xMatters *(alarmpoint)* engine

(CA SPECTRUM Network Fault Manager)

(x) matters®
The relevance engine company.

This manual provides information about xMatters. Every effort has been made to make it as complete and accurate as possible; however, the information it contains is subject to change without notice and does not represent a commitment on the part of xMatters No part of this document may be reproduced by any means without the prior written consent of xMatters

Monday, June 21, 2010

Contacting xMatters

You can visit the xMatters Web site at: http:///www.xmatters.com

From this site, you can obtain information about the company, products, support, and other helpful tips. You can also visit the Customer Support Site from the main web page. In this protected area, you will find current product releases, patches, release notes, a product knowledge base, trouble ticket submission areas and other tools provided by xMatters, inc.

xMatters, inc.
Corporate Headquarters
4457 Willow Road, Suite 220
Pleasanton, CA 94588

Sales and Technical Support:

Telephone: 925-226-0300
Facsimile: 925-226-0310

support@xmatters.com
sales@xmatters.com
Customer Support Site: http://connect.xmatters.com

This integration was designed and tested on an unmodified version of CA SPECTRUM Network Fault Manager, and this document describes how to configure xMatters to integrate with the default installation. If you have customized or altered your instance of CA SPECTRUM, this integration may need to be modified for your deployment. Please note that these integration changes are not part of the services offered by xMatters Technical Support, but can be performed through the xMatters Professional Services department. For more information, contact your xMatters Sales representative.

Table of contents

Introduction

Welcome to xMatters (alarmpoint) engine for CA SPECTRUM Network Fault Manager. This document describes how to install and configure the xMatters (alarmpoint) engine for CA SPECTRUM Network Fault Manager software integration. The intended audience for this document is experienced consultants, system administrators and other technical readers.

Summary

xMatters (alarmpoint) engine 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.

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

Using this integration, xMatters can become the voice and interface of an automation engine or intelligent application (the management system, such as CA SPECTRUM Network Fault Manager). When CA SPECTRUM detects something that requires attention, xMatters places phone calls, sends pages, messages, or emails to the appropriate personnel, vendors or customers.

xMatters is also persistent, escalating through multiple devices and personnel until someone accepts responsibility or resolves the problem. Once contacted, xMatters gives the notified person instant two-way communication with CA SPECTRUM Network Fault Manager. Responses are executed immediately on CA SPECTRUM, enabling remote resolution of the event.

This integration supports alarm notifications (from CA SPECTRUM to xMatters) by adding functionality to the CA SPECTRUM AlarmNotifier scripts which communicate via the xMatters integration agent. It also supports inbound actions (from xMatters to CA SPECTRUM), through the use of the Command Line Interface (CLI) tool, allowing users to own, acknowledge, and clear alarms.

You will need to modify this configuration to suit your particular business requirements and adjust it to suit your expected loads. For example, the default integration features automatic status annotations to the original event; in a high-volume production system, this can significantly affect performance. Consider your expected volume of injected events and your server capacity when combining your CA SPECTRUM deployment with xMatters.

Benefits

With the xMatters integration, the appropriate technician can be notified directly via voice, email, pager, BlackBerry, or other device. Information about the event will be presented to the event resolver and decisions can be made in real-time.

Once a response is selected on the recipient's remote device, xMatters will update the CA SPECTRUM alarm 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 state of the event.

During the process, every notification, response, and action is logged in xMatters. In addition, xMatters automatically annotates the original CA SPECTRUM alarm with status information.

The xMatters product features a self-service web user interface to allow accurate assignment of responsible personnel for each job. xMatters also includes an enhanced Subscription panel that allows both managed and self-subscription to CA SPECTRUM alarms. This Subscription panel queries the CA SPECTRUM Server directly in real time to retrieve lists of important settings and criteria, removing the need to create and maintain these lists.

Integration architecture

The software components in this integration include:

- CA SPECTRUM Network Fault Manager
- xMatters (alarmpoint) engine
- xMatters integration agent

