom-field" and either Custom Field is not specified, the "default" setting is used instead. For more information about creating Custom Fields in AlarmPoint, refer to the *AlarmPoint Installation and Administration Guide*.
$main.hp_omw_owner = "hostname\AlarmPointOMW"	Specifies the owner when "default" is set as the owner type; this value must be a valid HP OM-W Windows account in the HOSTNAME\USER format.
$main.hp_omw_password = "AlarmPointOMW"	Specifies the password when "default" is used for the main.hp_omw_owner_type

9. Right-click the **HP Operations Manager for Windows (Business)** folder, and then click **Validate**.
10. Click **Database** > **Check In**.
11. In the Check In dialog box, click **Create**, and then click **Close**.

You can now close the IDE, or use it to modify other variables described in "Configuration Variable Reference" on page 38.

3.2.2 Installing voice files

These files must be installed into an AlarmPoint deployment running a Voice Device Engine. This integration provide only English and English UK voice files.

For more information, refer to the *AlarmPoint Installation and Administration Guide*.

To install the voice files:

Copy all of the files in the `AP-HP-OM-W\components\alarmpoint\vox\<language>` folder from the extracted integration archive to the following node installs folder:

`$AP_HOME\node\phone-engine\Datastore\domains\common\recordings\<language>\phrases`

3.2.3 Defining an Event Domain

By default, this integration is set up to use a default Event Domain, "hp_operations_manager_win"; it is strongly recommended that you use the default Event Domain.

For the integration to be successful, the Event Domain must match the Client ID of the AlarmPoint Integration Agent.

Note: *The AlarmPoint Webserver must be running to perform this portion of the integration.*

To define an Event Domain:

1. Login to AlarmPoint as a Company Administrator, and click the **Developer** tab.
2. On the Event Domains page, click **Add New**.
3. Enter the following information into the form:
   - **Name**: hp_operations_manager_win
   - **Description**: HP Operations Manager for Windows
   - **Script Package**: HP Operations Manager for Windows
4. Click **Save**.
5. On the Event Domains page, click **hp_operations_manager_win**.
6. On the Event Domain details page, in the Integration Services area, click the **Add New** link.
7. On the Integration Service Details page, enter the following information:
   - **Name**: HPOMW
   - **Description**: HP Operations Manager for Windows Integration Service
8. Click **Save**.

3.2.3.1 Defining predicates

After defining the Event Domain and Integration Service, you must define the predicates, or tokens, within the events injected into AlarmPoint.

To define the integration predicates:

1. On the Event Domain details page, beside the Predicates heading, click the Add New link.
2. On the Event Domain Predicates page, type the name of the predicate, indicate its type, and then click **Save**.
   - For list-type predicates, enter the values defined in the table below as necessary.
3. Repeat step 2 for each of the predicates in the following table:

Predicate	Type	Important	Values
NODE	List	Yes	None; automatically populated. (See Note below.)
NODE_GROUPS	List		None; automatically populated. (See Note below.)
NODE_GROUPS_TEXT	Text		
APPLICATION	Text	Yes	
MESSAGE_GROUP	Text		
MSG_OBJECT	List		None; automatically populated. (See Note below.)
MSG_SOURCE	List		None; automatically populated. (See Note below.)
MSG_TEXT	Text		

Predicate	Type	Important	Values
SEVERITY	List	Yes	Manually entered; populate with the following values: • normal • warning • minor • major • critical

4. When you have finished adding all of the predicates, click Save on the Event Domain details page.

Note: *For more information about populating the NODE, NODE_GROUPS, MSG_OBJECT, and MSG_SOURCE predicates, see "Configuring the Subscription JSPs" on page 13.*

3.2.4 Configuring the default recipient

The integration is configured to directly target recipients as defined by the message group of the injected OM-W Event. To configure directly targeted notifications for injected Events, you must create an AlarmPoint Group associated with the message group of the event. When you use the AlarmPoint tools to create an Event, the message group is associated by default with Operations, the default Group in AlarmPoint.

Note: *If a message is sent from OM-W with a Message Group name that does not corresponding to an AlarmPoint Group, the notification will be delivered to the Failsafe Group.*

To configure the default recipient, confirm that the Operations Group exists in AlarmPoint, and that it has a single Team member: the default demonstration User, "bsmith". Follow the steps below to ensure the User exists, is assigned to the Operations Group, and has a virtual text phone Device:

To set up a two-way Device:

1. In the AlarmPoint Web User Interface, click the **Users** tab.
2. On the Find Users page, click **S**.
3. In the list of returned Users, click **Smith, Bob**.
4. On the Details for Bob Smith page, in the Common Tasks pane, click **User Devices**.
5. Verify that a virtual text phone Device exists.
6. Click **Reorder**, and set the virtual phone to be the first Device in the list.
7. Click **Save**.
8. Click the **Details for Bob Smith** link at the top of the page.
9. In the Common Tasks pane, click **Groups User Belongs To**.
10. Confirm that Bob Smith is assigned to the Operations Group.
11. Log out of AlarmPoint.

Note: *If this user is missing, or is not assigned to the Operations Group, create a user with a User ID of "bsmith", assign him to the Operations Group, and add a virtual text phone Device to the User. For more information and instructions on how to perform these tasks, refer to the* AlarmPoint User Guide.

3.2.5 Configure the Subscription Panel

To allow Users to subscribe to specific criteria on injected Events, you must configure the Subscription panel.

To configure the Subscription Panel:

1. Copy all of the files in the `AP-HP-OM-W\components\alarmpoint\sub_panel\jsp` folder from the extracted integration archive to the following folder:

   `$AP_HOME\AlarmPoint\webserver\webapps\cocoon\alarmpoint\jsp\subscription\omw`

2. Copy all of the files in the `AP-HP-OM-W\components\alarmpoint\sub_panel\lib` folder from the extracted integration archive to the following folder:

   `$AP_HOME\AlarmPoint\webserver\webapps\cocoon\WEB-INF\lib`

Note: *After copying the files, you must restart the AlarmPoint Webserver service.*

3.2.5.1 Configuring the Subscription JSPs

This integration is packaged with a custom subscription panel which reads Node and Node Group values from the Operations Manager database. The NODE, NODE_GROUPS, MSG_OBJECT, and MSG_SOURCE predicate values are sourced from Operations Manager using a WMI bridge. Each of the predicates requires a separate JSP that populates its values using search pop-ups.

If you want to populate the predicate value lists from the Operations Manager database using the WMI Bridge, you must configure the connection properties within each JSP file.

To configure the JSPs to connect to the Operations Manager database:

1. Navigate to the following folder on the AlarmPoint server:

   `$AP_HOME\webserver\webapps\cocoon\alarmpoint\jsp\subscription\omw`

2. From within the folder, open the `Configuration.jsp` file.
3. Replace the value within quotes for each parameter as described in the following table:

Parameter	Value
OM_HOST_URL	The appropriate IP address of the HP Operations Manager host machine.
OM_USER_NAME	The user name of a Windows user on the host machine of the HP Operations Manager Server (must be prefaced with `<Machine_Name>\\`). For example, if your machine name is "VIC-ESX-OMW810", and your user is "AlarmPointOMW", you would use the following entry: `final string OM_USER_NAME = "VIC-ESX-OMW810\\AlarmPointOMW";` **Note**: This user should be the same user configured in "Configuring an AlarmPoint OM-W Windows User" on page 8.

Parameter	Value
OM_PASSWORD	Password for the specified Windows user.
JDBC_DRIVER_CLASS_NAME	Class name of the JDBC driver used to retrieve policies from OM-W.
JDBC_URL	URL at which the database can be queried.
JDBC_USERNAME	User name with which to make database queries. (Must be a user with the required database access permissions.)
JDBC_PASSWORD	Password for the querying user.

4. Save and close the JSP.

Manually populating predicate lists

You can also allow Administrators to change the source of the content supplied by the search pop-ups from database queries to predefined predicate value lists. To configure the subscription panel in a demo mode, using pre-defined predicate list values, you must modify the search pop-up JSP files.

To manually populate the predicate list:

1. Open the `OMSearchNode.jsp`, `OMSearchNodeGroups.jsp` , `OMSearchPolicy.jsp`, and `OMSearchObject.jsp` files found in the following folder:

   ```
   $AP_HOME\webserver\webapps\cocoon\alarmpoint\jsp\subscription\omw
   ```

2. For each JSP, locate the boolean variable USE_PREDEFINED_LIST_VALUES, and set its value to `true`.
3. Save and close the files.
4. Login to AlarmPoint as a Company Administrator, and click the **Developer** tab.
5. On the Event Domains page, click the **hp_operations_manager_win** domain.
6. On the Event Domain Details page, click **NODE** in the Predicates list.
7. Add to the predicate list values.
8. Repeat steps 6 and 7 for **NODE_GROUPS**, **MSG_OBJECT**, and **MSG_SOURCE** in the Predicates list.

While configuring a Subscription when a search is done on either predicate, the search results will now result in the predefined list values instead of the database query results.

