AlarmPoint Systems

AlarmPoint for BMC Remedy Change Management

Validation Date
October 15, 2009
Version 2.1

Contents

1. Introduction

Welcome to the AlarmPoint for BMC Remedy Change Management integration. This document defines software requirements and describes installation, configuration, running select applications, and integration demonstrations for using BMC Remedy Change Management with AlarmPoint. These integration notes are intended for administrators and other technical readers.

1.1 Summary

AlarmPoint is an interactive alerting application, designed to capture and enrich important events, to route those events to the right person on any communication device, and to give that person the ability to solve, escalate, or enlist others to resolve the events remotely.

AlarmPoint allows you to take critical business information and contact the right people via voice phone, SMS, two-way pagers, instant message, and email.

Through integration modules, the AlarmPoint System can become the voice and interface of an automation engine or intelligent application (the Management System, such as BMC Remedy Change Management). When Remedy detects a change request that requires attention, AlarmPoint places phone calls, sends pages, messages, or emails to the appropriate personnel, vendors or customers.

The AlarmPoint System is also persistent, escalating through multiple devices and personnel until someone accepts responsibility or resolves the request. Once contacted, the AlarmPoint System gives the notified person instant two-way communication with BMC Remedy. Responses are executed immediately on BMC Remedy, enabling remote resolution of the request.

This integration supports event injection (from Remedy to AlarmPoint) via custom Remedy filters which trigger command line calls to the APClient utility. The injected event is enriched through web services within the APAgent to retrieve extra event parameters. The integration also supports response actions (from AlarmPoint to Remedy) through web services to approve, reject, hold, and annotate the original change request.

You will need to modify the configuration to suit your particular business requirements and adjust it to suit your expected loads. The default integration features automatic status annotations on the Remedy change request; in a high-volume production system, this can significantly affect performance. Consider your expected volume of injected Events and your server capacity when designing your own integration with AlarmPoint.

It is recommended that you set up a demonstration system as described in this documentation to become familiar with the integration before you deploy it into a pre-production or production environment.

1.1.1 Benefits

With the AlarmPoint integration, the support staff assigned to the change request can be notified directly via telephone, email, pager, BlackBerry, or other device. Information about the request will be presented to the approvers and decisions can be made in real-time such as accepting, rejecting, holding, and annotating the change request with informational messages.

Once a response is selected on the recipient’s remote device, AlarmPoint updates the Remedy change request in real-time. The benefit is that this process is immediate – significantly faster than the time required for support staff to handle the change request. In addition, the ability for approvers to update requests remotely allows for requests to be handled quickly and without the approver’s direct involvement.

During the process, every notification, response, and action is logged in AlarmPoint. In addition, AlarmPoint automatically annotates the original Remedy change request with status information.

The AlarmPoint product features a self-service web user interface to allow accurate assignment of responsible personnel for each job. AlarmPoint also includes an optional enhanced Subscription panel that allows both managed and self subscription to Remedy change requests. This Subscription Panel queries the Remedy Server directly in real time to retrieve lists of

Support Companies, Support Organizations and Support Groups, removing the need to manually create and maintain these lists.

The integration also includes a synchronization mechanism which implements AlarmPoint web services to load and update Remedy Groups, Users and Devices into AlarmPoint. Depending on the filters, Groups, Users or Devices are automatically synchronized with AlarmPoint whenever they are modified within Remedy. Users can also initiate batch synchronizations.

1.2 Configuration Overview

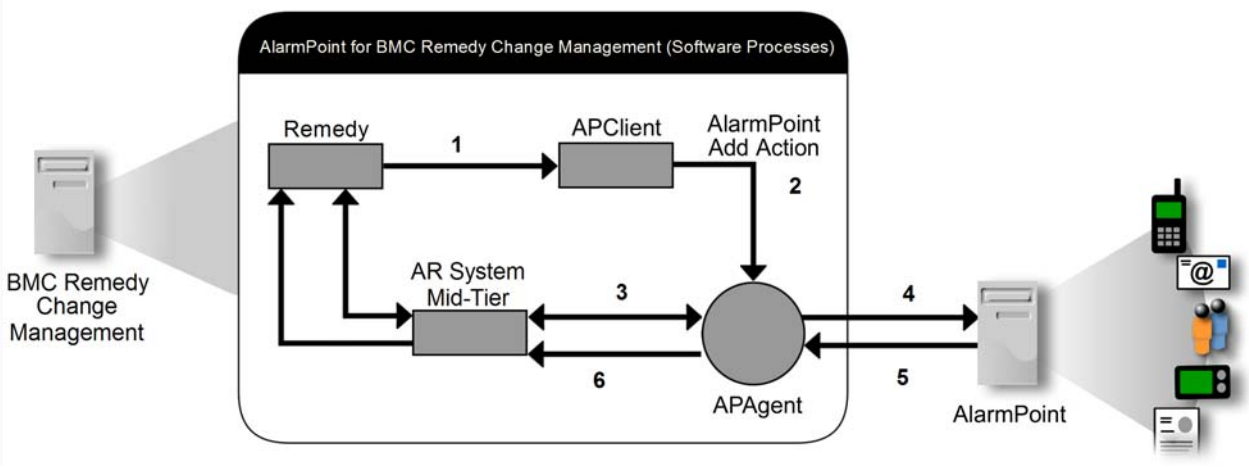
The basic steps of the integration are as follows:

1. Extract the integration archive file and install the integration components to the approporiate locations.
2. Configure the Remedy components to enable communication between AlarmPoint and Remedy.
   - This includes the Web Application URL, Object Structures, and Publish Channels.
3. Configure IBM WebSphere to add the configuration files to the classpath.
4. Configure the AlarmPoint components.
   - This includes the AlarmPoint scripts, the default AlarmPoint User, and the optional Subscription panel.
5. Configure the Data Synchronization tool to indicate what information in Remedy should be used to create Users and Groups in AlarmPoint.
   - The integration includes an "initial synchronization tool" that copies the user and group information from Remedy into AlarmPoint.
6. Validate the integration by transitioning a change request in Remedy, which triggers a notification in AlarmPoint. Respond to the notification on a User's Device, and view the results of the response in Remedy.

After you have completed the above configuration steps, you can use the instructions in "Optimizing and Extending the Integration" on page 39 to customize the integration for your deployment.

1.2.1 Architecture

This section provides a high-level overview of the major components of this integration.



The following steps occur during the notification process (the steps correspond to the figure above):

1. Remedy calls the AlarmPoint Java Client, mapping the change request ID.
2. The APClient sends an Add Action to the APAgent.

3. The APAgent retrieves the change request details via web service requests.
4. The enriched message is sent to AlarmPoint, which notifies the appropriate recipients.
5. The response returns to the APAgent.
6. APAgent updates the Remedy change request via web service requests.

Synchronization Architecture:

Manual or interactive Data Synchronization triggers the following steps:

1. A Remedy filter calls the AlarmPoint Java Client, mapping the object's request ID.
2. The Java Client retrieves the object's details using the Remedy Java API.
3. The object's AlarmPoint representative is updated or created via web service requests.

1.3 System Requirements

This integration requires the following products and components:

- AlarmPoint 4.0 (patch 004 or later) with AlarmPoint Java Client 4.0
  OR
  AlarmPoint 3.2.1 (patch 012 or later) with AlarmPoint Java Client 3.2.1
- BMC Remedy Action Request System 7.5 patch 002
- BMC Remedy Mid Tier 7.5 patch 002
- BMC Remedy Change Management 7.5.01

Consult the respective user manuals for a detailed list of hardware and supporting software requirements.

1.3.1 Supported Operating Systems

The following operating systems are supported by this integration:

- Microsoft Windows 2003 (validated)
- Sun Solaris 9, 10
- HP-UX PA-RISC B.11.23
- AIX 5.3
- Linux RedHat AS/ES 4, 5

1.4 Conventions & Terminology

This section describes how styles are used in the 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 location to which you install AlarmPoint is referred to throughout this document as `<APHOME>`.

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

The location to which you install the AlarmPoint Java Client is referred to as `<APJC>`.

1.4.2 Terminology

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

Term	Meaning
AlarmPoint Application Server Node	The core AlarmPoint application, consisting of various components that process events and perform notifications
AlarmPoint Notification Server Node	Delivers notifications to a User's Device
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 BMC Remedy Change Management
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 section describes how to install the AlarmPoint for BMC Remedy Change Management Integration

2.1 AlarmPoint System

This integration requires the following AlarmPoint applications to be installed:

- AlarmPoint
- AlarmPoint Java Client
- AlarmPoint Developer IDE

For installation instructions, refer to the *AlarmPoint Installation and Administration Guide*, the *AlarmPoint Java Client Guide*, and *AlarmPoint Online Developers Guide*.

2.2 Installing BMC Remedy Change Management

Consult the Remedy documentation for instructions on how to install and configure BMC Remedy Action Request System, BMC Remedy Mid-Tier, and BMC Remedy Change Management..

2.3 Installing the Integration

After the Remedy and AlarmPoint systems have been installed, integration components must be placed on each system. The integration archive contains updates for AlarmPoint and the AlarmPoint Java Client, and files for the BMC Remedy Server.

Note: *Before installing the integration, shut down or stop all AlarmPoint processes.*

Extract the `AP-BMC-REMEDY-CM` archive file.

The significant files and directories (in bold) of the archive are as follows:

```
|-- components
| |-- alarmpoint
| | |-- scripts
| | | |-- 3.2.1
| | | | '-- AP-BMC-Remedy-CM.aps
| | | '-- 4.0.x
| | |   '-- AP-BMC-Remedy-CM.aps
| | |-- sub_panel
| | | '-- CMSubscriptionForm.jsp
| | '-- vox
| |-- alarmpoint-java-client
| | |-- AP-BMC-Remedy
| | | |-- 3.2.1
| | | | |-- bmcremedycm.xml
| | | | |-- bmcremedy_batch_groups.xml
| | | | |-- bmcremedy_batch_memberships.xml
| | | | |-- bmcremedy_batch_users.xml
| | | | |-- bmcremedy_groups.xml
| | | | |-- bmcremedy_memberships.xml
| | | | |-- bmcremedy_users.xml
| | | | '-- bmcremedy_wod_report.xml
| | | '-- 4.0.x
```

```
| | | |-- bmcremedycm.xml
| | | |-- bmcremedy_batch_groups.xml
| | | |-- bmcremedy_batch_memberships.xml
| | | |-- bmcremedy_batch_users.xml
| | | |-- bmcremedy_groups.xml
| | | |-- bmcremedy_memberships.xml
| | | |-- bmcremedy_users.xml
| | | '-- bmcremedy_wod_report.xml
| | |-- BMC Remedy API
| | | |-- AIX
| | | | `-- arapi701
| | | |-- HP-UX
| | | | `-- arapi701
| | | |-- Linux
| | | | `-- arapi701
| | | |-- Solaris
| | | | '-- arapi701
| | | '-- Windows
| | | '-- arapi701
| | '-- lib
| |   |-- apbridge.jar
| |   '-- log4j.xml
| '-- remedy-cm
|   |-- AP_Configuration.arx
|   '-- APCMIntegration.def
|-- documentation
| '-- AP40-BMC-Remedy-CM.pdf
|-- release-notes.txt
'-- version.properties
```

2.3.1 Installing the AlarmPoint Java Client Integration Files

The following items must be installed for the AlarmPoint Java Client to communicate with Remedy:

- Integration Scripts
- AlarmPoint Bridge
- BMC Remedy AR System API

To install the AlarmPoint Java Client integration files:

1. Copy the `components\alarmpoint-java-client\AP-BMC-Remedy` folder from the extracted integration archive to `APAgent\etc\integrations`.
2. Open the `bmcremedycm.xml` file, and locate both occurences of the following code:

```
final String REMEDY_URL = "http://localhost:80";
final String REMEDY_NAME = "LOCALHOST";
final String REMEDY_USER = "Demo";
final String REMEDY_PASSWORD = "";
```

3. Replace the value between the quotes (do not replace the double quotes) with the following values:
   - **REMEDY_URL**: the IP or hostname of the machine hosting the Remedy AR System
   - **REMEDY_NAME**: the AR System server name
   - **REMEDY_USER**: the login name of a Remedy Approval Server Process Administrator with Override Only Admin authority (for more information about configuring a process administrator, see the Remedy Approval Server Guide)

- **REMEDY_PASSWORD**: the password for the Process Administrator

4. Save and close the `bmcremedycm.xml` file.

Note: *If you have already installed the AlarmPoint for BMC Remedy Service Desk Integration, you need to edit only the* `bmcremedycm.xml` *file. You can skip the remaining steps and proceed to* *"Importing the AlarmPoint Script Package" on page 8.*

5. Open the remaining XML files in the `APAgent\etc\integrations\AP-BMC-Remedy\` folder and locate the following code in each file:

```
String remedyUser = "Demo";
String remedyPass = "";
String remedyServer = "localhost";
```

6. For each file, replace the value within the quotes with the following values:
   - **remedyUser**: the login name of a user in Remedy with the following Permission Groups: Change Master, Change Submitter, Asset Viewer, Administrator.
   - **remedyPass**: the password for the remedyUser
   - **remedyServer**: the IP or hostname of the machine hosting the Remedy AR System
7. Save and close the files.
8. Copy the contents of `components\alarmpoint-java-client\lib` from the extracted integration archive to `APAgent\jre\lib\ext`.
9. In the extracted archive, locate the `arapi701` folder specific to the operating system on which the AR System is installed (for example, `components\alarmpoint-java-client\BMC Remedy API\HP-UX\arapi701`), and copy it to `APAgent`.
10. In the extracted archive locate the following jar files from the `arapi701` folder, and copy them to `APAgent\jre\lib\ext`:

```
arapi70.jar
arutil70.jar
axis.jar
```

Note: `axis.jar` *is a Windows-specific component, and is not required for Unix installations.*

2.3.2 Installing the Logging Configuration File

To enable additional logging for the integration, you must install the logging configuration file.