The following diagram illustrates the software processes used by this integration:
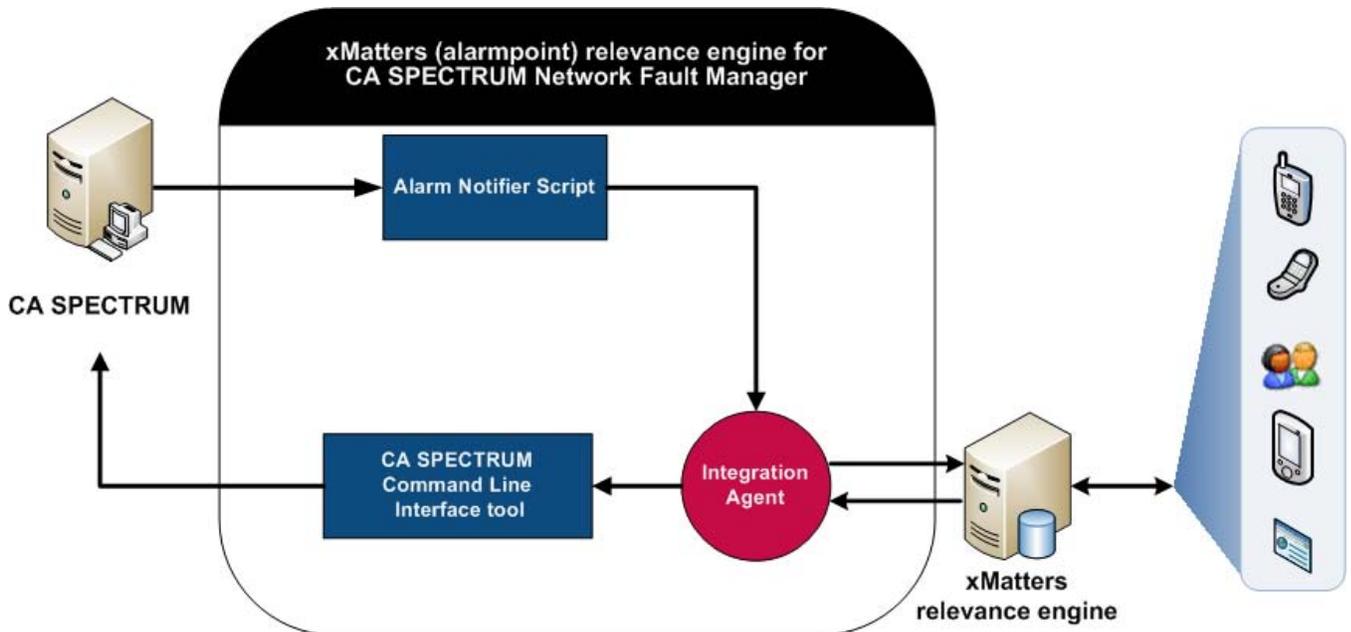
When a CA SPECTRUM event is detected, it triggers the following steps:

1. CA SPECTRUM injects the event to xMatters using a Command Line Interface (CLI) call to the integration agent via CA SPECTRUM the Alarm Notifier.
2. When the recipient responds to the notification, the response updates CA SPECTRUM using a CLI call via the integration agent.

System requirements

The following products must be installed and operating correctly prior to integration:

- xMatters (alarmpoint) engine 4.1 (patch 002 or later)
- xMatters integration agent 4.1
- xMatters Developer IDE
- CA SPECTRUM Network Fault Manager r9.1.2

Operating systems

The following operating systems are supported by this integration:

- Microsoft Windows 2003 (validated)

Conventions and terminology

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

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
- code samples

Directory paths

Except where explicitly stated, the directory paths in this document are listed in Windows format. Unix users must substitute the given paths with the Unix equivalents.

The xMatters installation folder is referred to throughout the documentation as <xMHOME>.

- On Windows systems, the default is `C:\Program Files\AlarmPointSystems\AlarmPoint\`
- On Unix systems, the default is `/opt/alarmpointsystems/alarmpoint/`

The xMatters integration agent installation folder is referred to throughout the documentation as <IAHOME>.

- On Windows systems, the default is `C:\Program Files\AlarmPointSystems\IntegrationAgent\`
- On Unix systems, the default is `/opt/alarmpointsystems/integrationagent`

Terminology

The following terms are used through the xMatters documentation.

Documentation terminology

Term	Meaning
Event	An *event* refers to any situation or item of interest detected by the management system, and which requires attention. Event is also used to refer to the incident or situation as it progresses through the xMatters system, from injection to notification to resolution. Each event must generate at least one alert or notification.
	Event can also be a generic term used to refer to an incident, change request, message, or other specific item within the management system. Different management systems use different terminology (CA SPECTRUM uses the term "alarm"), but xMatters treats all of these items as events.
Management system	A *management system* is any sort of monitoring or managing software that watches for events, and with which xMatters can combine; i.e., a synonym for CA SPECTRUM.
Device	A *Device* is the medium through which a recipient is contacted by xMatters; i.e., email, pager, phone, BlackBerry, etc.