Note: *Changing Subscriptions by adding or removing Event Domain predicates may cause existing Subscriptions to fail. For more information about working with Event and Subscription Domains, see the* AlarmPoint Installation and Administration Guide.

3.2.5.2 Configuring the Event and Subscription Domains

After you have configured the Subscription Panels, you must configure the Event Domain with the relevant predicates, and create a Subscription Domain to use as a base when creating Subscriptions.

1. Log in to AlarmPoint as an Adminstrator and click the **Developer** tab.
2. On the Event Domains page, click the **hp_operations_manager_win** Event Domain.
3. Ensure that the predicates described in “Defining predicates” on page 11 have been added to the Event Domain.
4. In the menu on the left side of the screen, click **Subscription Domains**.

5. On the Subscription Domains page, click **Add New**, and then select **hp_operations_manager_win** in the **Event Domain** drop-down list and click **Continue**.
6. On the Subscription Domain details page, create a Subscription Domain named **OMW** that includes the above predicates, and enter the following path for the **Custom Panel URL**:

```
jsp/subscription/omw/OMWSubscriptionForm.jsp
```

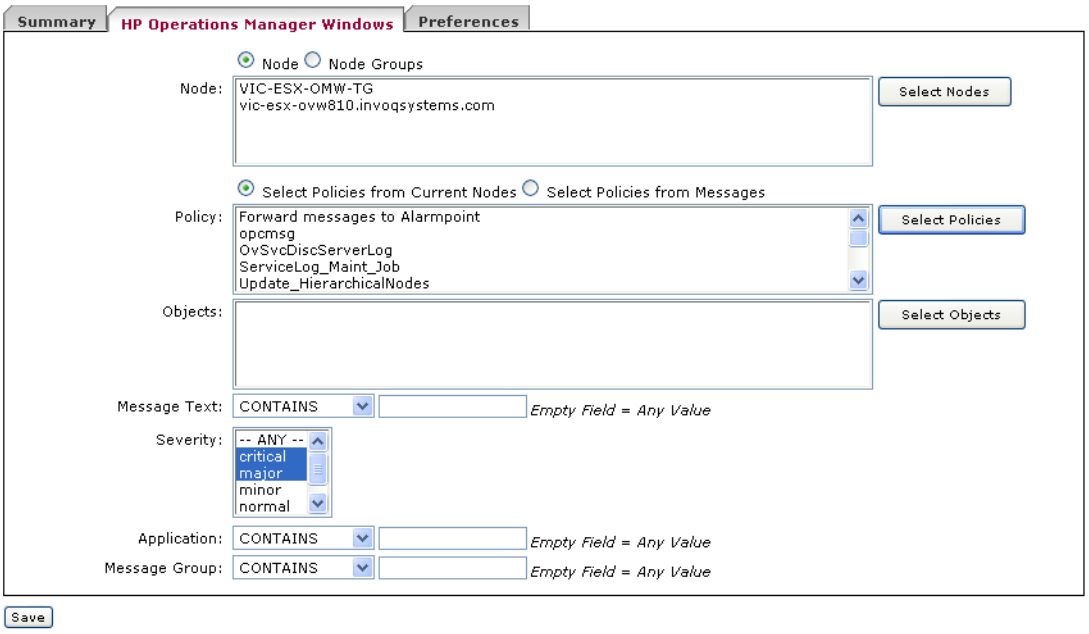
   - For the **Subscription Type**, select **Both**. This will allow you to assign the Subscription to Users, and to create a self-managed Subscription.
7. Click **Continue**.
8. If Users should have the ability to respond to Subscription notifications, add the following response choices:
   - **Acknowledge**
   - **Own**
   - **Ignore**
   - **Annotate**
9. Select all of the available predicates, click **Add**, and then click **Continue**.
10. Select the user roles that will be able to subscribe to omw events, then click add, and then save.

3.2.6 Creating a Subscription

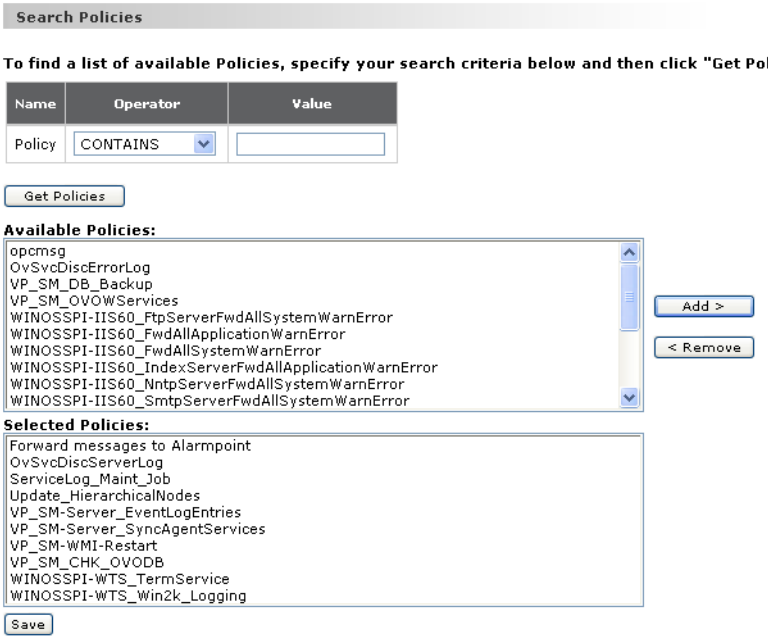
You can now use the Custom Subscription Panel to subscribe to OM-W Events of specific severities, or that match any specified criteria.

To create a Subscription:

1. On the Alerts tab of the AlarmPoint Web User Interface, click **My Subscribed Alerts**.
2. Click the **Add New** link above the Self-made Subscriptions table.
3. On the Select a Subscription Domain page, in the **Subscription type** drop-down list, select **OMW**, and then click **Continue**.
4. On the Subscription Details page, specify a name for the Subscription, and set the Subscription criteria using the HP Operations Manager Windows and Preferences tabs.
   - The HP Operations Manager Windows tab (Ctrl-click to select more than one item in a list):



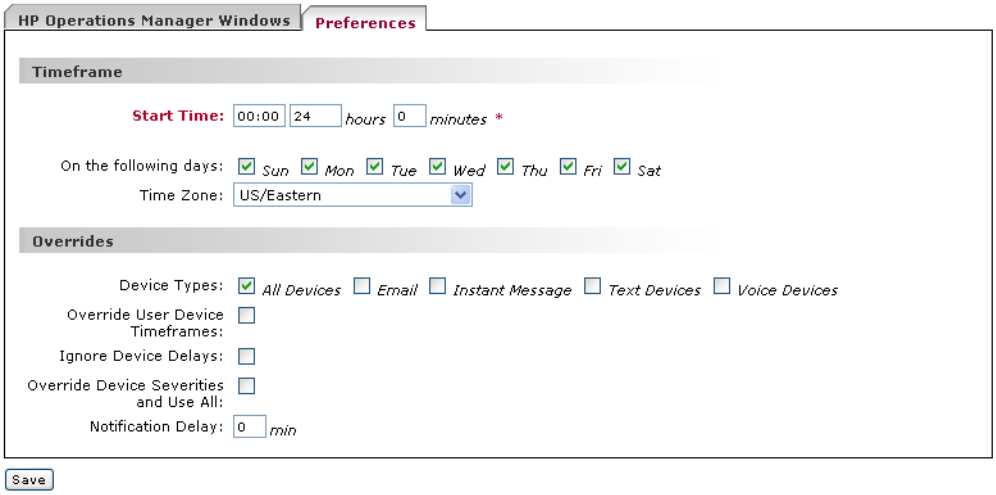
- To populate the list of available Nodes, Policies, or Objects, click the **Select** button beside the element you want to populate; the Search page opens and displays the search criteria:



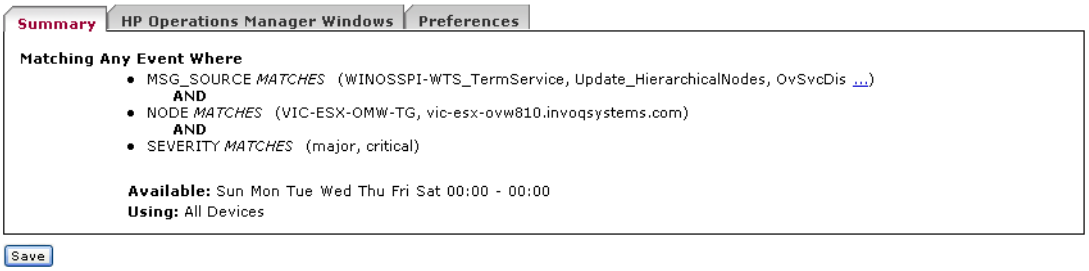
Note: *Do not attempt to open more than one Search page at a time; an "Access Denied Error" can occur if more than one pop-up window is loaded before the window is finished loading. If the error occurs, you must restart the AlarmPoint Webserver.*

- On the Search page, use the search criteria to locate the elements and click **Add** to add them to the Selected list. Click **Save** to add the list of selected items to the list on the tab and close the Search page.