To install the logging configuration file:

Copy the `components\alarmpoint-java-client\log4j.xml` file from the extracted integration archive to the installation folder of the Java Client; for example, `APAgent`.

2.3.2.1 Specifying a Logging Level

The level of log detail can be configured independently for each logger. The available levels, from highest to lowest detail level, are: TRACE, DEBUG, INFO, WARN, ERROR, and FATAL.

To change the logging detail:

1. Open the `APAgent\log4j.xml` file.

2. Locate the logger for which you want to change the logging detail.
3. Change the value attribute to the desired level.
4. Save and close the file.

Example:

```
<!-- Logger for APAgent Scripts: ie bmcremedy_users -->
<logger name="com.invoqsystems.integration.remedy">
  <level value="WARN"/>
...
```

Change WARN to the desired logging level; e.g., DEBUG for more detailed logging.

2.3.3 Importing the AlarmPoint Script Package

This step requires the AlarmPoint Developer IDE.

To import the AlarmPoint Script Package:

1. Launch the 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. Select the `components\alarmpoint\scripts\AP-BMC-Remedy-CM.aps` file in the extracted integration archive.
4. In the File dialog box, click **Open**, and then click **OK**.
5. Right-click the **BMC Remedy Change Management (Business)** folder.
6. In the Validation dialog box, select **Validate**.
7. Right-click the **BMC Remedy Change Management (Business)** folder, and then click **Check In**.
8. In the Check In dialog box, click **Create**.
9. In the Check In dialog box, click **Remove**, and then click **Close**.
10. Close the IDE.

2.3.4 Installing the Web Services Library

To enable web service calls between the AlarmPoint and Remedy servers used in the subscription panel, you must copy the JAR file into the AlarmPoint Webserver library folders.

Source File:

```
components\alarmpoint\sub_panel\lib\apbridge.jar
```

Web Server Destination Directory:

- **Windows**:

  `C:\Program Files\AlarmPointSystems\AlarmPoint\webserver\webapps\cocoon\WEB-INF\lib`

- **Solaris**:

  `/opt/alarmpointsystems/alarmpoint/webserver/webapps/cocoon/WEB-INF/lib`

Note: *If you have installed more than one web server, install the JAR file into the library folder for each one.*

2.3.5 Installing the Subscription Files

To use the optional Subscription Panel, you must copy the JSP file into the AlarmPoint installation folder. If you have more than one web server, repeat the following steps for each one.

To install the JSP file:

1. On the AlarmPoint server, locate the web server installation folder:

   `<AlarmPoint>\webserver\webapps\cocoon\alarmpoint\jsp\subscription`

2. Create a subfolder named `bmcremedycm`.
3. Copy the `components\alarmpoint\sub_panel\CMSubscriptionForm.jsp` file from the extracted integration archive into the new `bmcremedycm` directory.
4. Restart the AlarmPoint Webserver.

2.3.6 Installing the Voice Files

These files must be installed into an AlarmPoint deployment running a Voice Device Engine. For more information, refer to the *AlarmPoint Installation and Administration* Guide.

To install the voice files:

Copy all of the files in the `components\alarmpoint\vox\english` folder from the extracted integration archive to the following node installs folder:

`<AlarmPoint>\node\phone-engine\Datastore\domains\common\recordings\english\phrases`

Note: *This integration provides a complete set of English voice files.*

3. Configuration

Before you can begin using the integration, you must configure BMC Remedy and AlarmPoint. This chapter explains the configuration processes required for each product and the configuration processes for the optional User Synchronization and Custom Remedy Forms components.

3.1 Configuring BMC Remedy

Configuring Remedy for the integration requires the following steps:

- Import the workflow definition file
- Import default configuration values
- Add the AlarmPoint entry for WorkLog information
- Modify the APSDI:AlarmPointConfiguration Form
- Configure includes and excludes for synchronization
- Verify the filter settings

3.1.1 Importing the Workflow Definition File

The workflow described in this document is provided in a definition file that must be imported into Remedy.

To import the workflow definition file:

1. Login to the BMC Remedy Developer Studio, and then click **File > Import**.
2. Select **BMC Remedy Developer Studio > Object Definitions**, and then click **Next**.
3. Select the AR System server into which you want to upload the integration objects, and then click **Next**.
4. Do one of the following:
   - Type in the location of the `APCMIntegration.def` file.
   - Click the **Browse** button to the right of the text field and navigate to the location of the `APCMIntegration.def` file. Select the file, and then click **Open**.
5. Click **Next**.
   - If you have already imported a workflow definition file, ensure that you select the **Replace Objects on the Destination Server** check box, but note that any changes you have made to those objects will be lost. If you are sure the changes you made are necessary for your installation, ensure that the existing objects will work with the objects being imported, and do not select the **Replace Objects on Destination Server** check box.
6. Click **Finish**.

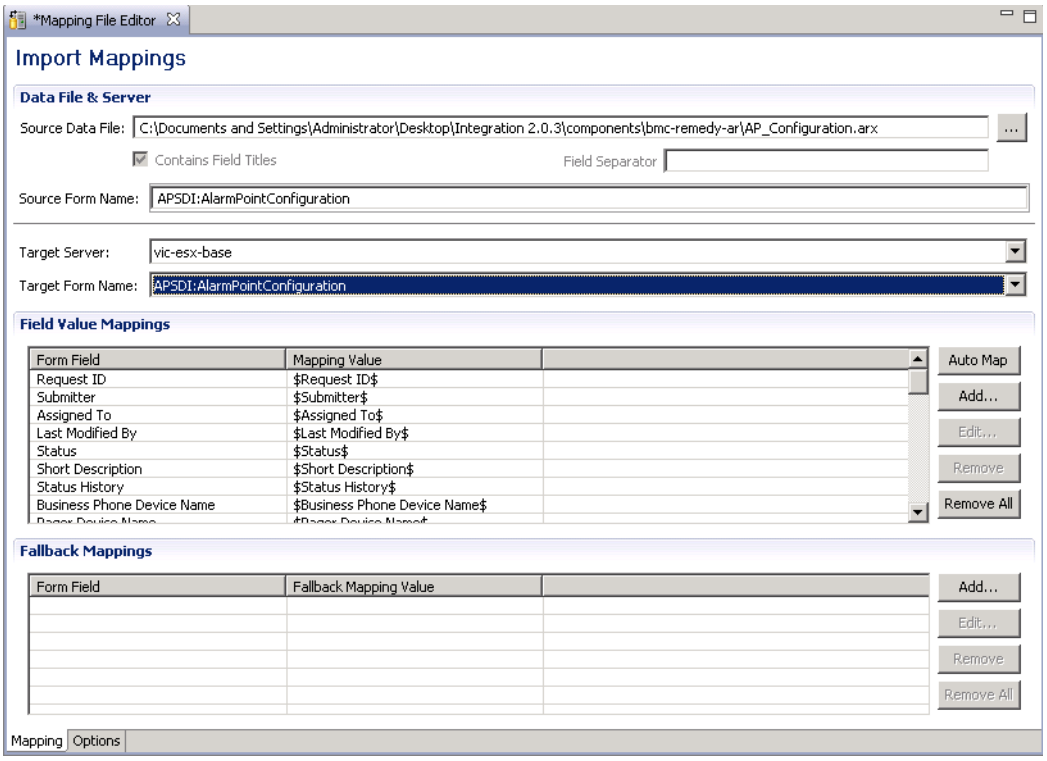
3.1.2 Import Default Configuration Values

The default configuration values imported into Remedy include web service connection values for the AlarmPoint Server and default values for synchronized Users, Groups, and Devices. This data is used by the forms to provide categories, items, and types for end-users to select when using this example integration.

When the data is imported, one instance of APSDI:AlarmPointConfiguration is created and named “default”; the integration is designed to look for an instance of APSDI:AlarmPointConfiguration with this name.

To import the default configuration values:

1. Login to the BMC Remedy Data Import tool, and then click **File > New Mapping**.
2. In the **Source Data File** field, do one of the following:
   - Type the location of the `AP_Configuration.arx` file.
   - Click the **Browse** button to the right of the text field, and navigate to the location of the `AP_Configuration.arx` file, located by default in the `AP-BMC-Remedy-ServiceDesk\components\bmc-remedy-ar` folder. Select the file and then click **Open**.
3. In the **Target Server** drop-down list, select the AR System server where the integration objects were imported.
4. In the **Target Form Name** drop-down list, select **APSDI:AlarmPointConfiguration**.
5. In the Field Value Mappings section, click **Auto Map**, and then select **Import > Start Import**.



3.1.3 Adding the AlarmPoint entry for Work Log Information

To enable annotations, add the AlarmPoint category to the CHG:WorkLog form.

To add the AlarmPoint category to the WorkLog Menu:

1. Login to the BMC Remedy Developer Studio tool, and expand the server tree where the integration objects are installed.
2. In the server tree, expand **All Objects**, and then double-click **Forms**.
3. In the Forms dialog box, locate and double-click **CHG:WorkLog**.
4. Double-click the **Work Info Type** field.
5. In the Properties view, expand the property **Attributes**, and then select the sub-property **Selections**.
6. Click the Browse button to the right of the **Value** column.

7. In the Selections dialog box, click **Add**.
   - A default value should be added to the bottom of the list.
8. In the **Selection Values** and **Alias** fields, specify **AlarmPoint**.
9. If the default ID provided is not acceptable, change it to a unique number.
10. Click **OK**, and then click **File** > **Save**.

3.1.4 Modify the APSDI:AlarmPointConfiguration form

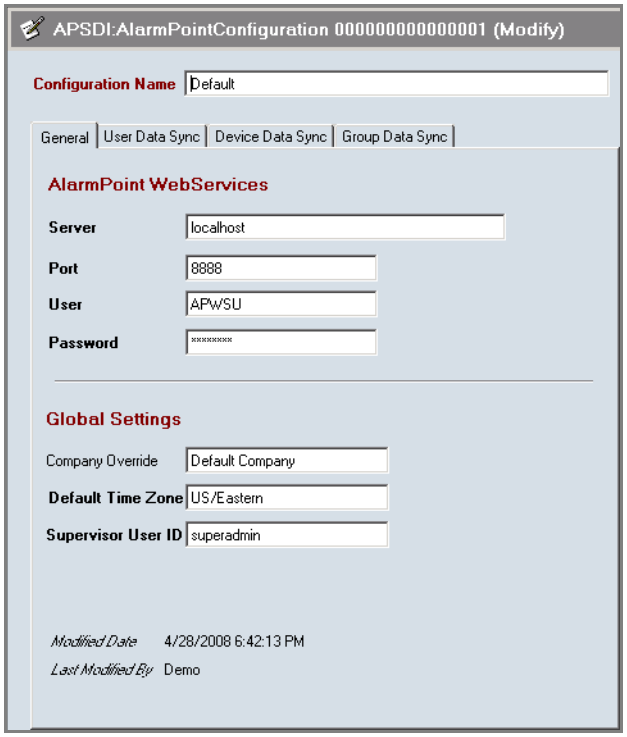
To enable Remedy to AlarmPoint communication via web services, the Server, Port, User and Password must be set on the APSDI:AlarmPointConfiguration form.