Term	Meaning
User	In xMatters, people who can receive notifications are called *Users*. Each person in the xMatters system is defined by a set of User details, including ID number, user name, login password, and so on.
Group	*Groups* are used to collect and organize Users and Devices into notification schedules. For a complete explanation of Groups in xMatters, see the *xMatters User Guide*.

Installation and configuration

This chapter provides information about installing the xMatters (alarmpoint) engine for CA SPECTRUM Network Fault Manager integration. This chapter also contains complete instructions on how to configure xMatters, CA SPECTRUM, and the integration components.

The following checklist provides a complete list of the high-level steps required to combine xMatters with CA SPECTRUM:

Installing the xMatters (alarmpoint) engine

> **Installing the xMatters integration agent**

Installing CA SPECTRUM Network Fault Manager

Installing the integration

> **Running the installer file**
>
> > Integration components

Configuring CA SPECTRUM

> **Modifying the CA SPECTRUM Alarm Notifier scripts**

Configuring xMatters

> **Importing the script package**
>
> **Configuring the Event Domain**
>
> > Defining an Integration Service
> >
> > Defining Event Domain Constants
>
> **Configuring the default User**
>
> **Configuring the Subscription panel**
>
> > Defining Event Domain predicates
> >
> > Defining a Subscription Domain
> >
> > Configuring the Subscription JSP
> >
> > Creating a Subscription
> >
> > Creating a fail-safe Group

Installing the xMatters (alarmpoint) engine

Consult the xMatters Installation and Administration Guide and the xMatters Integration Agent Guide for installation instructions and details. When installing xMatters, note that you must select the option to install the xMatters web server.

Note: *You will also need to install the xMatters Developer IDE, as described in the* xMatters Online Developer's Guide.

Installing the xMatters integration agent

The xMatters integration agent must be installed on the same machine as CA SPECTRUM Network Fault Manager as the integration agent manages the communication between xMatters and CA SPECTRUM using locally-run CLI commands.

The integration agent service must be configured to log in as a user with sufficient permissions to update CA SPECTRUM events with notification responses using the CA SPECTRUM CLI. Typically, the user and credentials used to run the CA SPECTRUM Server is also used as the integration agent service user.

On Windows, configure the xMatters integration agent service using the following steps:

1. Open the Windows Services tool.
2. Right-click the **xMatters integration agent** service, and then select **Properties**.
3. Click the **Log On** tab, and then click **This account**.
4. Configure the User and Password credentials with a Windows User with sufficient permissions to update CA SPECTRUM alerts (e.g. `.\Administrator`).
5. Click **OK** and restart the integration agent service.

Installing CA SPECTRUM Network Fault Manager

For information on installing CA SPECTRUM, consult the CA SPECTRUM Network Fault Manager documentation.

Installing the integration

This section describes the installation process for the xMatters (alarmpoint) engine for CA SPECTRUM Network Fault Manager integration. For information on how to install the integration components manually, see "Manual installation" on page 28.

Running the installer file

This integration includes an installer file that automatically copies the required files and folders into their appropriate locations in the xMatters (alarmpoint) engine and integration agent installation folders.

In addition to installing the appropriate files, the installer performs the following tasks:

- Creates a log file `caspectrum-install.log`, containing a detailed list of the files installed during the install process. (If you encounter any errors during the installation, include this file when contacting Support.)
- Creates a folder named `caspectrum_install_<date and time>` containing:
    - A folder named `integration` that contains all of the integration files and components required for the integration. This folder can be used to perform a manual installation if required, as described in "Manual installation" on page 28.
    - An `installer.properties` file containing all of the environment variable that existed when the installer was run. (If you encounter any errors during the installation, include this file when contacting Support.)
    - A folder named `backup` containing copies of any existing Subscription JSP files in the system.

To install the integration files:

1. Extract the contents of the `xM-CA-SPECTRUM_<version>.zip` file into both the `<xMHOME>` and the `<IAHOME>` folders.
    - The archive file contains two files, `caspectrum_install.bat` and `caspectrum_install.jar`.
2. Open a command line window and navigate to the `<xMHOME>` folder.
3. Run the `caspectrum_install.bat` file.
    - The installer will prompt you to confirm the installation:

4. Navigate to the `<IAHOME>` folder.

5. Run the `caspectrum_install.bat` file.

   - The installer will prompt you to confirm the installation.

6. Open the `caspectrum.js` file found in `<IAHOME>\integrationservices\caspectrum` and modify the paths within the "initvariables()" method to match your CA SPECTRUM installation.