- The Preferences tab:



5. When you are satisfied with the criteria, click **Save** to create the Subscription.
   - You can review the Subscription details at any time on the Summary tab:



3.2.6.1 Testing the Subscription Panel

To test the Subscription Panel, create a User in AlarmPoint and assign them a Subscription that matches the parameters injected from the Operations Manager when executing the Test Tools. You can use the following criteria to match against:

- **Application**: OMWAlarmPoint
- **Message_Group**: Operations
- **Severity**: Critical

If the parameters of an injected message match the Subscription criteria, AlarmPoint sends a Subscription notification to the configured User.

Note: *Ensure that Subscriptions are enabled within the Action Script Package. For more information, see "FYI and Subscription Notification Variables" on page 38.*

3.2.6.2 Create a Fail-Safe Group

If a notification is submitted to AlarmPoint when the fail-safe functionality is enabled, and if it matches the necessary circumstances, AlarmPoint sends the notification to the fail-safe recipient. The fail-safe recipient is typically a Group, but can be configured as a User.

To create a fail-safe Group:

1. Login to AlarmPoint as a Company Administrator, and click the **Groups** tab.
2. Create a new Group named **OMW FailSafe**, with at least one User as a Team member to receive notifications.

For more information about creating Groups and Teams, see the AlarmPoint User Guide.

Note: *If you want to use a pre-existing group or a different group name, modify the value for the* `$fail_safe_group` *variable defined in the Initial Business script in the AlarmPoint Action Scripts.*

3.3 Configuring HP Operations Manager for Windows

This section explains the configuration procedures for HP Operations Manager software for Windows required for this integration.

3.3.1 OM-W Integration Module Distribution Components

The integration module components are distributed as a compressed archive with the following component structure:

Table 3-1. OM-W Integration Module Distribution Components

Component	Description
Environment Variables	As described in the Installation chapter, the APOMW environment variable is required, and must default to the location of the `hp-omw` directory (e.g., `C:\Program Files\HP\HP BTO Software\install\hp-omw`). Additionally, the AlarmPoint Integration Agent must be installed on a machine other than the OM-W server.
ap_policies.bat	This script is located in the `$APOMW\policies` directory, and is used to import the default AlarmPoint policies to activate the integration within OM-W.
ap_tools.bat	This script is located in the `$APOMW\tools` directory, and is used to import the AlarmPoint test tool into the AlarmPoint tools folder.
apt_tls.mof	This file contains the test tool that will be imported into OM-W. It was created by OM-W when the tool was exported for distribution, and will be accessed by the ap_tools.bat file during the tool import process.
OmwApt.vbs	This script is located in the `$APIA_HOME\integrationservices\hpomw` directory, and contains the logic to handle response injections from AlarmPoint to OM-W. It processes messages from AlarmPoint to OM-W, using the same GUID in support of the two-way integration.

3.4 OM-W Policies and Tools

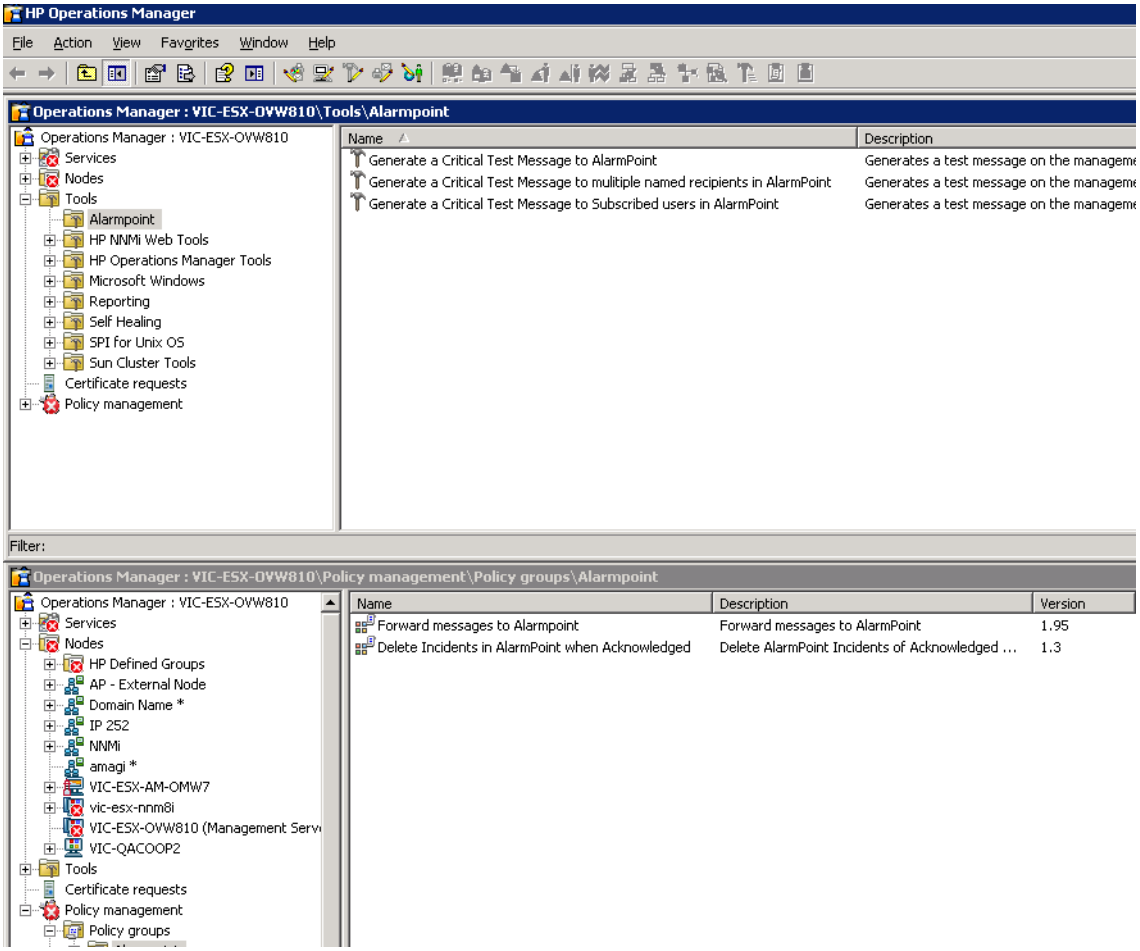
When creating, updating or maintaining OM-W Policies, you may need to add or remove tokens that are being passed to AlarmPoint. When doing so, you must duplicate your changes to the tokens being passed by the policy in the following files:

- **`omw.vbs`**: Any changes to the tokens being passed out of OM-W must be reflected here, specifically in the `ProcessArguments()` and `injectEvent()` subroutines.
- **`hp_operations_manager_win.xml`**: The data map is defined in the `<mapped-input method="add" subclass="action">` and must reflect the changes made to the policy and the `omw.vbs` file.

Note: *It is important to consider all policies that will be using this integration when you make your changes; be careful not to remove tokens used by other policies and ensure that you provide default values for tokens that are not supplied by every policy. For an example of adding tokens, see "Adding Custom Data Elements" on page 29.*

Table 3-2. OM-W Policies and Tools

Policy/Tool Name	Policy Type	Tool Type	Description
Forward Messages to AlarmPoint	WMI		Forwards messages to AlarmPoint from OM-W.
Delete Incidents in AlarmPoint	WMI		Forwards a "del" event to AlarmPoint when an Operations Event is acknowledged in the Console.
Generate a Critical Test Message to AlarmPoint		WMI	Generates a critical test message, triggering an alert via AlarmPoint for a single User/Group; runs on the Management Server only; can be associated with a service and/or Node. The Forward Messages to AlarmPoint Policy must be deployed on the Management Server before using this command.
Generate a Critical Test Message to mulitiple named recipients in AlarmPoint		WMI	Generates a critical test message, triggering an alert via AlarmPoint for a multiple Users/ Groups; runs on the Management Server only; can be associated with a service and/or Node. The Forward Messages to AlarmPoint Policy must be deployed on the Management Server before using this command.
Generate a Critical Test Message to Subscribed users in AlarmPoint		WMI	Generates a critical test message triggering an alert via AlarmPoint for a subscribed User; runs on the Management Server only; can be associated with a service and/or Node. The Forward Messages to AlarmPoint Policy must be deployed on the Management Server before using this command.



3.4.1 Policy Notification Activation

Once the scripts have been placed in their appropriate directories, and the default policies have been imported into OM-W, activate the integration by deploying the policy to the OM-W Server:

1. From within OM-W's AlarmPoint policy group (**Operations Manager > Policy Management > Policy Groups > AlarmPoint**), right-click the **Forward messages to AlarmPoint** policy and select **All tasks > deploy on**.
2. Expand the **Nodes** tree and select only the check box for your server, **SERVER (Management Server)**.
3. Click **OK** to deploy the policy.
4. Repeat the above steps for the Delete Incidents in AlarmPoint policy.

The **Forward Messages to AlarmPoint** Policy may be customized by adding additional rules. Each rule can call a separate `omw.vbs` script to handle and enrich specific parameters.