To configure the AlarmPoint form:

1. Log in to the Remedy User application.
2. In the Remedy User interface, under AlarmPoint Remedy - Change Management Integration in the Quick Links list, select **APCM:AlarmPoint Integration Console**.
3. In the Integration Console's left menu, click **Configuration** to expand.
4. Click **AlarmPoint Configuration**.
5. Change the settings on each of the tabs according to the following tables, and then click **Save** and **Close**.

3.1.4.1 Settings

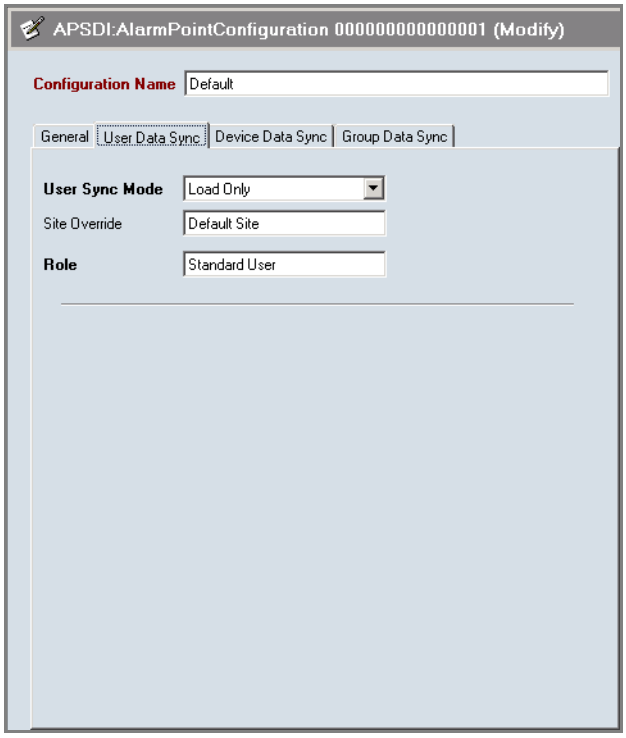
APSDI:AlarmPointConfiguration form - General tab:



Note: *If you are using an AlarmPoint installation with unmerged Administrator accounts, change the* ***Supervisor User ID*** *field to* `companyadmin`.

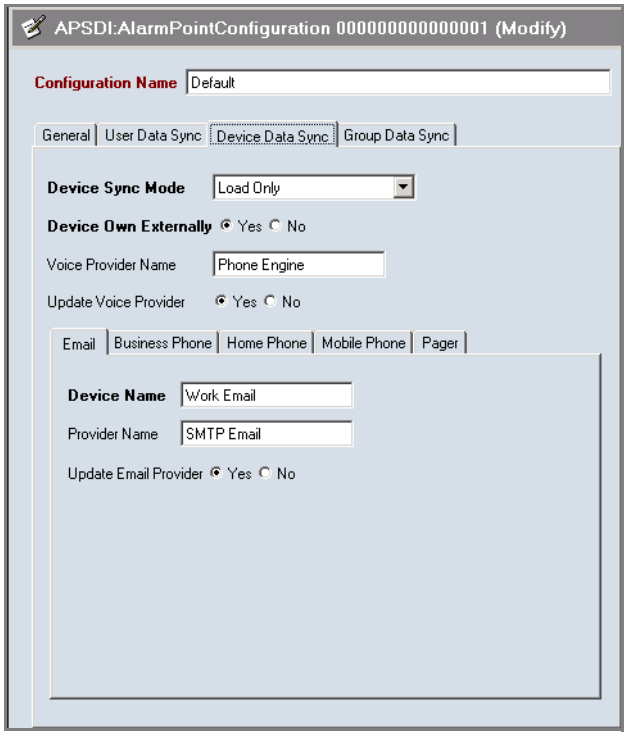
	Setting	Description
AlarmPoint Webservices	**Server**	Specifies the IP address or hostname of the machine running the AlarmPoint Webserver.
	Port	Specifies the port listening for requests to the AlarmPoint Webserver; the default is 8888.
	User	Specifies the name of the Web Service User created in AlarmPoint. The default is “APWSU”.
	Password	Specifies the password for the AlarmPoint Web Service User. The default value is “password”.
Global Settings	**Company Override**	Specifies a Company defined in AlarmPoint to use when loading or synchronizing Groups and Users from Remedy to AlarmPoint.
	Default Time Zone	Specifies the time zone to use when loading or synchronizing Groups from Remedy to AlarmPoint.
	Supervisor User ID	Specifies the supervisor in AlarmPoint to use when loading or synchronizing Users from Remedy to AlarmPoint.

APSDI:AlarmPointConfiguration form - User Data Sync tab:



Setting	Description
User Sync Mode	Specify one of the following options: • **Load Only**: Only update to AlarmPoint if the User does not already exist. • **Synchronize**: Always update the value from Remedy to AlarmPoint. • **No Load**: Take no action when a User is created or edited in Remedy.
Site Override	Specifies a Site in AlarmPoint to associate with all Users loaded or synchronized to AlarmPoint.
Role	Specifes a Role to assign to all Users loaded or synchronized to AlarmPoint. Note that future updates will not overwrite a User's Role settings.

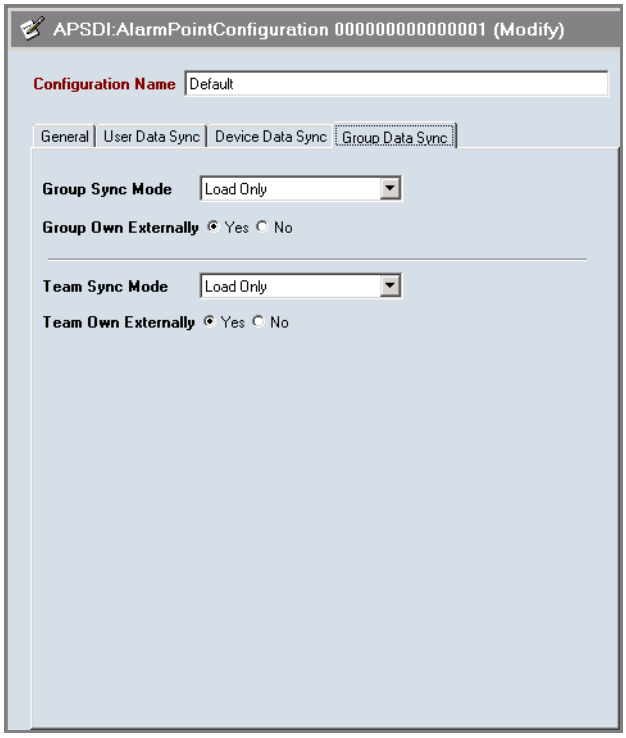
APSDI:AlarmPointConfiguration form - Device Data Sync tab:



	Setting	Description
Device Sync Settings	**Device Sync Mode**	Specify one of the following options: • **Load Only**: Only update AlarmPoint if the Device does not already exist. • **Synchronize**: Always update the value from Remedy to AlarmPoint. • **No Load**: Take no action when Device is created or edited in Remedy.
	Device Own Externally	Specifies whether all Devices updated to AlarmPoint should be labelled as "Externally Owned".
	Voice Provider Name	Specifies the name of a User Service Provider to associate with any business phone, home phone, or mobile phone updated to AlarmPoint.
	Update Voice Provider	Specifies whether updates to AlarmPoint will change the User Service Provider for voice Devices that already exist.

	Setting	Description
Device tabs	**Email**	**Device Name**: Specifies the Device Name to assign to all email Devices updated to AlarmPoint. **Provider Name**: Specifies the User Service Provider to associate with all email Devices updated to AlarmPoint. **Update Email Provider**: Specifies whether updates to AlarmPoint will change the User Service Provider for email Devices that already exist
	Business Phone	**Device Name**: Specifies the Device Name to assign to all business phones updated to AlarmPoint.
	Home Phone	**Device Name**: Specifies the Device Name to assign to all home phones updated to AlarmPoint.
	Mobile Phone	**Device Name**: Specifies the Device Name to assign to all mobile phones updated to AlarmPoint.
	NOTE: The Device Names for business phones, home phones, and mobile phones must be unique.	
	Pager	**Device Name**: Specifies the Device Name to assign to all pagers updated to AlarmPoint. **Provider Name**: Specifies the User Service Provider to associate with all pager Devices added to AlarmPoint. **Two-Way Pager**: Specifies whether pagers updated to AlarmPoint should be flagged as two-way capable. **Update Pager Provider**: Specifies whether updates to AlarmPoint will change the User Service Provider for pager Devices that already exist.

APSDI:AlarmPointConfiguration form - Group Data Sync tab:



Setting	Description
Group Sync Mode	Specify one of the following options: • **Load Only**: Only update AlarmPoint if the Group does not already exist. • **Synchronize**: Always update the value from Remedy to AlarmPoint. • **No Load**: Take no action when Group is created or edited in Remedy.
Group Own Externally	Specifies whether all Groups updated to AlarmPoint should be labelled as “Externally Owned”.
Team Sync Mode:	Specify one of the following options: • **Load Only**: Only update AlarmPoint if the Team does not already exist. • **Synchronize**: Always update the value from Remedy to AlarmPoint. • **No Load**: Take no action when Team is created or edited in Remedy.
Team Own Externally	Specifies whether all Teams updated to AlarmPoint should be labelled as “Externally Owned”.

3.1.5 Configuring Synchronization Includes and Excludes

This integration includes a data load and synchronization mechanism that provides for Event-driven updates of BMC Remedy data into Alarmpoint. Once pertinent data is initially loaded into AlarmPoint, it is automatically updated whenever modified by Remedy processes such as User or Group changes and deletes and Device creation and modification.

You can use the synchronization include and exclude filters to specify what Users and Groups are synchronized between Remedy and AlarmPoint.

To specify the synchronization includes and excludes:

1. Log in to the Remedy User application.
2. In the Remedy User interface, under AlarmPoint Remedy - Change Management Integration in the Quick Links list, select **APCM:AlarmPoint Integration Console**.
3. In the Integration Console's left menu, click **Configuration** to expand.
4. Do one of the following:
   - To filter Groups, click **Group Includes / Excludes**.
   - To filter Users, click **User Includes / Excludes**.

3.1.5.1 Adding User or Group entries in Include/Exclude

1. Click **Actions > New** to enter New Mode.
2. In the **Class** drop-down list, select **Exclude** or **Include**.
3. Type the name to filter:
   - For Group, type the Group name in the **Support Group Name** field.
   - For User, type the User ID in the **Remedy Login Name** field.
4. Click the **Add** button.

Note: *The exclude filter has precedence.*

3.1.5.2 Deleting User or Group entries in Include/Exclude

1. Click **Actions > Search** to enter Search Mode.
2. Specify your search criteria, and then click **Search**.
3. Select an entry in the results list, and then click **Actions > Delete**.
4. In the confirmation dialog box, click **OK**.

3.1.5.3 Populating Filters by Importing

The Includes / Excludes entries can also be imported using the Remedy Import tool. The following example illustrates an initial load; customize the steps as necessary to suit your integration.