7. Open the `IAConfig.xml` file found in `<IAHOME>\conf\` and add the following line to the "service-configs" section:

`<path>caspectrum/caspectrum.xml</path>`

8. To enable the logging of user annotations and notification delivery annotations to `<IAHOME>\logs\AlarmPoint_Notification.txt`, backup the `<IAHOME>\conf\log4j.xml` file and then replace it with the file provided at:

`<IAHOME>\integrationservices\caspectrum\log4j.xml`.

9. Restart the xMatters web server and the integration agent service.

Integration components

The following table describes some of the notable integration components:

Component Name	Description
com.alarmpoint.spectrum.jar	Contains the following libraries: - **CLI Manager**: Java code to communicate to CA SPECTRUM using the CLI, and to transform the responses into Java objects that can be processed by the scripts and subscription panel. - **Integration Agent Service Tag Library**: Java tag library and supporting code to allow the subscription panel to communicate to the integration agent. - **Subscription Panel Tag Library**: Java tag library to produce the subscription panel.
SpectrumSubscriptionForm.jsp	Custom Subscription Panel that allows users to subscribe to events associated with specific criteria, such as a particular source or severity.
SpectrumSubscriptionSubSearch.jsp	Custom Subscription popup JSP that allows users to limit the number of records that appear for a predicate in a subscription.
xM-CA-SPECTRUM.aps	Contains the scripts required for this integration.
caspectrum.xml, caspectrum.js	The JavaScript and XML service configuration files that define the service on the integration agent.

Configuring CA SPECTRUM

Configuring CA SPECTRUM to combine with xMatters requires the following steps:

- Modify the CA SPECTRUM notifier scripts for xMatters.

Modifying the CA SPECTRUM Alarm Notifier scripts

To configure the Alarm Notifier scripts, you must add a reference to the APClient.bin.exe tool that injects the CA SPECTRUM events into the integration agent.

To modify the CA SPECTRUM Alarm Notifier scripts:

1. On the CA SPECTRUM server, navigate to the `C:\win32app\SPECTRUM\Notifier\` directory.
2. Open the `ClearScript` file in a text editor.
3. Add the following code to the end of the file:

```
######################################################################
# xMatters (alarmpoint) engine
######################################################################

if [ "$SEV" = "CRITICAL" -o "$SEV" = "MAJOR" ]
  then
    echo "Injecting AlarmPoint event for cleared SPECTRUM AlarmID:" $AID
    "C:/Program Files/AlarmPointSystems/IntegrationAgent/bin/APClient.bin.exe" \
    --map-data caspectrum "" "$AID" "$RAW_ALARM_TIME" "$DTYPE" "$MTYPE" "$MNAME" "$AID" \
    "$SEV" "$CAUSE" "$REPAIRPERSON" "$STATUS" "$SERVER" "$LANDSCAPE" "$MHANDLE" "$MTHANDLE" \
    "$IPADDRESS" "$SECSTR" "$ALARMSTATE" "$ACKD" "$CLEARABLE" "$AGE" "Cleared Event" \
    "$EVENTMSG" "CLEARED EVENT" "CLEAR" "$CLEARED_BY_USER_NAME"
  else
    echo "Not injecting AlarmPoint event for cleared SPECTRUM AlarmID:" $AID
fi
```

4. Open the `C:\win32app\SPECTRUM\Notifier\SetScript` file in a text editor.
5. Add the following code to the end of the file:

```
######################################################################
# xMatters (alarmpoint) engine
######################################################################
# Parse Alarm Title info

if [ "$SEV" = "CRITICAL" -o "$SEV" = "MAJOR" ]
  then
    ALARMTITLE=`echo $PCAUSE | awk '{ print substr($0,1,index($0,"SYMPTOMS:") - 2) }'`
    echo "Injecting AlarmPoint event for SPECTRUM AlarmID:" $AID
    "C:/Program Files/AlarmPointSystems/IntegrationAgent/bin/APClient.bin.exe" \
    --map-data caspectrum "" "$AID" "$RAW_ALARM_TIME" "$DTYPE" "$MTYPE" "$MNAME" "$AID" \
    "$SEV" "$CAUSE" "$REPAIRPERSON" "$STATUS" "$SERVER" "$LANDSCAPE" "$MHANDLE" "$MTHANDLE" \
    "$IPADDRESS" "$SECSTR" "$ALARMSTATE" "$ACKD" "$CLEARABLE" "$AGE" "$PCAUSE" "$EVENTMSG" \
    "$ALARMTITLE" "SET" ""
  else
    echo "Not injecting AlarmPoint event for SPECTRUM AlarmID:" $AID
fi
```

6. Open the `C:\win32app\SPECTRUM\Notifier\.alarmrc` file in a text editor.
7. Locate the GET_EXISTING_ALARMS parameter, and set its value to false.
   - If you do not change the GET_EXISTING_ALARMS parameter, CA SPECTRUM will attempt to notify any existing alarms in the system, regardless of whether they have already been notified, whenever you run the AlarmNotifier.exe command.
8. In the command prompt, run the `C:\win32app\SPECTRUM\Notifier\AlarmNotifier.exe` command.