3.4.2 Forward Messages to AlarmPoint Policy

This policy provides the logic to determine which events warrant the generation of an alert via AlarmPoint notification processing. Any message may be used to generate a notification by simply adding a rule to this policy, and modifying the Automatic command with the appropriate parameters. This policy passes the entire TargetInstance ID of the current message to the `omw.vbs` script so that it may process the components into an appropriate AlarmPoint Integration Agent call.

The example policy forwards all CRITICAL messages to AlarmPoint for processing. After eliminating any messages generated by the integration itself, the policy checks for messages with a CRITICAL Severity (a value of 32). The automatic

action calls the integration script, injecting a list of parameters defined within the Rule for the Policy. This injected message will be enriched by omw.vbs, which links the Node and Node Groups associated with the message and injects it to AlarmPoint for Notification.

This policy is intended only as an example of how to forward messages to AlarmPoint. Using this policy as a guide, you must create policies that match your specific business requirements for event notification.

3.4.3 Delete Incidents in AlarmPoint Policy

When an Event is acknowledged within the Operations Console, it is considered closed and no further actions are allowed other than to re-open the Event. The "Delete Incidents in AlarmPoint" Policy forwards a "del" Event to AlarmPoint, closing the associated AlarmPoint Event. This prevents the AlarmPoint User from taking actions on a closed Event.

3.4.4 Debugging Visual Basic Scripts

This integration incorporates three Visual Basic scripts used for passing messages to and from OM-W to the AlarmPoint Java Client. To aid in troubleshooting the intermediate VB scripts, the scripts include the logType flag.

You can specify the amount of logging performed on the Visual Basic scripts by setting the debugLogLevel variable for each of the `forward.vbs`, `omw.vbs`, and `OmwApt.vbs` files. The debugLogLevel can be set to one of the following values:

Value	Description
false (default)	Shows critical logging only; excludes INFO logging
true	Shows all messages, including INFO logging. (Note that INFO messages contain a lot of data, and can cause large log files.)

The log files for each of the Visual Basic scripts are located as follows:

Visual Basic Script	Log File
`forward.vbs`	`$APOMW\logs\AP-OMW-Forwarded.log`
`omw.vbs`	`$APOMW\logs\AP-OMW-Inject.log`
`OmwApt.vbs`	`$APIA_HOME\logs\AP-OMW-Response.log`

By default, debugging information is written to a file. You can also choose to annotate the debug information to the original OM-W Event, or disable the logging behavior entirely. To log a debug message, you must set the message to write in the logText variable, specify the type of logging behavior in the logType variable, and then call the WriteLog method.

The logType flag may specify one of the following behaviours:

Value	Description
toFile	Logs the debug message to a file.
toEvent	Annotates the debug message to the OM-W Event.
toBoth	Logs the debug message to a file, and annotates the message to the OM-W Event.

If any other value is supplied, the logging behavior is disabled.

Note: *The messageGUID must be associated with a valid OM-W Event for the annotating to work.*

3.5 Software Component Validation

It is recommended that you start the applications in the following order:

1. HP Operations Manager Server
2. AlarmPoint Application and Notification Servers
3. AlarmPoint Integration Agent

Consult the respective user manuals for details on starting these applications.

3.5.1 AlarmPoint Integration Agent

Verify that the AlarmPoint Integration Agent and AlarmPoint Server are connected and running. In `$APIA_HOME`, execute the following command:

```
iadmin.bat get-status
```

If the systems are configured properly, the output will be similar to the following figure:

```
amagi@vic-amagi /cygdrive/c/Program Files (x86)/AlarmPointSystems/IntegrationAgent/bin
$ ./iadmin.bat get-status
Version: 4.0.0 build.158 r31229
Release date: Nov 18, 2008
Agent started: Feb 4, 2009 9:08:42 AM
Agent ID: vic-amagi/192.168.168.60:8081

Integration Services:
        Event domain: default
                Name: sample
                URL: http://localhost:8081/default_sample
                Started: Feb 4, 2009 9:08:43 AM
                Last request: none
                Status: ACTIVE
                Pending request count: 0
                Normal priority inbound APXML queue size: 0
                High priority inbound APXML queue size: 0
                Normal priority outbound APXML queue size: 0
                High priority outbound APXML queue size: 0


        Event domain: del
                Name: del
                URL: http://localhost:8081/del_del
                Started: Feb 4, 2009 9:08:43 AM
                Last request: Feb 4, 2009 1:05:20 PM
                Status: ACTIVE
                Pending request count: 0
                Normal priority inbound APXML queue size: 0
                High priority inbound APXML queue size: 0
                Normal priority outbound APXML queue size: 0
                High priority outbound APXML queue size: 0


        Event domain: hp_operations_manager_win
                Name: HPOMW
                URL: http://localhost:8081/hp_operations_manager_win_HPOMW
                Started: Feb 4, 2009 9:08:43 AM
                Last request: Feb 4, 2009 11:39:25 AM
                Status: ACTIVE
                Pending request count: 0
                Normal priority inbound APXML queue size: 0
                High priority inbound APXML queue size: 0
                Normal priority outbound APXML queue size: 0
                High priority outbound APXML queue size: 0
```

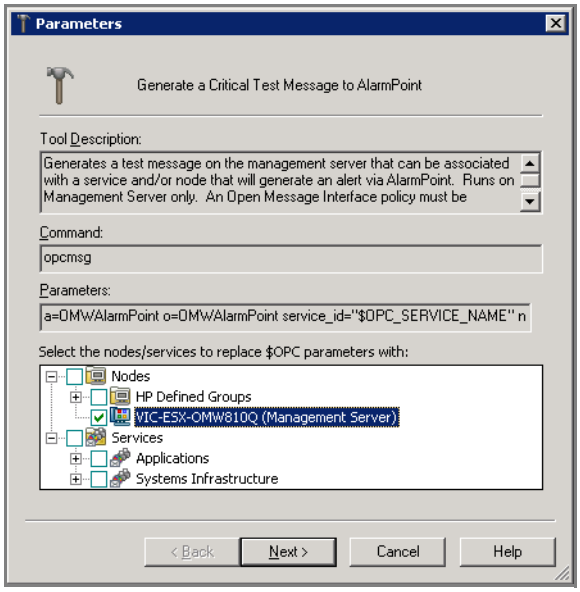
4. Software Component Integration

This section describes how to trigger a test notification and respond to it using a virtual BlackBerry Device. (You can respond to the test notification using any of the AlarmPoint Virtual Devices.)

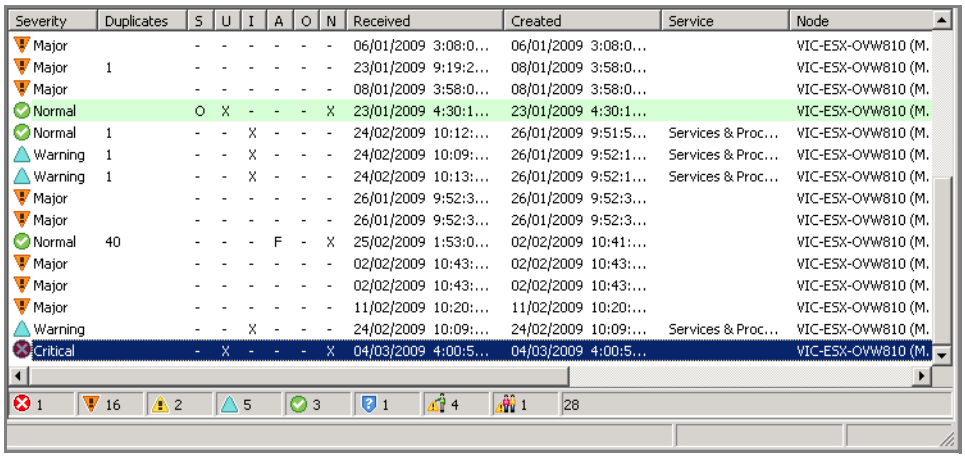
4.1 Trigger a Notification

In this example, the Test Tool included with this integration is used to trigger a CRITICAL event in OM-W. The AlarmPoint policy detects that a CRITICAL event has occurred and sends notifications to AlarmPoint. Follow the steps below to send a CRITICAL event:

1. Open the Operations Manager Console and display the Test Tool by expanding the tree in the left pane in the **Operations Manager** to **Tools > AlarmPoint**.
2. Double-click **Generate Critical Test Message to AlarmPoint**.
   - The Edit Login and Parameters dialog box opens:



3. Select the <**SERVER_NAME> (Management Server)** check box.
4. Click **Next**, and then click **Launch**.
   - The Nodes area displays the CRITICAL event:



Severity	Duplicates	S	U	I	A	O	N	Received	Created	Service	Node
Major		-	-	-	-	-	-	06/01/2009 3:08:0...	06/01/2009 3:08:0...		VIC-ESX-OVW810 (M.
Major	1	-	-	-	-	-	-	23/01/2009 9:19:2...	08/01/2009 3:58:0...		VIC-ESX-OVW810 (M.
Major		-	-	-	-	-	-	08/01/2009 3:58:0...	08/01/2009 3:58:0...		VIC-ESX-OVW810 (M.
Normal		O	X	-	-	-	X	23/01/2009 4:30:1...	23/01/2009 4:30:1...		VIC-ESX-OVW810 (M.
Normal	1	-	-	X	-	-	-	24/02/2009 10:12:...	26/01/2009 9:51:5...	Services & Proc...	VIC-ESX-OVW810 (M.
Warning	1	-	-	X	-	-	-	24/02/2009 10:09:...	26/01/2009 9:52:1...	Services & Proc...	VIC-ESX-OVW810 (M.
Warning	1	-	-	X	-	-	-	24/02/2009 10:13:...	26/01/2009 9:52:1...	Services & Proc...	VIC-ESX-OVW810 (M.
Major		-	-	-	-	-	-	26/01/2009 9:52:3...	26/01/2009 9:52:3...		VIC-ESX-OVW810 (M.
Major		-	-	-	-	-	-	26/01/2009 9:52:3...	26/01/2009 9:52:3...		VIC-ESX-OVW810 (M.
Normal	40	-	-	-	F	-	X	25/02/2009 1:53:0...	02/02/2009 10:41:...		VIC-ESX-OVW810 (M.
Major		-	-	-	-	-	-	02/02/2009 10:43:...	02/02/2009 10:43:...		VIC-ESX-OVW810 (M.
Major		-	-	-	-	-	-	02/02/2009 10:43:...	02/02/2009 10:43:...		VIC-ESX-OVW810 (M.
Major		-	-	-	-	-	-	11/02/2009 10:20:...	11/02/2009 10:20:...		VIC-ESX-OVW810 (M.
Warning		-	-	X	-	-	-	24/02/2009 10:09:...	24/02/2009 10:09:...	Services & Proc...	VIC-ESX-OVW810 (M.
Critical		-	X	-	-	-	X	04/03/2009 4:00:5...	04/03/2009 4:00:5...		VIC-ESX-OVW810 (M.

1 16 2 5 3 1 4 1 28

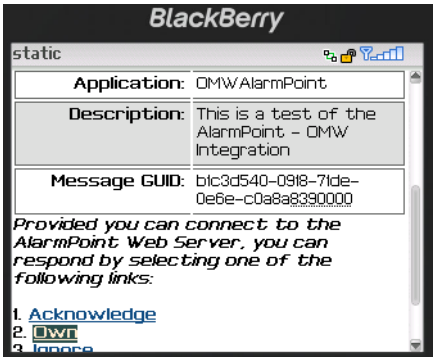
By default, a notification is sent to AlarmPoint's default Group (Operations).

4.2 Responding to a Notification

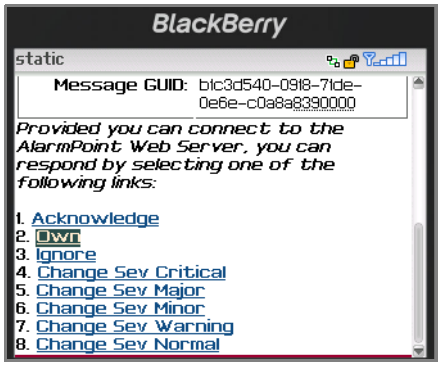
This section describes how to respond to a page using a BlackBerry:

1. When a notification arrives for the default user, the BlackBerry indicates the number of calls that have been received:

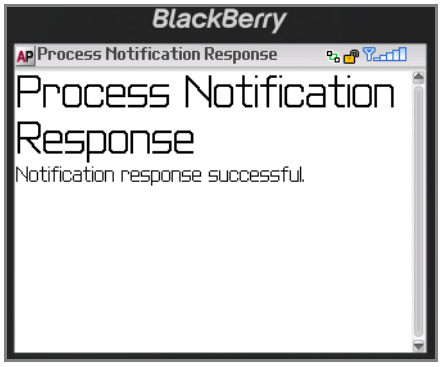
2. Open the notification to display its details:



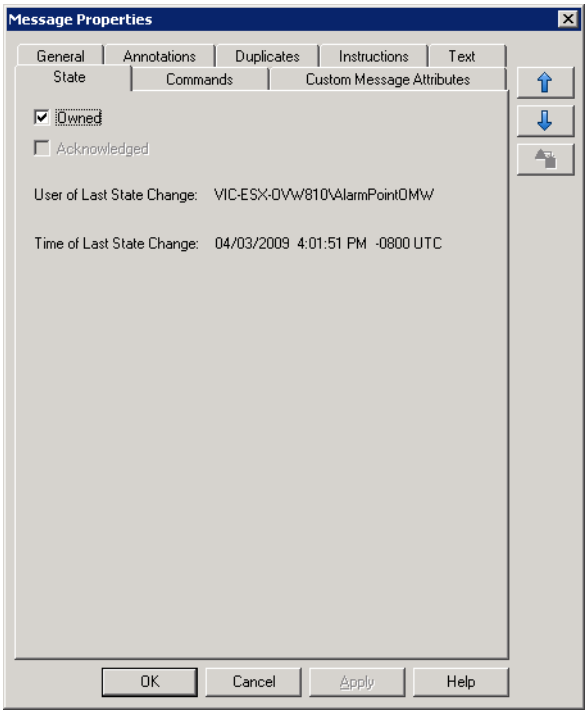
3. The list of possible response choices is available at the bottom of the notification:



4. Click **Own**, and the BlackBerry will send the response to AlarmPoint:



5. AlarmPoint then sends the **Own** response to OM-W, which updates the Message State:

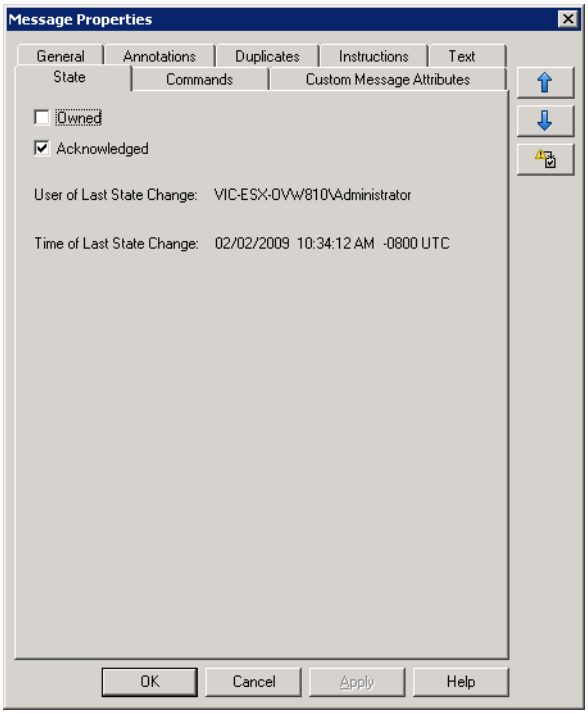


6. If the BlackBerry accesses the notification details again and responds with **Acknowledge**, the Message State is updated again, and the message is cleared from the Message Pane.

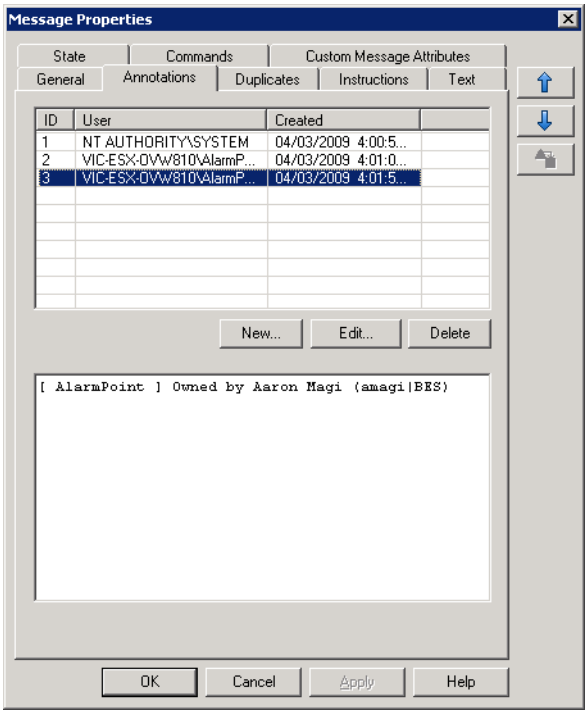
4.3 View Request Results

In the Operations Manager Console, the message will be cleared from the message pane once it has been acknowledged.

1. To view, right-click **Nodes**, and then select **View** > **Acknowledged Messages**.
2. Select the Message from the list, and in the Message Properties dialog box, click the **State** tab:



3. Click the **Annotations** tab to view the note added by AlarmPoint:



5. Optimizing and Extending the Integration

This section describes some of the available methods you can use to optimize or extend the AlarmPoint for HP Operations Manager for Windows Integration.