To import Includes/Excludes:

1. Create a .csv file using one of the following templates:
   - For Group Includes / Excludes:

```
"Class","Support Group Name"
"Include","Executive Group"
"Exclude","Administration Group"
"Include","Network Managers"
```

   - For User Includes / Excludes:

```
"Class","Remedy Login Name"
"Include","Mary McBride"
"Exclude","Will Fuller"
"Exclude","Carol O'Grady"
```

2. Login to the BMC Remedy Data Import tool, and then click **File > New Mapping**.
3. In the **Source Data File** field, do one of the following:
   - Type the location of the `template.csv` file.
   - Click the **Browse** button to the right of the text field, and navigate to the location of the `template.csv` file. Select the file and then click **Open**.
4. Select the **Contains Field Titles** check box.
5. In the **Target Server** drop-down list, select the AR System server where the integration objects were imported.
6. In the **Target Form Name** drop-down list, select **APSDI:CONF:GroupIncludesExcludes** or **APSDI:CONF:UserIncludesExcludes**.
7. In the Field Value Mappings section, click **Auto Map**.
8. Select menu item **Import > Start Import**.

3.1.6 Verifying Filter Settings

This section describes how to verify the settings of filters that call the Java Client. The filter shown here responds to an active link when a notification is triggered by calling the AlarmPoint Client (`APClient.bin`) utility.

To verify filter settings:

1. Launch the BMC Remedy Developer Studio, if it is not already running.
2. In the AR System navigation pane, expand the server, and then expand the **All Objects** item.
3. Double-click **Filters**.
4. Using the Filtering Options feature, type `APCM:` in the value field, and then press **Enter**.
5. Double-click the **APCM:APActionInjection** filter.
6. Review the **If Actions > Run Process** entry, and ensure the path to `APClient.bin.exe` is correct.
7. If changes are required, expand the Run Process entry, edit the **Command Line: field**, and then click **File > Save**.
8. Repeat steps 3 to 6 for each of the following:
   - APCM:APActionDelete
   - APSDI:APC:BatchGroupSync
   - APSDI:APC:BatchMembershipSync
   - APSDI:APC:BatchUserSync
   - APSDI:APC:GroupModifications
   - APSDI:APC:MembershipModifications
   - APSDI:APC:UserModifications
   - APSDI:WOD:RequestReport

3.2 Configuring AlarmPoint

Configuring AlarmPoint requires the following steps

- Configure the AlarmPoint Java Client.

- Define an Event Domain.
- Set up a User with a two-way text phone.
- Add an AlarmPoint Web Services User.
- Configure the custom Subscription panel.

3.2.1 Configuring the AlarmPoint Java Client

The following AlarmPoint Java Client components must be modified to configure this integration:

- APAgent.xml
- Java Client scripts
- APAgent.conf
- Environment variables

3.2.1.1 Modifying APAgent.xml

To configure the domain mapping file to identify the `bmcremedycm.xml` file, add the following lines to the `APAgent\etc\APAgent.xml` file:

```
<alarmpoint-agent-client id="bmcremedycm" filename="integrations/bmcremedycm.xml" />
<alarmpoint-agent-client id="bmcremedy_batch_groups" filename="integrations/AP-BMC-Remedy/
bmcremedy_batch_groups.xml" />
<alarmpoint-agent-client id="bmcremedy_batch_memberships" filename="integrations/AP-BMC-
Remedy/bmcremedy_batch_memberships.xml" />
<alarmpoint-agent-client id="bmcremedy_batch_users" filename="integrations/AP-BMC-Remedy/
bmcremedy_batch_users.xml" />
<alarmpoint-agent-client id="bmcremedy_groups" filename="integrations/AP-BMC-Remedy/
bmcremedy_groups.xml" />
<alarmpoint-agent-client id="bmcremedy_memberships" filename="integrations/AP-BMC-Remedy/
bmcremedy_memberships.xml" />
<alarmpoint-agent-client id="bmcremedy_users" filename="integrations/AP-BMC-Remedy/
bmcremedy_users.xml" />
<alarmpoint-agent-client id="bmcremedy_wod_report" filename="integrations/AP-BMC-Remedy/
bmcremedy_wod_report.xml" />
```

3.2.1.2 Modifying APAgent.conf (Windows only)

On Windows systems, the APAgent configuration file must be modified to identify the Java library.

To modify the APAgent configuration file:

1. Open the `APAgent\etc\APAgent.conf` file in an editor.
2. In the section of the `APAgent.conf` file after the APAgent's environment variables (-D parameters) are declared, add the following:

```
#Include Remedy API
-Djava.library.path="C:\APAgent\arapi701"
```

Note: *The parameter must match the location of the* `arapi701` *folder specified in "Installing the AlarmPoint Java Client Integration Files" on page 6. Do not include a trailing slash (\) in the path as it prevents the Java library loader from finding the shared DLLs for the Remedy API.*

3.2.1.3 Multiple-Company Deployment Configuration

If you are installing the integration in an AlarmPoint deployment that contains multiple Companies, you will need to modify the configuration files to identify the Company for which you are configuring the integration.

To configure the APAgent for a multiple-Company deployment:

1. Open the `bmcremedycm.xml` file.
2. Add the following line in the <mapped-input.> XML element, just before the </mapped-input> closure tag:

```
<parameter index="constant" type="string" value="Default Company">company</parameter>
```

   - Change the “Default Company” string to reference the target Company name.

3. Save and close the file.
4. Open the `APAgent.xml` file.
5. Locate the “Default Company” string, and modify it to reference the target Company name.
6. Save and close the file.

3.2.1.4 Modifying the Environment Variables

The environment variables must be modified to locate the AR System Java API. Before performing these modifications, ensure that you have Administrator access.

To modify the Environment Variables:

1. Update the java.library.path environment variable as follows:
   - For Windows, include the extracted AR System API path (i.e., `C:\APAgent\arapi701`) in the Windows SYSTEM environment variable Path.
   - For Unix, edit `/opt/alarmpointsystems/APAgent/APAgent`, and change the JAVACMD variable to include the extracted Remedy API:

```
-Djava.library.path=\"/opt/alarmpointsystems/APAgent/arapi701\"
```

Note: *Incorporate the line before the -Dapagent.home variable in the JAVACMD variable.*

The resulting JAVACMD variable should resemble the following:

```
JAVACMD="\"${JAVAEXEC}\" ${JAVADEBUG} ${MINMEM} ${MAXMEM} -Djava.library.path=\"/
opt/alarmpointsystems/APAgent/arapi701\" -Dapagent.home=\"${INSTALL_DIR}\" -
Dpython.home=\"${INSTALL_DIR}/jython\" -Dwebapp=APAgent APAgent"
```

Note: *Match the exact case of the Unix path.*

---

2. Add the shared library paths to the `/opt/alarmpointsystems/APAgent/APAgent` launch script:
   - For Solaris, after the line:

```
JAVAEXEC="${INSTALL_DIR}/jre/bin/java"
```

   insert the following:

```
LD_LIBRARY_PATH="/opt/alarmpointsystems/APAgent/arapi701:${LD_LIBRARY_PATH}"
export LD_LIBRARY_PATH
```

- For HP-UX, after the line:

```
JAVAEXEC="${INSTALL_DIR}/jre/bin/java"
```

insert the following:

```
SHLIB_PATH="/opt/alarmpointsystems/APAgent/arapi701:${SHLIB_PATH}"
export SHLIB_PATH
```

- For AIX, after the line:

```
JAVAEXEC="${INSTALL_DIR}/jre/bin/java"
```

insert the following:

```
LIBPATH="/opt/alarmpoint/APAgent/arapi701:${LIBPATH}"
export LIBPATH
```

Note: *To load properly, the shared library files must have their execute bit set. Otherwise, exception errors occur when executing APBridge from the Remedy Response Action Script in the AlarmPoint Java Client environment.*

3.2.2 Defining an Event Domain

The AlarmPoint webserver must be running to perform this portion of the integration.

To define an Event Domain:

1. Log in 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**: bmcremedycm
   - **Description**: BMC Remedy Change Management Integration
   - **Script Package**: BMC Remedy Change Management
4. Click **Save**.

Note: *It is strongly recommended that you use the Event Domain name specified above. For the integration to be successful, the Event Domain name must match the Client ID of the AlarmPoint Java Client.*

3.2.3 Setting up a two-way Device

Depending on the level of assignment within Remedy for the change request, a Group or User can be targeted for notification. The following steps describe how to ensure the default User, “Bob Smith” exists and has a virtual text phone Device:

To set up a two-way Device:

1. In AlarmPoint, 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 exists.
6. Click **Reorder**, and set the virtual text phone to be the first Device in the list.
7. Click **Save**.

Note: *If this User is missing, create a User with a virtual text phone Device and use them as a target for notifications. For more information and instructions on how to perform these tasks, refer to the* AlarmPoint User Guide.

3.2.4 Adding an AlarmPoint Web Service User

This integration requires an AlarmPoint Web Service User for Remedy to communicate with AlarmPoint via web services.

For the integration to be successful, the AlarmPoint Web Service User must have the User ID and Password configured within the APSDI:AlarmPointConfiguration form; for more information, see “Modify the APSDI:AlarmPointConfiguration form” on page 12.

To setup an AlarmPoint Web Service User:

1. On the Users tab, in the Users menu, click **Add Web Service User**.
2. Enter the following information into the form:
   - **User ID**: APWSU
   - **Description**: AlarmPoint Web Service User
   - **Password**: type the User’s password (the default is “password”)
   - **Verify Password**: retype the password to verify it.
3. In AlarmPoint 4.0, select all of the web services listed in the Denied Web Services list, and then click **Add**.
4. Click **Save**.

3.2.5 Configuring the Subscription Panel

To allow Users to subscribe to specific criteria on injected Events, you must configure the Subscription panel. Configuring the Subscription Panel requires the following steps:

- Define the Event Domain predicates.
- Define a Subscription Domain.
- Configure the Subscription JSP.
- Create a Subscription.
- Create a fail-safe Group.

Note: *Before you can configure the optional Subscription Panel, you must install the CMSubscriptionForm.jsp file and the apbridge.jar, as described in “Installing the Subscription Files” on page 9.*

3.2.5.1 Defining Event Domain predicates

The default Subscription panel provided with the integration requires the following Event Domain predicates (case sensitive):

- SUPPORT_COMPANY
- SUPPORT_ORGANIZATION

- SUPPORT_GROUP
- CHANGE_TYPE
- STATUS
- IMPACT
- URGENCY
- PRIORITY

Note: *You can also use the following steps to add other predicates that you consider important and plan to add to the integration.*

To define the Event Domain predicates:

1. In AlarmPoint, click the **Developer** tab.
2. On the Event Domains page, click **bmcremedycm**.
3. On the Event Domain Details page, click **Add New**.
4. Add the following predicates to the Event Domain:

Predicate	Type	Important	Values	Description
SUPPORT_COMPANY	List		Automatically generated	A list of possible Support Companies from Remedy associated with the Change Manager.
SUPPORT_ORGANIZATION	List		Automatically generated	A list of possible Support Organizations from Remedy associated with the Change Manager.
SUPPORT_GROUP	List	**Yes**	Automatically generated	A list of possible Support Groups from Remedy associated with the Change Manager.
CHANGE_TYPE	List		Manually entered	A list of Change Types associated with Remedy Change Requests. Possible values are: • Project • Change • Release • Asset Configuration • Asset Management • Asset Lease • Purchase Requisition • Asset Maintenance

Predicate	Type	Important	Values	Description
STATUS	List	**Yes**	Manually entered	A list of the states for which a Remedy Change Request may be injected for notification. Possible values are: • Request For Authorization • Planning In Progress • Scheduled • Scheduled For Approval • Implementation In Progress • Rejected • Completed
IMPACT	List		Manually entered	A list of Remedy Change Request Impact values. Possible values are: • 1-Extensive/Widespread • 2-Significant/Large • 3-Moderate/Limited • 4-Minor/Localized
URGENCY	List		Manually entered	A list of Urgencies for the Remedy Change Request. Possible values are: • 1-Critical • 2-High • 3-Medium • 4-Low
PRIORITY	List	**Yes**	Manually entered	A list of Reported Priorities for a Remedy Change Request. Possible values are: • Critical • High • Medium • Low

Note: *For more information on the automatically generated list predicates, see "Configuring the Subscription JSP", below.*

3.2.5.2 Defining a Subscription Domain

The Subscription Domain is the reference point of the optional Subscription panel. You must create a Subscription Domain before you can create Subscriptions with the new panel.

By default, the scripts support the following response choices on "Approval" notifications only: "Approve", "Reject" and "Hold", and "Annotate" for email notifications. To enable two-way communications for Subscriptions, define the response

choices on the Select Appropriate Response Choices page as described in the following steps. If you require only one-way, informational notifications, do not specify any response choices.

To create a Subscription Domain:

1. On the Developer tab, in the Developer menu, click **Add Subscription Domain**.
2. In the Event Domain drop-down list, select **bmcremedycm**, and then click **Continue**.
3. On the Subscription Domain Details page, in the **Name** field, type `Remedy CM`.
4. In the **Description** field, type `BMC Remedy Configuration Management Sample Subscription`.
5. In the **Custom Page URL** field, enter the following path:

   ```
   jsp\subscription\bmcremedycm\CMSubscriptionForm.jsp
   ```

6. Click **Continue**.
7. On the Select Appropriate Response Choices page, specify the available responses for this Subscription, and then click **Continue**.

Note: *By default, Subscriptions are FYI (informational-only notifications). To enable two-way subscription notifications, set the $subscription_fyi variable to* false *in the configuration block of the initial PROCESS script.*

---

8. On the Select Appropriate Predicates page, add all of the predicates to the **Applied Predicates** list, and then click **Continue**.
9. On the Select Roles page, specify the Roles you want to be able to create Subscriptions on the Domain, and then click **Save**.

Note: *For more information about working with Event and Subscription Domains, see the* AlarmPoint Installation and Administration Guide.

3.2.5.3 Configuring the Subscription JSP

This integration is packaged with an optional Subscription panel which reads the Change Managers Support Information (Company, Organization and Group) from Remedy through web services. The source of the content supplied for these lists can be changed from web service calls to predefined predicate value lists, allowing Administrators to configure the Subscription panel in a demo mode.

To manually populate the predicate lists:

1. Open the `CMSubscriptionForm.jsp` found in `<AlarmPoint>\webservers\webapps\cocoon\alarmpoint\jsp\subscription\bmcremedycm` folder on the AlarmPoint Webserver.
2. Set the Boolean variable QUERY_PREDICATE_VALUES to `false`.
3. Save and close the `CMSubscriptionForm.jsp` file.
4. In AlarmPoint, click the **Developer** tab.
5. On the Event Domains page, click **bmcremedycm**.
6. On the Event Domain Details page, in the **Predicates** list, click **SUPPORT_COMPANY**.
7. Add to the predicate list values.
8. Repeat steps 6 and 7 for **SUPPORT_ORGANIZATION** and **SUPPORT_GROUP** in the predicates list.

The Support Company, Support Organization, and Support Group lists on the Subscription will now be populated with the predefined list values instead of the Web Service Call 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.*

If you want to populate the predicate values lists from Remedy through web service calls rather than the predefined predicate list values, you must configure the connection properties within the JSP file.

To configure the Subscription JSP to connect to Remedy through web services:

1. Open the `CMSubscriptionForm.jsp` found in `<AlarmPoint>\webservers\webapps\cocoon\alarmpoint\jsp\subscription\bmcremedycm` folder on the AlarmPoint Webserver install.
2. Within the Subscription JSP, locate the following section:

```
final String REMEDY_URL = "http://localhost:80";
final String REMEDY_NAME = "LOCALHOST";
final String REMEDY_USER = "Demo";
final String REMEDY_PASSWORD = "";
```

3. Replace the value within quotes for each parameter as described in the following table:

Parameter	Description
REMEDY_URL	The URL for the Remedy web services
REMEDY_NAME	The Remedy ARS Server name that hosts web services
REMEDY_USER	Remedy User Login Name
REMEDY_PASSWORD	Password for the Remedy User Login Name

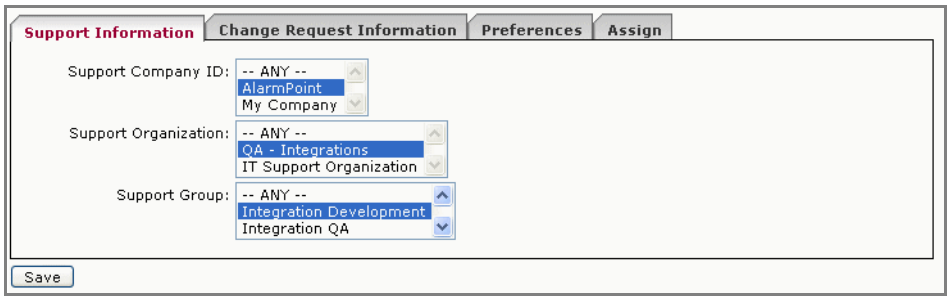
4. Save and close the JSP.

3.2.5.4 Creating a Subscription

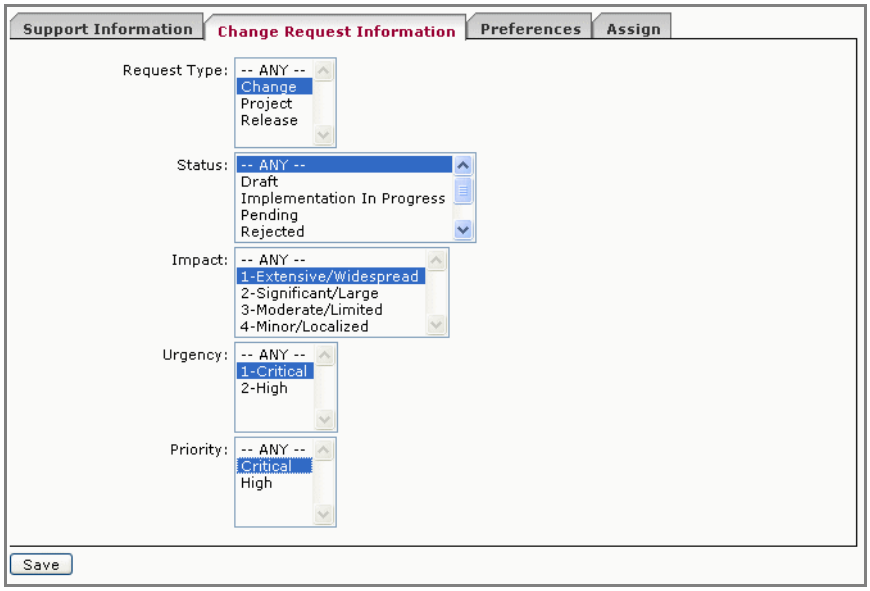
You can now use the custom Subscription panel to subscribe to Remedy Events of specific criteria, such as those of "Critical" Priority.

To create a Subscription:

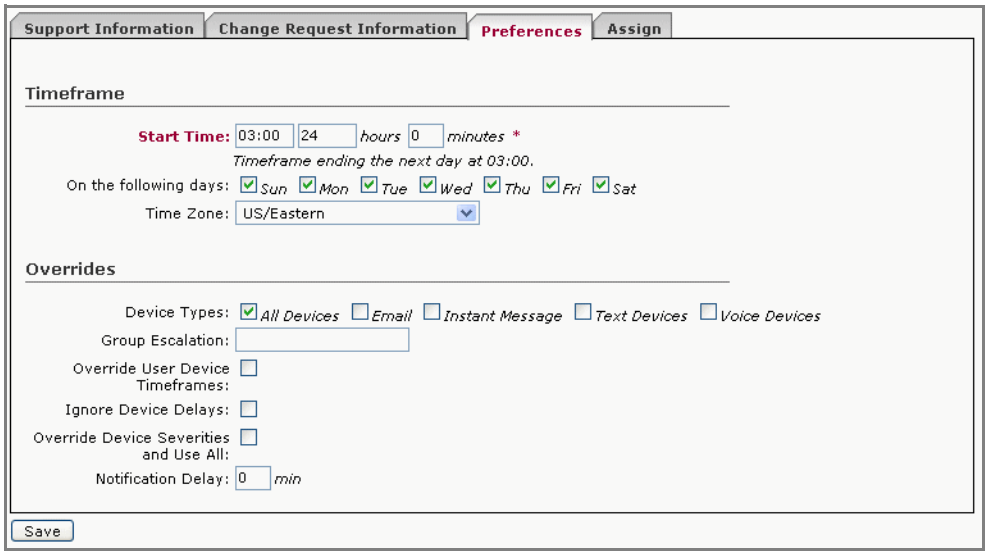
1. On the Alerts tab, in the Alerts menu, click **Assign Alerts**.
2. Select the **Remedy CM** Subscription Domain, and click the **Add New** link.
3. On the Subscription Details page, specify a name for the Subscription, and set the Subscription criteria using the Support Information, Change Request Information, and Preferences tabs.
   - The Support Information tab (Ctrl-click to select more than one value):



- The Change Request Information tab (Ctrl-click to select more than one value):

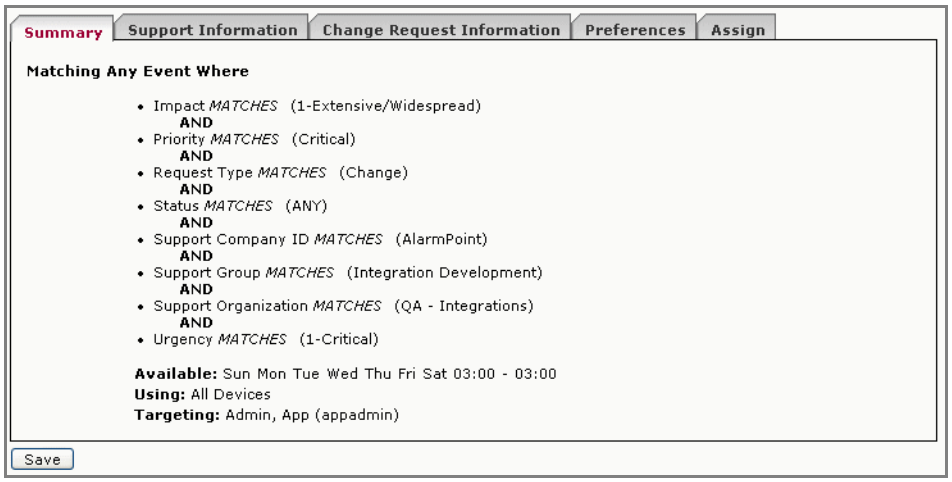


- The Preferences tab (defines the Timeframe and Overrides applied to Events for Subscription notifications):



4. The Assign tab (allows managers to assign Subscriptions to specific recipients):

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:



3.2.5.5 Creating 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. In AlarmPoint, click the **Groups** tab.
2. Create a new Group named **Remedy 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 an existing Group or a different Group name, modify the value for the $fail_safe_group variable defined in the initial PROCESS script in the AlarmPoint Action Scripts. You can also eliminate notifying any fail-safe group by setting $fail_safe to disabled.*

3.2.5.6 Excluding Subscriptions

If you do not want to use the Subscription functionality, you can remove it from the AlarmPoint Action Scripts by setting the $enable_subs variable to “false” in the configuration block of the initial PROCESS script.

3.3 Custom Remedy Forms

The AlarmPoint integration is installed with the APCM:AlarmPoint Integration Console that provides a launching point to view injected notifications, initiate synchronization, submit manual notifications, and query the Who Is On Duty Report. These are used to synchronize Remedy information with AlarmPoint, send messages to Groups and Users, and determine who is on duty at the moment.