5.1 Adding Custom Data Elements

Additional data elements, or tokens, can be forwarded to AlarmPoint by adding them to the policy in OM-W or retrieving them via WMI in the `omw.vbs` enrichment and injection script. The following steps explain how to add a custom data element to inject the OriginalText for the OV_Message to AlarmPoint and display within the notification content.

5.1.1 Adding custom parameters to the AlarmPoint policy

If you have already configured a custom policy for your integration to forward messages to AlarmPoint, edit your custom policy rather than the Forward Messages to AlarmPoint policy.

To add a data element to the policy for forwarding messages to AlarmPoint:

1. Open the Operations Manager Console and under Policy Management, click **Policy Group** > **AlarmPoint**.
2. Open the **Forward Messages to AlarmPoint** policy for editing.
3. Within the Rules section, select the rule to which you want to add the custom data element and click **Modify**.
4. On the Actions Tab of the Rule pop-up, click **Automatic command**.
5. Add the data element to the end of the command line.
   - For example, `"<$WBEM:TargetInstance.OriginalText>"`
6. Click **OK**, and then click **OK** again.
7. On the Rules tab of the policy, under Rule Summary, locate the **Start Automatic Command** block, and verify that the new parameter has been added.

The following is a representation of the Start Automatic Command block with the added custom data element. The custom parameter is the original text of the Event, "<$WBEM:TargetInstance.OriginalText>":

```
Start Automatic command (cmd /c cscript.exe /NoLogo %APOMW%omw.vbs
"hp_operations_manager_win" "OPERATIONS MANAGER EVENT" "no"
"<$WBEM:TargetInstance.Severity>" "<$WBEM:TargetInstance.Id>"
"<$WBEM:TargetInstance.MessageGroup>" "<$WBEM:TargetInstance.NodeName>"
"<$WBEM:TargetInstance.Application>" "<$WBEM:TargetInstance.Object>"
"<$WBEM:TargetInstance.TimeCreated>" "<$WBEM:TargetInstance.TimeReceived>"
"<$WBEM:TargetInstance.ServiceId>" "<$WBEM:TargetInstance.Source>"
"<$WBEM:TargetInstance.OriginalText>")
```

Within the `omw.vbs` enrichment and injection script, this custom parameter must be retrieved from the injected message received from OM-W.

To retrieve the custom parameter from the injected message:

1. Determine an appropriate name for the parameter and instantiate it within the Globals section at the beginning of the script.
   - For example, `Dim apdt_original_text`
2. To retrieve this parameter add a line to the ProcessArguments Sub method:

```
apdt_original_text = Trim( args.Item( 13 ) )
```

Note: *The* `args.item(#)` *must match the location of the custom parameter within the Start Automatic Command block of the policy. This number starts at 0, where 0 matches* `"hp_operations_manager_win"`*, the first parameter after the executed application name (*`%APOMW%\omw.vbs`*).*

Alternately, you can retrieve custom parameters through WMI Objects within the `omw.vbs` script.

To retrieve a parameter using WMI:

1. Determine if the parameter is associated with the OV_Message or OV_ManagedNode Objects, and the name of the desired field
   - The WMI Object Browser is useful in making this determination.
2. Within the retrieveNodeAndNodeGroups Sub method, after the Node or Message Object has been retrieved, set the custom parameter to the value of the associated field:

```
apdt_original_text = HPWMIMessageObject.OriginalText
```

   - OriginalText is an example field, and is the same as the parameter injected through the AlarmPoint policy in OM-W.

5.1.2 Injecting custom parameters to AlarmPoint

Once you have retrieved the custom parameter in omw.vbs, you can inject it into AlarmPoint.

To inject custom parameters to AlarmPoint:

1. Add a line to the MapData section of the injectEvent Sub method which escapes and appends the custom parameter to the MapData parameter:

```
MapData = MapData & "&" & MapDataURI & Escape( apdt_original_text )
```

   - The custom parameter will now be injected to the integration file (`hp_operations_manager_win.xml`). The DataMap within the file must be updated to pass the custom parameter on to the AlarmPoint Server.
2. Add a line to the mapped input section, similar to the following:

```
<parameter index="21" type="string">original_text</parameter>
```

   - This parameter will now be injected to AlarmPoint and can be added to the Action Scripts as content for Devices (Email, Pagers, etc.).

5.1.3 Adding the custom parameter to notification content

Once you have injected the custom parameters, you can add the parameter to the notification content for Devices. The following steps explain how to add the custom parameter to email notifications; adding content for other Device types is similar and requires the presentation script to be modified for the specific Devices.

To add custom parameters to email notification content:

1. Open the AlarmPoint Developers IDE and checkout the HP Operations Manager for Windows (BUSINESS) Script Package.
2. In the Presentation Action Script, add a line to the email content creation section:

```
$content.message = $content.message & "Original Text:      " & $event.original_text
& "\n"
```

3. You can also add a check in the Initial script to confirm that the custom parameter was injected properly and therefore exists within the Action Scripts:

```
IF ( ! EXISTS( $event.original_text ) )
  $event.original_text = $undefined_default
  IF ( $main.debug )

    @script::log( $main.log_prepend & "Optional token 'original_text' not found,
defaulting to '" & $event.original_text & "'" )
  ENDIF
ENDIF
```

Your Custom Parameter should now be injected to AlarmPoint with every OM-W message and appear in your Notification content for Email.

Note: *This example affects only the content of normal notifications. To display this content in Subscription Notifications, update the subscriptionPresentation script accordingly.*

5.2 Policy Customizations

Each time you want to add additional conditions for notification, you must add an additional rule to the integration's **Forward messages to AlarmPoint** Policy.

5.2.1 FYI Notifications

FYI notifications are informational only; they are delivered to recipients with no expectation that the recipients will act upon the notification. Regular, or non-FYI, notifications are delivered to recipients with an expectation that the recipients will act upon the event.

To generate FYI notifications from AlarmPoint, the policy must be specifically changed to instruct the integration to deliver the notifications as FYI. In the out-of-the-box integration, the third parameter injected to the `omw.vbs` script is used to indicate whether the policy is generating FYI or non-FYI notifications.

Non-FYI:

```
cmd /c cscript.exe /NoLogo "%APOMW%\omw.vbs" "openview_operations" "OPERATIONS
MANAGER EVENT" "no" "<$WBEM:TargetInstance.Severity>" "<$WBEM:TargetInstance.Id>"
"<$WBEM:TargetInstance.MessageGroup>" "<$WBEM:TargetInstance.NodeName>"
"<$WBEM:TargetInstance.Application>" "<$WBEM:TargetInstance.Object>"
"<$WBEM:TargetInstance.TimeCreated>" "<$WBEM:TargetInstance.TimeReceived>"
"<$WBEM:TargetInstance.ServiceId>" "<$WBEM:TargetInstance.Source>"
```

FYI:

```
cmd /c cscript.exe /NoLogo "%APOMW%\omw.vbs" "openview_operations" "OPERATIONS
MANAGER EVENT" "yes" "<$WBEM:TargetInstance.Severity>" "<$WBEM:TargetInstance.Id>"
"<$WBEM:TargetInstance.MessageGroup>" "<$WBEM:TargetInstance.NodeName>"
"<$WBEM:TargetInstance.Application>" "<$WBEM:TargetInstance.Object>"
"<$WBEM:TargetInstance.TimeCreated>" "<$WBEM:TargetInstance.TimeReceived>"
"<$WBEM:TargetInstance.ServiceId>" "<$WBEM:TargetInstance.Source>"
```

Note: *Do not submit all OM-W Events for notification. Tailor the policies to fit your specific business requirements and the capacity of your OM-W Server, AlarmPoint Server and communications infrastructure.*

5.3 Response Choices

The AlarmPoint Integration allows recipients to respond to notifications with several default choices, some of which are injected back to the Operations Manager server, updating the original event. Users notified on email devices also have the ability to respond with an extra annotation message which will be logged in the original Operations Manager event.

The following is a list of the default response choices available with the AlarmPoint Integration and their associated actions on the AlarmPoint event and the Operations Manager event:

Table 5-1. Default Response Choices

Response Choice	AlarmPoint Action	Operations Manager Event Update	Default Device Availability
Acknowledge	Delinks everyone from the AlarmPoint event (deletes/ terminates the event), and halts notifications from being delivered. **Note**: If FYI Subscriptions are being delivered, they are allowed to finish.	Removes the message for the active messages browser view and puts it in the acknowledged message browser.	Email, BES, Browser. For other non-FYI mobile devices an Acknowledge is represented as an Ack.
Own	Delinks all users other than the responder from the event, not allowing them to submit responses. The owner will not be notified further, but has the ability to affect the event by responding on one of their Devices or from the browser. For example, a User owns the event in AlarmPoint, and then changes the severity of the event. They may also acknowledge or annotate the owned event.	The owner gains exclusive read/ write access to the message. Other users can see the message, but have limited access.	All non-FYI devices.
Ignore	Signifies that the User rejects the notification. The rejection causes the action script to escalate to the next recipient in the Group.	A reject message is sent back to OM-W and logged as an Annotation; it has no effect on the state of the OM-W message.	Email, BES and Browser. For other non-FYI mobile Devices an Ignore is represented as an Ign.