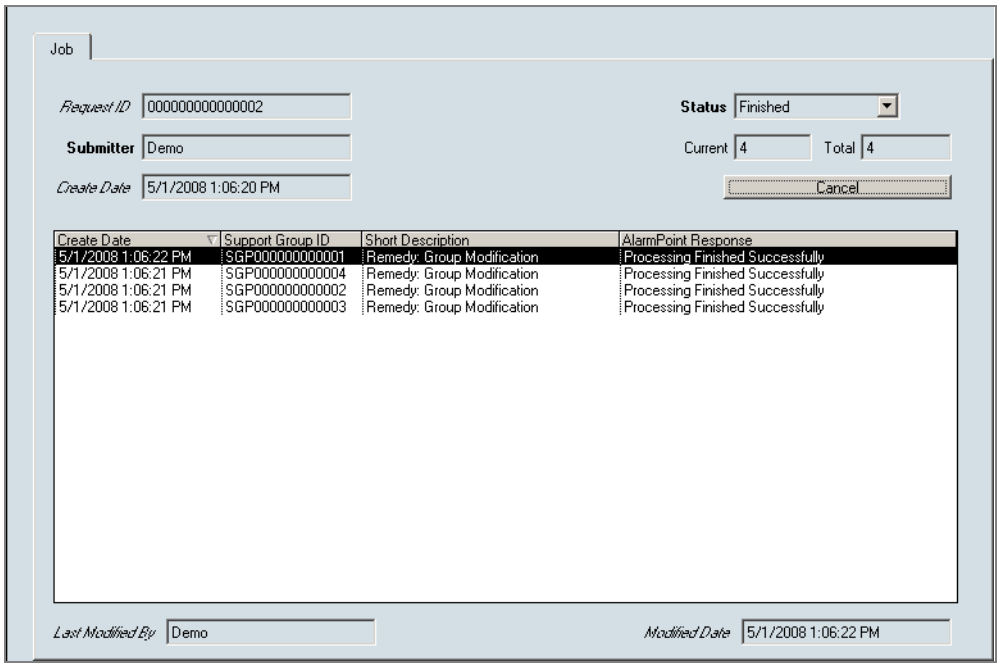
3.3.1 Synchronization

From the Integration Console, system-wide synchronizations can be launched for Users, Groups and Group Memberships.

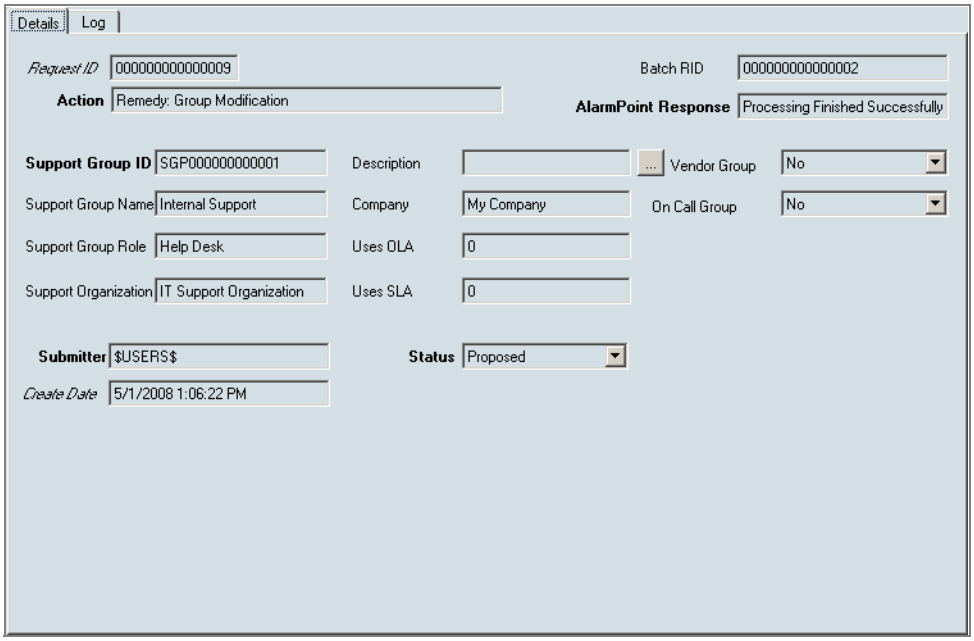
Note: *Users, Groups and Group Memberships are automatically synchronized whenever they are modified within Remedy. To customize this configuration, see "Modify the APSDI:AlarmPointConfiguration form" on page 12.*

To manually synchronize Users, Groups, and Group Memberships:

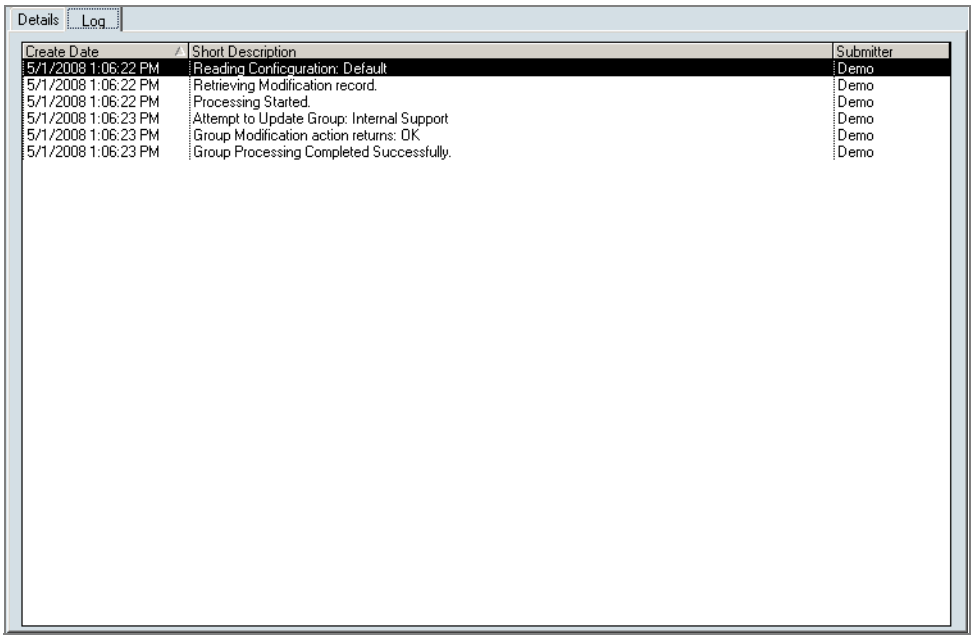
1. Launch the Remedy User application (**Start > Programs > Action Request System > BMC Remedy User**).
2. Provide a User Name and Password (as an example, the Demo account with no password is used).
3. In the Remedy User interface, under AlarmPoint Remedy - Change Management Integration in the Quick Links list, select **APCM:AlarmPoint Integration Console**.
4. In the AlarmPoint Integration Console's left menu, select one of the following, and then click **Initiate**:
   - **Batch User Synchronization**
   - **Batch Group Synchronization**
   - **Batch Membership Synchronization**
5. To monitor the progress, click **Actions > Search**.
6. Click **Search**.
7. Navigate to the newly initiated synchronization.
   - The status of each User or Group is displayed in the lower table:



8. Double-click an entry in the lower table to display further details:



9. Click the **Log** tab to view logging details:



Note: *When synchronizing Users, only Support Users are synchronized. Batch Membership is organized by Group; each Support Group will only get processed once.*

3.3.1.1 Synchronization Filters

The filters used by the synchronization are:

Dynamic:

The following create UserModificationForm object in Remedy:

- APSDI:UserAdd_Push
- APSDI:UserDelete_Push
- APSDI:UserModify_Push

The following create GroupModificationForm object in Remedy:

- APSDI:GroupAdd_Push
- APSDI:GroupDelete_Push
- APSDI:GroupModify_Push

Batch:

- Create UserModificationForm object in Remedy from within `bmcremedy_batch_users.xml`
- Create GroupModificationForm object in Remedy from within `bmcremedy_batch_groups.xml`

3.3.2 Quick Message

The Quick Message page sends a message through AlarmPoint to a list of Groups or Operators. All messages contain the following information:

- **Company, Group, User**: Use these fields to filter on the Group or User to notify.

- **Change ID**: A field that is meant to help associate this message with an change request.
- **Submitter**: Automatically populated with the User ID of the person submitting the message.
- **Message**: Text of the message you want to send to the targeted Groups and Operators.

With the exception of the Submitter field, all of these fields are auto-populated if redirected from another form. The Change Form can be configured to redirect to the Quick Message form; for more information, see “Adding Buttons to the Change Request Console” on page 40.

To redirect a notification from the APCM:AlarmPoint Integration Console, select an item in the list, and then click **Manual Message**.

3.3.3 Who Is On Duty Report

The Who Is On Duty Report is a method of determining who is on duty for a Group at the current time. The behavior of this page is as follows:

1. Use the **Company** and **Group** fields to specify the Group for which you want to run the report.
   - This Group is taken from a list in Remedy, therefore the Group may not be synchronized to AlarmPoint.
2. Click **Request Report** to send the request to the AlarmPoint Webserver.

To view the result in Remedy:

1. While in the APSDI:WhosOnDuty form, click **Actions > Search**.
2. Click the **Search** button
3. Select the newly created report (the Status must be Report Completed before values displayed are correct).
4. Click **Notify Group** to notify the Group.
5. Double-click one of the Users listed in the table to notify the User.

4. Software Component Integration

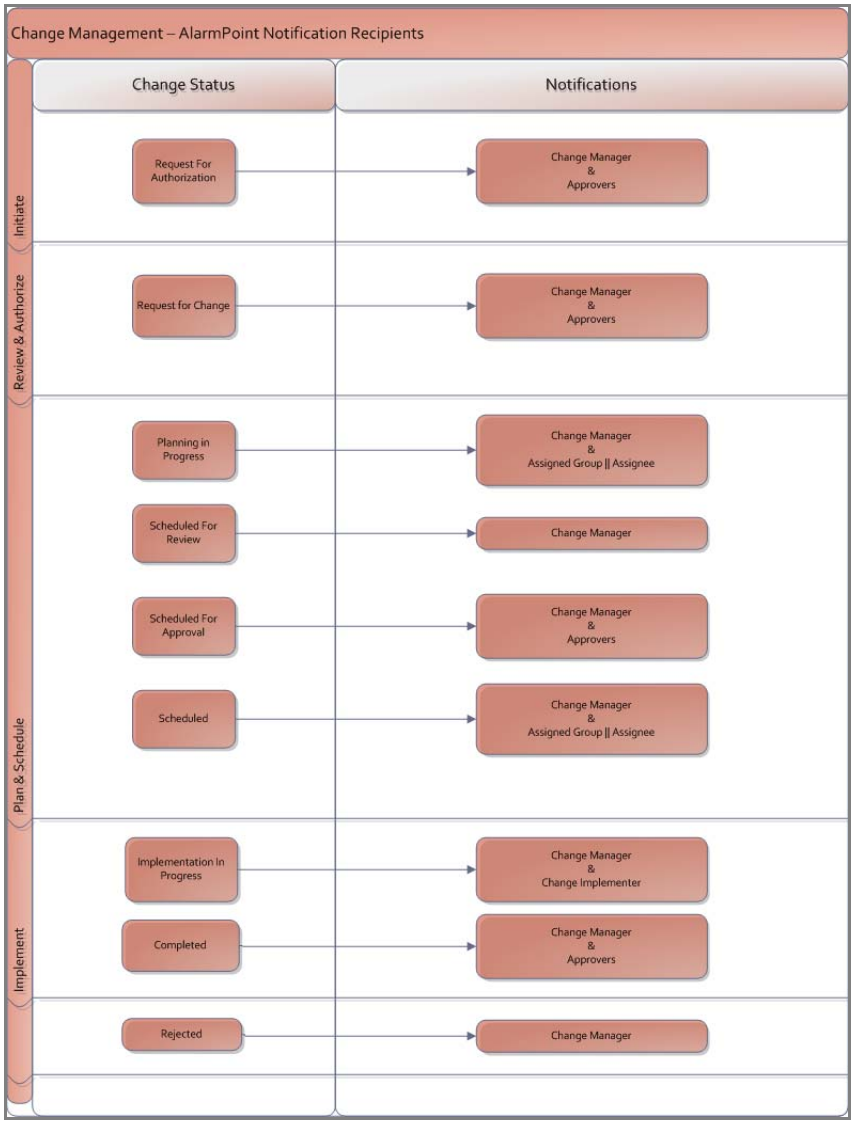
It is recommended that the applications be run in the following order:

- BMC Remedy Change Management
- AlarmPoint Java Client
- AlarmPoint Application and Notification Server Nodes

Consult the respective user manuals for details on starting these applications. The following sections will test the integration for notification delivery and response, User and Group Synchronization, and the Subscription Panel.

4.1 Notification Process

The following diagram illustrates which users receive notifications when a change request with a priority of critical or high has a particular change status (for more information about change transitions, refer to the BMC Remedy IT Service Management 7.5.00 Configuration Guide).



The Remedy filters installed with this integration that control these notifications begin with APCM and have the name of the change status as part of the name; for example, the filter that controls the sending of notifications to the Change Manager when a change request has a status of “Implementation in Progress” is called APCM:ImplementationInProgress_PUSH_ChangeManager.

4.2 Synchronizing a User and Group

The following validates that one-way communication from Remedy to AlarmPoint User and Group Synchronization is properly configured.

Note: *For this example, it is recommended that you set the text phone Device's User Service Provider to use the virtual text phone. This will help when troubleshooting problems in later testing.*

To test the User, Group and Membership Synchronization:

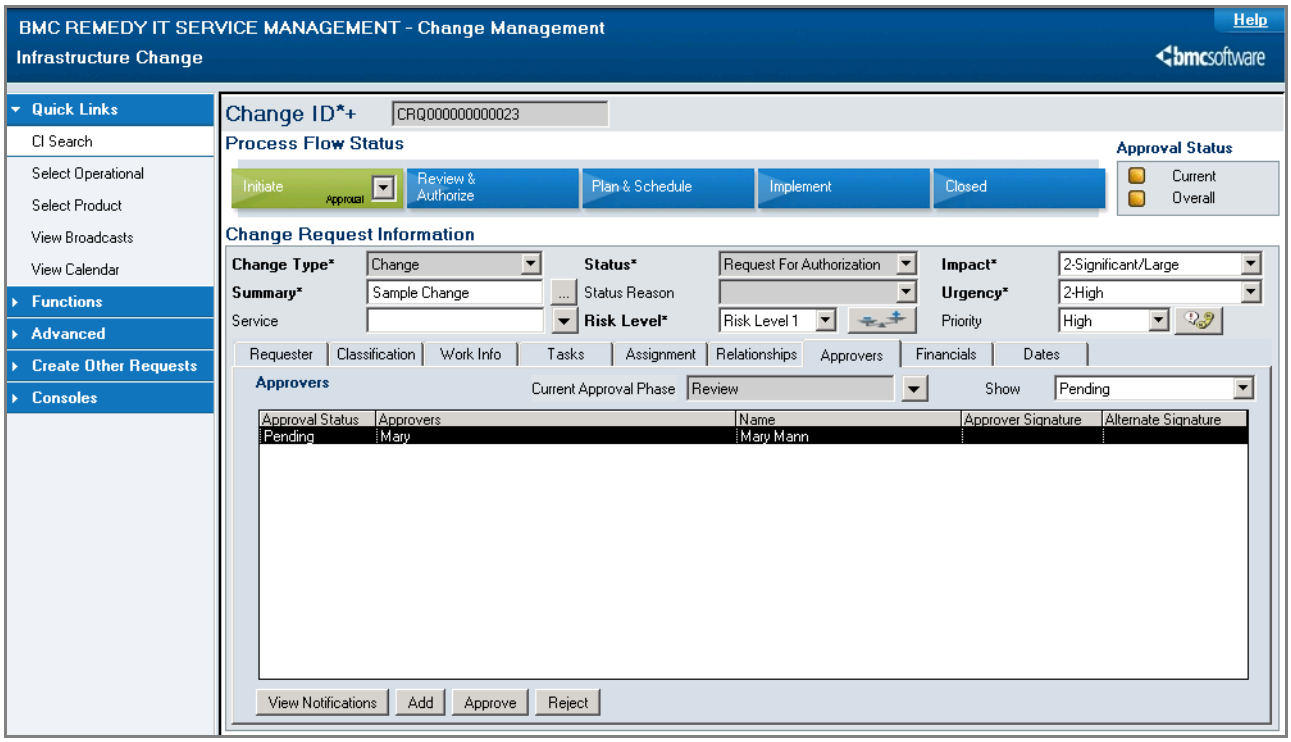
1. Launch the Application Administration Console, and then click **People** > **Create**.
2. In the Create a New Person dialog box, create a new Support User with a text phone Device.
3. Create a new Support Group and assign the newly created Support User to the Support Group.
   - To complete the test, see the following section.

4.3 Triggering a Notification

In this example, a Remedy Change Request will be injected to AlarmPoint for notification, targetting the User and Group configured in the previous section.

To trigger a notification:

1. In Remedy, create a new Change Request.
2. Click **Save**.
3. Add the Support User created in section 4.2, above, as an Approver to the Change Request.
   - If the Support User is not available, the synchronization may not be configured correctly.
4. Push the Change Request to an Approval Stage (e.g., “Request For Authorization”, “Scheduled For Approval” or “Completed”):



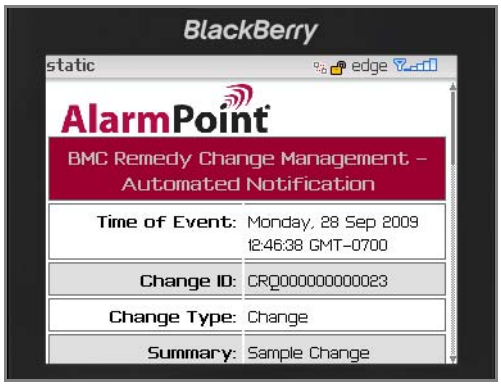
This should inject the request's parameters into AlarmPoint, triggering a new notification targeting the Change Manager and Approver you synchronized with AlarmPoint.

4.4 Responding to a Notification

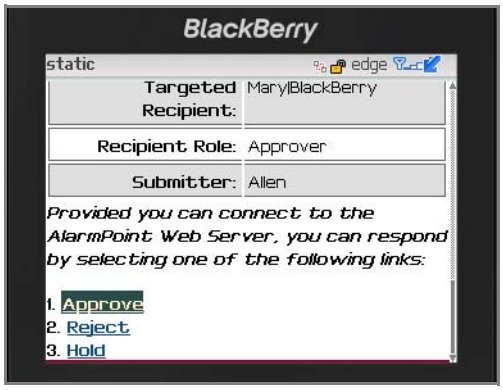
This section describes how to respond to a notification using a virtual BlackBerry:

1. When a notification arrives, the Device indicates the number of notifications that have been received:

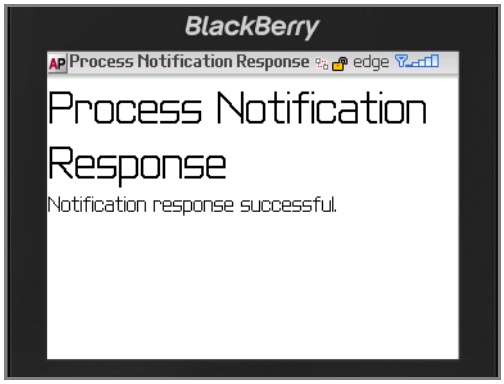
2. Selecting the notification displays the details of the incident:



3. Scroll down to view the details and the list of possible responses:



4. Send the response, and AlarmPoint will send an instruction to Remedy:



Note: *For information on adding or changing the available response choices, see "Response Choices" on page 40.*

4.5 View Request Results

When an action is taken on an AlarmPoint notification, that action is reflected in the Remedy Change Request. When AlarmPoint makes changes to a ticket, it also adds details to a Work Info entry on the change request.

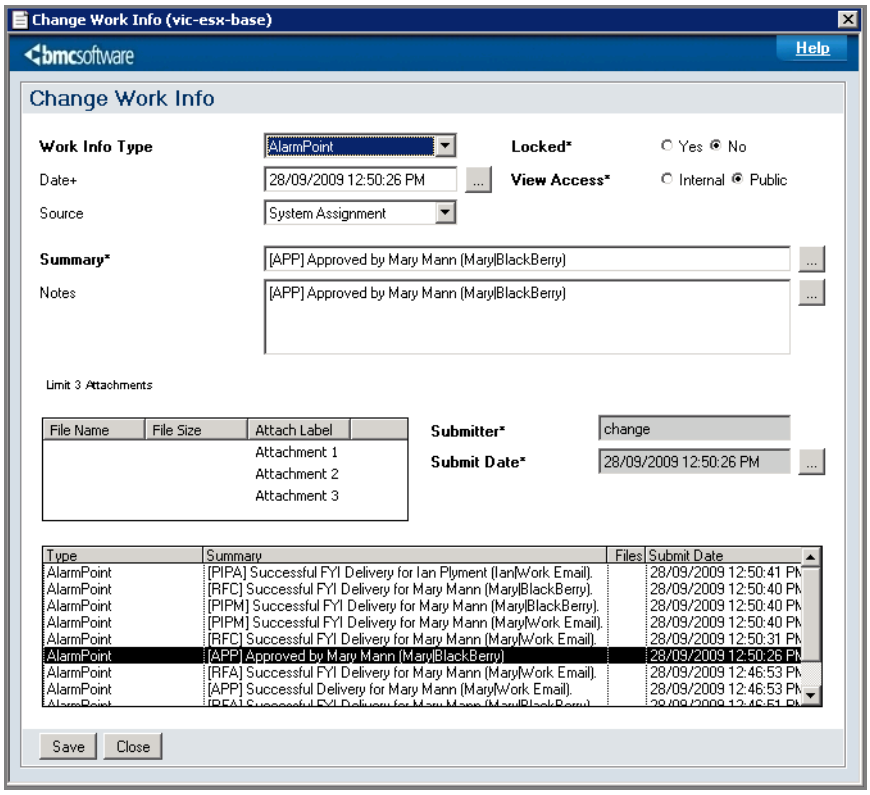
To view the updates:

1. In Remedy, click **File > Open > Object List**.

2. Double-click **Change Management Console**.
3. Under General Functions, select **Search Change**.
4. In the **Change ID** field, type the change ID of a ticket, and then click **Search**.
5. Select the change request, and then click the **Work Info** tab.

Note: *The change request will have entered the next state.*

6. Double-click an Annotation to display the Change Work Info dialog box:



For each User Device notified during this process, the Work Info entry will be annotated with a message indicating "Successful Delivery for Bob Smith - Text Phone" where "Bob Smith" is the recipient and "Text Phone" is the Device.

4.6 Testing the Subscription Panel

To test Subscriptions, ensure that Subscriptions are enabled in the Action Script Package with the $enable_subs variable. (For more information, see "Configuration Variable Reference" on page 46.)

Trigger a notification specifying the criteria that you configured your Subscription to match. You should receive an FYI-notification (informational only) which will not have any response choices available.

5. Optimizing and Extending the Integration

This section describes some of the available methods you can use to optimize or extend the AlarmPoint for BMC Remedy Change Management integration.

5.1 Adding Custom Data Elements

Additional data elements can be forwarded to AlarmPoint by adding them to the `bmcremedycm.xml`. The following steps explain how to add a new Event token to the Event injected to AlarmPoint. Note that all code samples in the following section are case-sensitive.

The following code represents the portion of the `bmcremedycm.xml` file that controls additional data elements:

```
/* To inject Custom Data Elements uncomment the following section and modify the
code to retrieve the desired parameter. Duplicate the code for each additional
parameter desired.
getParameter( "ap_param_name", "Web_Service_Param_Name", changeNode );
*/
```

To add an Event Token:

1. Open the `bmcremedycm.xml` file (by default, located in `APAgent\etc\integrations`).
2. Locate the section of code displayed above and copy the getParameter line of code to the bottom of the populateEvent method, immediately after the other getParameter calls.
3. Replace the following values with the appropriate names of the parameter you want to inject:
   - **ap_param_name**: the name of the parameter as displayed in AlarmPoint.
   - **Web_Service_Param_Name**: the name of the parameter as displayed in the Remedy Web Service Data Map.
4. Repeat steps 2 and 3 for each new custom parameter you want to inject.
5. Save and close the XML file.
6. Restart the AlarmPoint Java Client.

5.1.1 Adding custom parameters 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 Developer IDE and check out the BMC Remedy Change Management (BUSINESS) Script Package.
2. In the Presentation Action Script, add the following line to the email content creation section:

```
$content.message = $content.message & "TokenName: " & $event.tokenvalue & "\n"
```

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

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

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

Your custom parameter should now appear in your notification content for email Devices.

5.2 Adding Buttons to the Change Request Console

Using the default integration, you can add two buttons to the Change Request Console that allow interactions with AlarmPoint notifications:

- The “Halt AlarmPoint Notifications” button halts and clears all current notifications delivered to AlarmPoint for the current Change Request.
- The “Notify AlarmPoint” button redirects the user to the APCM:APAction form for manual notifications which are populated with information from the current Change Request.