Table 5-1. Default Response Choices

Response Choice	AlarmPoint Action	Operations Manager Event Update	Default Device Availability
Change Severity	Halts delivery of notifications to any other Devices the responding User may have configured. Delinks all Users other than the User changing the severity.	Handles the changing of the message severity. Possible severities: • Critical • Major • Minor • Warning • Normal Owns the OM-W message as though the severtity was changed in the OM-W console.	Email, BES and Browser. If notified on a phone Device, the option to change severity is provided as a phone menu. (Can only change the severity up or down, but have unlimited number of times to do so; i.e., it requires four times to go from critical to warning.)
Annotate	Halts delivery of notifications to any other Devices the responding User may have configured.	Allows the User to provide a message to be posted to the annotation of the message. When an annotation is provided the state of the OM-W message does not change.	This functionality is available for text-based Devices only.

5.3.1 Adding Annotation Messages

Two-way email Device notifications (not FYI) can add extra annotations which will be added to the Operations Manager Event as an annotation on the Annotations tab. To add an extra annotation, respond to an email notification with the following format in the subject line:

```
RESPONSE <Choice> <Message>
```

<Choice> can be any of the response choices listed in the table above, and <Message> can be any content you want to add as the annotation.

Note: *FYI Notifications are for informational purposes only, and do not allow for a response to be injected. For Voice FYI Notifications, the options are to delete, save or repeat the notification. These options delete the notifications from the phone queue, save it, or repeat the message; they have no effect on the OM-W Event.*

5.3.2 Changing and Adding Response Choices

You can change the response choices available within the presentation script (Subscription Responses are configured while setting up the Subscription Domain; consult the *AlarmPoint Installation and Administration Guide* for more information). The behavior of responses can be configured in the response business script in the Action Script set. Any new behavior will also need to be reflected in the `OmwApt.vbs` script to accommodate the communication with the OM-W server. Response communication with the OM-W server is done through WMI methods on the original WMI Message Object.

As an example, the following code illustrates the Acknowledge response and all its components:

AlarmPoint Action Scripts:

- Presentation script:

```
  $content.choices = "Acknowledge"
  $content.choices::add( "Own" )
```

- Response script:

```
  # Handle responses
  $reply = $response.reply
  $reply::toLowerCase()
  # For Mobile Devices Choices are Ack, Own and Ign so only check for beginning of
response choice.
  $own= $reply::startsWith( "own" )
  $change_sev = $reply::startsWith( "change sev" )
  $acknowledge= $reply::startsWith( "acknowledge" )
  IF (! $acknowledge )
    $shortAck= $reply::startsWith( "ack" )
  ENDIF
  $ignore = $reply::startsWith( "ignore" )
  IF (! $ignore )
    $shortIgn= $reply::startsWith( "ign" )
  ENDIF
  $annotate  = $reply::startsWith( "annotate" )
  IF (! $annotate )
    $shortAnnotate  = $reply::startsWith( "ann" )
  ENDIF
  $length = $reply::length()

  IF ( $acknowledge || $shortAck)
    # The AP Event will "delinkAll and delivered" for the responding user.
    # "You acknowledge a message when the problems causing that message to appear
are
    # resolved. Typically, you acknowledge messages when you have finished working
with
    # them. Acknowledging a message removes it from the active messages browser
view and
    # places it in the acknowledged messages browser." - OMW Help


    $message_note = "Acknowledged by " & $responder_name
    IF ( $main.debug )
      @script::log( $main.log_prepend & $message_note )
    ENDIF

    # Acknowledge OM-W Event
    @acknowledgeRequest = @event::createExternalServiceMessage()
    $acknowledgeRequest.request_text = "ApOmwAck"
    IF ( $main.annotate )
      # Annotation Optional Description Message
      $acknowledgeRequest.message_text   = $message_note
    ENDIF
    $acknowledgeRequest.omw_owner = $omwOwner
    $acknowledgeRequest.omw_password = $omwPassword
    @acknowledgeRequest::send()

    # delivered/delink-all
    @event::delivered( $responder_id )
    @event::delinkAll()

    IF ( $length > 12 && $acknowledge )
      # Retreive Extra Annotation Message
```

```
      $message_note = $response.reply::substring( 11 )
      $include_annotation = true
    ELSE-IF ( $length > 4 && $shortAck)
      # Retreive Extra Annotation Message
      $message_note = $response.reply::substring( 3 )
      $include_annotation = true
    ENDIF

    IF ( EXISTS($include_annotation))
      IF ( $main.debug )
        @script::log( $main.log_prepend & "Annotation - " & $message_note  & " -
Added by " & $responder_name )
      ENDIF

      # Annotate OM-W Event
      @annotationMessage = @event::createExternalServiceMessage()
      $annotationMessage.request_text = "ApOmwAnnotate"
      $annotationMessage.message_text = $message_note & " - Annotated by: " &
$responder_name
      $annotationMessage.omw_owner = $omwOwner
      $annotationMessage.omw_password = $omwPassword
      @annotationMessage::send()
    ENDIF
```

Note: *This is only a brief overview of the required components, For more information about AlarmPoint responses and scripting, refer to the AlarmPoint Action Scripts and the* AlarmPoint Developers Guide & Scripting Reference.

The ExternalServiceMessage injects the acknowledgeRequest to the Response Action Script of the `hp_operations_manager_win.xml` script. The following determines the type of response injected and creates a command line call to be executed which injects the necessary information to `OmwApt.vbs`:

```
  if ( APDT_request_text != void )
  {
    String [] StrObjArray = new String [0];
    String [] paramList = new String [] {"cmd", "/c", "cscript.exe", "/Nologo",
"integrationservices\\hpomw\\OmwApt.vbs", "-MsgId", APDT_incident_id, "-Request"};
    List commandList = new ArrayList( Arrays.asList( paramList ) );
    if ( APDT_request_text.equalsIgnoreCase( "ApOmwAck" ) )
    {
     paramList = new String [] {APDT_request_text, "-arg1", "\"" +
APDT_message_text + "\"", "-omwuser", "\"" + APDT_omw_owner + "\"", "-
omwpassword", "\"" + APDT_omw_password + "\""};
      commandList.addAll( Arrays.asList( paramList ) );
      execute( commandList.toArray( StrObjArray ) );
    }
```

Note: *For more information about working with the Response Action Script, refer to the* AlarmPoint Integration Agent Guide.

OmwApt.vbs:

```
'===== Main Program
…
```

```
Select Case g_strEventType
    Case "apomwack"
        AcknowledgeOmwMessage()
…
Sub AcknowledgeOmwMessage()

' Send Message to OMW to Acknowledge this Event
'================================================================================
==================================
 logText = "[ ACTION ] Acknowledging message " & messageGuid
 logType = "toFile"
 WriteLog

  Dim oOutParams

  Set oOutParams = g_objOVMessage.ExecMethod_( "Acknowledge")

  If (oOutParams.ReturnValue < 0) Then
    ' Update the Annotation message to include the error
    g_strArg1 = "[ AlarmPoint ] An error occurred while trying to process an
Acknowledge action: OV_Message Class WMI Acknowledge() had a ReturnValue: " &
oOutParams.ReturnValue
  AddOmwAnnotation()

    logText = "[ ERROR ] Acknowledge OV_Message WMI Call ReturnValue: " &
oOutParams.ReturnValue
  Else
    logText = "[ DEBUG ] Acknowledge OV_Message WMI Call ReturnValue: " &
oOutParams.ReturnValue
  End If

  logType = "toFile"
  WriteLog

 ' Check if optional Annotation value is present
 If ( g_strArg1 <> "" )  Then
  AddOmwAnnotation()
 End If
'================================================================================
==================================
End Sub   'AcknowledgeOmwMessage  Send Message to AlarmPoint to Acknowledge this
Event
```

Note: *For more examples of Visual Basic script, see the other scripts provided with the integration, and consult the* Microsoft Visual Basic Scripting Reference.

5.3.3 Filtering and Suppression of Event Data

If a subset of Event Data from the OM-W policies needs to be filtered or suppressed, you can construct a rule similar to the following:

"Suppress all messages from Nodes that match *DEV* for a period from 17:00 to 08:00 M-F and 24h on Sat & Sun."

For more information about filtering and suppression, see the *AlarmPoint Integration Agent Guide*.

5.3.4 Annotations

This integration extensively annotates the originating OM-W Event, but this may not be desirable in all environments. To turn off successful delivery annotations, set both $track_delivery and $main.annotate to *false* in the initial script, as illustrated by the following code sample:

```
# Track when each device is delivered to. Setting this to false may give a performance
# advantage, but you lose any information about whether a delivery was susccessful
# or not.
$track_delivery = false

# Enables submission of annotations back to the Management System.
$main.annotate = false

# Track when each device is delivered to for Subscriptions.
$track_subscriptionDelivery = false

# Enables submission of Subscription annotations back to the Management System.
$main.subscription_annotate = false
```

6. Configuration Variable Reference

This section outlines and describes the configuration variables available in the initial Action Script.