To add the AlarmPoint buttons:

1. In the Server Window, click **Forms**, and then double-click the **CHG:Infrastructure Change** form.
2. Right-click the form, and then click **Button**.
3. Drag the button to a location on the form where you want the button to appear.
4. Click **Field Properties**, and then make the following changes:
   - On the **Display** tab, type `Halt AlarmPoint Notifications` in the **Button Label** field.
   - On the **Database** tab, type `zAPCMStopNtfnBtn` in the **Name** field.
   - On the **Permissions** tab, select **Public** from the list on the left side, and then click **Add**.
   - On the **Active Links** tab, select **(0) APINT:CHG:StopNotifications**, and then click **Add**.
5. Click **File > Save Form**.
6. Right-click the form, and then click **Button**.
7. Drag the button to a location on the form where you want the button to appear.
8. Click **Field Properties**, and then make the following changes:
   - On the **Display** tab, type `Notify AlarmPoint` in the **Button Label** field.
   - On the **Database** tab, type `zAPCMNtfnBtn` in the **Name** field.
   - On the **Permissions** tab, select **Public** from the list on the left side, and then click **Add**.
   - On the **Active Links** tab, select **(0) APCM:CHG:ManualMessage**, and then click **Add**.
9. Click **File > Save Form**.

5.3 Response Choices

This integration allows recipients to respond to notifications with several default choices, some of which are injected back to the Remedy server, updating the original change request. Users notified on email Devices also have the ability to respond with an extra annotation message which will be logged in the original Remedy change request.

The following is a list of the default response choices available with the integration and their associated actions on the AlarmPoint Event and the Remedy change request:

Response Choice	AlarmPoint Action	Remedy Update	Default Device Availability
Approve	Delinks all Users, not allowing them to submit responses. Any pending notifications for the Event will be terminated.	The change approver's status will update to Approved. If there are no other change approvers, the change request will enter the next state. Any additional notes added to the response are recorded on the change request's Work Info tab.	Email, BES and browser. For other non-FYI mobile Devices, an Approve is represented as "App".
Reject	Current notifications for the Event are delinked for all Users, not allowing them to submit responses. Any pending notifications for the Event will be terminated. A Reject notification is sent to the Event submitter.	The status for the change request is changed to Rejected and any additional notes added to the response are recorded on the change request's Work Info tab.	Email, BES and browser. For other non-FYI mobile Devices, a Reject is represented as "Rej".
Hold	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 holds the Event in AlarmPoint, and then approves the Event. They may also annotate the held Event.	The change approver's status is updated to Hold, and any additional notes added to the response are recorded on the change request's Work Info tab.	Email, BES and browser. For other non-FYI mobile Devices, a Hold is represented as "Hld".
Annotate	Halts delivery of notifications to any other Devices the responding User may have configured.	Any additional notes added to the response are recorded on the change request's Work Info tab.	This functionality is available for email Devices only.

5.3.1 Response choices for FYI notifications

FYI notifications do not have response choices, except for those FYI notifications sent to voice Devices. Voice FYI notifications offer the following response choices so Users can navigate between multiple notifications. (This navigation is not required on other Devices.)

Response	Description
Delete	Removes the notification from the User's list.
Save	Saves the notification and stops attempting to deliver it to the User's other Devices. Users may select this option to delay listening to the notification when it is delivered, and access the details by calling in, or via the AlarmPoint Web User Interface, at a later time.

Response	Description
Repeat	Repeats the notification.

5.3.2 Changing and adding response choices

The response choices and behavior can be changed in the response script in the Action Script Set (to change Subscription responses, update the subscriptionResponse script). Actions must be handled as requests in the `bmcremedycm.xml` Java Client file, located by default in `APAgent\etc\integrations\AP-BMC-Remedy-CM`.

For example, the following code illustrates the Approve response and all its components:

AlarmPoint Action Scripts:

Presentation Script

```
$content.choices = "Approve"
```

Response Script

```
# Handle responses
$reply = $response.reply
$reply::toLowerCase()
$approve= $reply::startsWith( "app" )
...
IF ( $approve )

 ...
    @ownRequest = @event::createExternalServiceMessage()
    @ownRequest.request_text = "Approve"
 ...
    $ownRequest.change_id = $event.change_id
    $ownRequest.approver = $event.user
    @ownRequest::send()
```

To ensure that the request_text passes the IF block located at the main routine for handling Remedy change request responses, include the following as part of the IF statement:

```
APDT_request_text.equalsIgnoreCase( "myRequest" )
```

To handle the Remedy update, the appropriate Web Service Action must be called from the Response Action Script. For more information, refer to the BMC Remedy Action Request System documentation.

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

5.3.3 Adding annotation messages to responses

Two-way email Device notifications (not FYI) can add extra annotations that will be added to the Remedy change request as a Work Info entry. 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 of response choices (above), and <Message> can be any string you want to add as the annotation.

5.4 Annotations

This integration extensively annotates the originating Remedy change request, but this may not be desirable in all environments. To add or remove the annotations to the Remedy change requests, edit the AlarmPoint Action Scripts.

All annotations are prefaced by a comment indicating that the following call is an annotation:

```
# Annotate CM Request
@someMessage = @event::createExternalServiceMessage()
$someMessage.request_text = "Annotate"
$someMessage.message_text = $message_note
$someMessage.change_id = $event.change_id
@someMessage::send()
```

The *$message_note* variable is the message to be annotated to the CM Request; specify the annotation message prior to the code block or replace $message_note with an annotation message.

To prevent the annotation of a Remedy change request, comment out the `@someMessage::send()` line as shown:

```
# @someMessage:send()
```

For additional annotations, use the above code example setting $message_note.

Note: *To disable all annotations, set the $main.annotate flag in the Configuration Section of the initial script to* false.

5.5 Altering the duration of Events

You can modify the amount of time AlarmPoint will send out notifications for a particular Event before it times out by changing the `$main.timeout` variable in the initial PROCESS script. This variable stores the number of seconds the notifications will be allowed to continue before timing out.

The default value is 86400, which is the number of seconds in a 24-hour period. You can change the delay to a two-hour timeout by changing the line to:

```
$main.timeout = 7200
```

5.6 FYI notifications

You can make all notifications informational only (i.e., the User is not offered any response choices). Setting the `$force_fyi` flag to “on” makes all normal and Subscription notifications one-way (FYI).

In the initial PROCESS script, locate the following line:

```
$force_fyi = disable
```

Change the line to:

```
$force_fyi = on
```

5.6.1 Generating FYI notifications for specific change requests

The FYI parameter is an optional parameter injected from Remedy into AlarmPoint. If it is set to “yes”, any notifications generated for the Event will be informational-only, and have no response choices.

To use this feature, in the `bmcremedycm.xml` file ensure the following line of code is executed within the main routine for handling the Change Request Input Action:

```
message.setDataTag( "fyi", "yes", "string" );
```

Note: *Within the script, you can choose to ignore the injected FYI variable to make an Event informational-only by setting the* `$force_fyi` *variable to "on" in the configuration section of the initial PROCESS script. For more information, see "Configuration Variable Reference" on page 46.*

5.6.2 Generating two-way notifications for Subscriptions

When using subscriptions to inform Users about change requests, you may want to enable responses from notifications generated for subscriptions.

To accomplish this, ensure that the configuration section of the initial PROCESS script has the following:

```
$subscription_fyi = false
```

By default, subscriptions are FYI notifications only. To provide responses for subscriptions if the $subscription_fyi flag is set to false, populate the Response Choices list within the Subscription Domain. The available response choices for approval notifications are Approve, Reject and Hold. Providing response choices for non-approval notifications will not change the FYI behaviour for these notifications.

The `$enable_subs` variable must also be set to true. See the section on configuration variables in the initial PROCESS script for details.

Note: *For more information about the variables in this section, see "Configuration Variable Reference" on page 46.*

5.7 Constructing BES and HTML Email Notifications

You can configure AlarmPoint to create BES and HTML email notifications. To enable BES and HTML email, the BMC Remedy Change Management (Business) script package set must be checked into the Developer IDE Database.

This feature requires the AlarmPoint Developer IDE. For installation instructions, refer to the *AlarmPoint Developer's Guide & Scripting Reference*.

Note: *Some email clients, such as Microsoft Outlook 2007, may not display HTML elements correctly. It is recommended that you test the HTML compatibility of your email client before implementing the HTML email feature.*

To enable BES and/or HTML email:

1. Launch the Developer IDE.
2. Check out the BMC Remedy Change Management (Business) Production script package.
3. In the Global Configuration Variables section of the initial PROCESS script, do the following:
   - Set `$main.enable_HTML_Email` to *true*.
   - Set `$main.use_logo` to true or false depending on whether you want your HTML email to show a logo.
   - Set `$AlarmPoint_URL` to the base URL of your AlarmPoint web server. (default is localhost).

Note: *HTML links will not redirect properly and logos will not be displayed if the $AlarmPoint_URL variable is not set to an external IP address or DNS name.*

4. Optionally, you can also do any of the following:

- Change `$main.HTML_form_url` to point to a JSP page that you want to process any responses from the HTML email. (the default setting should work out-of-the-box).
- Change `$main.logo` to a URL that holds the image you want to display at the top of HTML emails (by default, it points to the AlarmPoint logo).
- Set `$main.logo_alt_text` to the text you wish to display when the logo cannot be fetched. This can be displayed if the email client is configured not to show images, or it could be displayed because the email client cannot access the AlarmPoint Webserver directly and thus cannot respond by using the links in the HTML.
- If you are using BES and have access to a BES server, you can set the URL to the BES server in the `$main.bes_pushurl` variable.

5. Save and validate the script, and check in the script package.

Note: *For more information about these and other configuration variables, see “Configuration Variable Reference” on page 46.*

5.8 Java Client Logging

All integration errors can be logged to log files, as described in “Installing the Logging Configuration File” on page 7. To enable additional logging, use the following code within the `bmcremedycm.xml` file, located by default in `APAgent\etc\integrations`:

```
util.getLogger().error("My Log Message");
```

Note: *The utility must be initialized before it is available to use.*

5.9 Uninstalling

For instructions on removing an AlarmPoint deployment, refer to the *AlarmPoint Installation and Administration Guide*.

6. Configuration Variable Reference

This section outlines and describes the configuration variables available in the initial PROCESS AlarmPoint 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.

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**	Configures 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.
$subscription_fyi = true	Forces Subscription notifications to be informational only; recipients of a Subscription notification will not be able to respond to the Event. **Note**: If the `$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 "Creating a Fail-Safe Group" on page 29.*

Variable	Description
$fail_safe = "for-recipients"	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 = "Remedy 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.
$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 is unnecessary, but you will lose any information about whether a delivery was successful.

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	Specifies the amount of time (in seconds) the Event is allowed to run before timing out. (86400 seconds = 24 hours.)

Variable	Description
$main.debug = false	Indicates whether to log informational messages for debugging purposes. Disabling this variable may improve performance, but will provide less information.
$main.use_logFile = false	Specifies whether to use an alternate log file for debugging messages. This variable is ignored unless `$main.debug` is also set to true.
$main.logFile = "../logs/ BMC_Remedy_CM_Script.log"	Defines the file used to log debugging information (only if `$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 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.enable_reject = true	Specifies whether the "Reject" response is available.
$main.subscription_annotate = false	Enables submission of Subscription information back to the Management System. (As with `$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 Webserver'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.

Variable	Description
$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.]"	Specifies 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 is valid and responses will not be injected from HTML Devices (email and BES).
$main.numeric_pager_number = "555-1212"	Specifies 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 has occurred.
$main.bes_pushurl = "http://localhost:8888/static"	Specifies the URL of the BES server. (Optional.)

7. Contacting AlarmPoint

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, and other 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

E-mail: support@alarmpoint.com
Website: http://www.alarmpoint.com

Customer Support Site:
https://connect.alarmpoint.com

BMC Software, Inc.

1030 West Maude Avenue
Sunnyvale, CA 94085-2810

Phone: 408-571-7000
Fax: 408-571-7001

8. Copyright

AlarmPoint Systems, Inc. produced this integration document to assist customers with joint BMC Software / AlarmPoint Systems implementations. BMC Software and AlarmPoint Systems have 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.