6.1 Local Configuration Variables

These variables are available only in this script, and control how the script runs. For more information about the initial process script, consult the *AlarmPoint Developer's Guide & Scripting Reference*

6.1.1 FYI and Subscription Notification Variables

The following variables configure the behavior of informational-only, or FYI, notifications. The value assigned to each variable is the default value within the script

Note: For more information on the behavior associated with informational-only notifications, see "FYI Notifications" on page 31.

Variable	Description
$force_fyi = "disable"	Forces notifications to be informational only rather than requiring responses. Possible values are: • **disable**: nothing is forced. • **on**: notifications are forced to be FYI. • **off**: notifications are forced not to be FYI.
$use_email_for_fyi = true **$use_phone_for_fyi = false** **$use_im_for_fyi = true** **$use_text_phone_for_fyi = true** **$use_text_pager_for_fyi = true** **$use_numeric_pager_for_fyi = true** **$use_bes_for_fyi = true** **$use_generic_for_fyi = true**	Configure Device filters for informational-only (FYI) notifications. Setting these flags to `false` prevents that Device type from being notified with informational (FYI) messages.
$enable_subs = true	Enables Subscription functionality. If set to `true`, Users subscribed to criteria matching the event will be notified. If set to `false`, no subscribed Users will be notified even if they match the criteria of the event.

Variable	Description
$subscription_fyi = true	Forces Subscription notifications to be informational only; recipients of a Subscription notification will not be able to respond to the event. **Note**: If the `$use_phone_for_fyi` flag is set to `true`, a User can respond with “delete”, which removes the notification from the phone queue, “save”, which moves to the next notification without deleting, or “repeat”, which replays the notification. The `$force_fyi` flag also forces subscriptions to be informational only. If both the `$force_fyi` flag and the `$subscription_fyi` flag are set to `false`, AlarmPoint will use the FYI flag submitted with the event from the Management System.

6.1.2 Fail-safe Configuration Variables

The following variables configure the fail-safe functionality, and specify when notifications will be sent to the fail-safe recipient. The value assigned to each variable is its default value within the script.

Note: *For instructions on how to set up a fail-safe recipient, see “Create a Fail-Safe Group” on page 17.*

Variable	Description
$fail_safe = "enabled"	Controls whether the fail-safe recipient is notified, and under which circumstances. Possible values are: • **enabled**: notify the fail-safe Group if no Subscriptions match and there are no notifiable recipients. • **for-subscriptions**: notify if the Subscription functionality is enabled and no Subscriptions match. • **for-recipients**: notify if there are no notifiable recipients. • **disabled**: disable the fail-safe functionality; no notifications will be sent to the fail-safe recipient.
$fail_safe_group = "OMW FailSafe"	Identifies the fail-safe recipient, which is typically a Group, but may be a User.

6.1.3 Alert Configuration Variables

The following variables configure Alert behavior. The value assigned to each variable is its default value within the script.

Variable	Description
$override_timeframes = false	Overrides any Device Timeframes that have been configured for a User for this notification.
$use_emergency_devices = false	Forces the use of emergency Devices as part of the Device resolution processing.

Variable	Description
$track_delivery = true	Configures the notification to run a response script when the delivery of a notification is successful. As this can limit Node performance, you can set this value to false if the custom behavior for successful delivery events is unnecessary, but you will lose any information about whether a delivery was successful.

6.1.4 Message Ownership Variables

The following AlarmPoint initial script variables configure the owner of the HP OM-W message when a User responds on a Device. For more information about these variables, see “Importing the AlarmPoint script package” on page 9.

Variable	Description
$main.hp_omw_owner_type = "default"	Specifies who should own the OM-W message for Own, Ack, or Change Severity responses. Possible values are: • **default**: Sets the WMI user to the value specified by the $main.hp_omw_owner variable. • **custom-field**: Sets the owner to the value specified in the AlarmPoint Custom Fields for the responding AlarmPoint User; must be a valid HP OM-W Windows account in the DOMAIN\USER format. To use the “custom-field” setting, you must create two custom text fields in AlarmPoint named “HP OMW User” and “HP OMW Password” and specify a valid DOMAIN\USER account and password for each User. If this variable is set to “custom-field” and either Custom Field is not specified, the “default” setting is used instead. For more information about creating Custom Fields in AlarmPoint, refer to the *AlarmPoint Installation and Administration Guide*.
$main.hp_omw_owner = "AlarmPoint"	Specifies the owner when “default” is set as the owner type; this value must be a valid HP OM-W Windows account in the DOMAIN\USER format.
$main.hp_omw_password = "PASSWORD"	Specifies the password when “default” is used for the main.hp_omw_owner_type

6.2 Global Configuration Variables

These variables are available throughout the script package, and are parameters of the “main” object. The value assigned to each variable is its default value within the script.

Variable	Description
$main.timeout = 86400	Amount of time (in seconds) the event is allowed to run before timing out. (86400 seconds = 24 hours.)
$main.debug = false	Indicates whether to log informational messages for debugging purposes. Disabling this variable may improve performance, but will provide less information.

Variable	Description
$main.use_logFile = false	Specify whether to use an alternate log file for debugging messages. This variable is ignored unless `$main.debug` is also set to true.
$main.logFile = "../logs/HP_OM-W_Script.log"	Defines the file used to log debugging information (only if `$main.use_logfile` is set to `true`).
$main.maxInvalidResponses = 3	Specifies the maximum number of invalid responses allowed before the notification will no longer be requeued. If a recipient sends an invalid response and this number has not been exceeded, they will be renotified with the same content, prefixed with a message indicating that their response was invalid.
$main.annotate = true	Enables submission of information back to the Management System. Information is logged throughout the script progress; if this variable is set to `true`, these logged messages will be annotated to the originating Management (OM-W) Event. Setting this variable to `false` may improve performance, but will make debugging difficult as some information may not be annotated to the originating event.
$main.subscription_annotate = false	Enables submission of Subscription information back to the Management System. (As with `$main.annotate`, but specifically for Subscription information.) Most Subscriptions are informational only; this variable can be enabled, for debugging and informational purposes but may reduce performance.
$main.enable_HTML_Email = true	Enables HTML Email functionality for email clients able to support HTML emails. If a client cannot support HTML than the plain text version will be passed.
$AlarmPoint_URL = "http://localhost:8888"	Identifies the AlarmPoint URL used for the HTML response form and AlarmPoint logo. If the specified URL cannot be reached, the logo will not appear, and the response links will not work.
$main.HTML_form_url = $AlarmPoint_URL & "/jsp/ProcessNotificationResponse.jsp"	Specifies the URL of the AlarmPoint Web Server's Process Notification Response JSP form, used by HTML email and BES to inject responses through the system.
$main.use_logo = true	Specifies whether HMTL email notifications will display the AlarmPoint (or custom) logo.
$main.logo = $AlarmPoint_URL & "/static/images/logos/alarmpoint/UNKNOWN.gif"	Specifies the path to the graphic displayed on HTML (email and BES) notifications.
$main.logo_alt_text = "[If the logo does not appear you may be blocking images or you may be outside a firewall. If the latter, the links will not work for responding and you should respond by replying to this email as described below.]"	The alternate text to display if the HTML email logo is unavailable. **Note**: If the logo does not display, it is unlikely that the `HTML_form_url` will work.

Variable	Description
$main.numeric_pager_number = "555-1212"	The phone number to display for calling in to retrieve event information. This variable has a non-existent number as a default value; a real call-in number must be supplied, or a message indicating that an AlarmPoint event notification has occurred.
$main.bes_pushurl = "http://localhost:8888/static"	Specifies the URL of the BES server. (Optional.)

7. Contact Us

You can access the AlarmPoint Systems Web Site at http://www.alarmpoint.com. From this site you can obtain information about the Company, the Products, Support and other helpful information. You may also access the Customer Support Site from the main web page. In this protected site you will find current product releases, helpful hints, patches, release notes, a helpful product knowledge base, trouble ticket submission areas and other helpful tools provided by AlarmPoint Systems, Inc.

AlarmPoint Systems, Inc.

4457 Willow Road, Suite 220
Pleasanton, CA 94588

Phone: 925-226-0300

Fax: 925-226-0310

Email: support@alarmpoint.com

Website: http://www.alarmpoint.com

Hewlett-Packard Company

3000 Hanover Street
Palo Alto, CA 94304-1185 USA

Phone: 650-857-1501

Fax: 650-857-5518

Support: http://support.openview.hp.com

Website: http://www.openview.hp.com

8. Copyright

xMatters produced this integration document to assist customers with joint HP/AlarmPoint Systems implementations. xMatters has made every effort to ensure that the information contained in this document is accurate, but do not guarantee any accuracy now or in the future. xMatters®, AlarmPoint Systems™, and AlarmPoint® are a trademark and registered trademark, respectively, of xMatters, inc. in the United States, United Kingdom and other jurisdictions. HP Operations Manager software for Windows is a registered trademark of HP Software, Inc. All other trademarks are the property of their respective owners.