Configuring xMatters

Configuring xMatters to combine with CA SPECTRUM requires the following steps:

- Importing the script package
- Configuring the Event Domain
- Configuring the default User
- Configuring the Subscription panel

Importing the script package

This step requires the xMatters Developer IDE. For installation instructions, refer to the *xMatters Online Developer's Guide*.

To import the xMatters Script Package:

1. Launch the IDE, and then configure the database connection.
2. Click **Workspace > Import**.
3. Select the xM-CA-SPECTRUM.aps file extracted from the integration installer into the following directory:

   `<xMHOME>\caspectrum_install_<date_and_time>\integration\components\alarmpoint\scripts\`

4. In the File dialog box, click **OK**, and then click **OK** again.
5. Right-click the **CA SPECTRUM Network Fault Manager (BUSINESS)** folder, and then select **Validate**.
6. Right-click the **CA SPECTRUM Network Fault Manager (BUSINESS)** folder, and then select **Check In**.
7. In the Create Script Package dialog box, click **Create**.
8. In the Check In dialog box, click **Close**.

Configuring the Event Domain

By default this integration is set up to use an Event Domain of "caspectrum"; it is strongly recommended that you use this default Event Domain. For the integration to be successful, the Event Domain name must match the domain variable specified in the caspectrum.xml file installed on the integration agent.

Note: *The xMatters web server must be running to perform this portion of the integration.*

To define an Event Domain:

1. Sign on to xMatters as a Company Administrator, and click the **Developer** tab.
2. In the Developer menu on the left side of the screen, click **Event Domains**.
3. On the Event Domains page, click **Add New**.
4. Enter the following information into the form:
   - **Name**: caspectrum
   - **Description**: CA SPECTRUM Integration
   - **Script Package**: CA SPECTRUM Network Fault Manager
5. Click **Save**.

Defining an Integration Service

This integration uses a default integration service of "caspectrum"; it is strongly recommended that you use this default integration service. For the installation to be successful, the integration service name must match the service

specified in the caspectrum.xml file installed on the integration agent.

To define an integration service:

1. In xMatters, on the Event Domains page, click the **caspectrum** Event Domain.
2. On the Event Domain Details page, in the Integration Services area, click **Add New**.
3. Enter the following information into the form:
   - **Name**: caspectrum
   - **Description**: CA SPECTRUM Integration Service
4. Click **Save**.

Defining Event Domain Constants

Company Administrators and Developers can create Event Domain Constants that will be available in scripting for all event objects associated with an Event Domain. This integration uses Event Domain Constants to define custom values for the integration script package.

The integration script package uses the names of the constants defined in the table below to look up the values; it is strongly recommended that you use the names specified. Note that the values for the **alarmpointurl** and **bespushurl** constants should be modified to specify the address of the xMatters web server (to enable the HTML response options) and the BES device server.

To add an Event Domain Constant:

1. In xMatters, click the **Developer** tab, and then, in the menu on the left side of the screen, click **Event Domain Constants**.
2. In the **Event Domain** drop-down list, select **caspectrum**.
3. On the Event Domain Constants page, click **Add New**.
4. Define a **Constant Name**, **Value**, and **Description** for the new constant, according to the table below.
5. Click **Save**.
6. Repeat the above steps for each of the constants you want to add.
   - Note that if the constants are not defined in the web user interface, the scripts will use the values listed in the Default Value column of the following table.

Event Domain Constants

Constant Name	Default Value	Description
alarmpointurl	http://localhost:8888	Used to specify the address of the xMatters web server. The links provided in notification content use the alarmpointurl constant value to locate the xMatters web server which would process the response. For these links to work, this address must be reachable from the device where the user will receive the notification; normally, this is the IP address or fully-qualified host name of the xMatters web server.
bespushurl	http://localhost:8888/static	Used to specify the address of the BES device server. Populates the $main.bes_pushurl parameter.

Constant Name	Default Value	Description
forcefyi	disable	Force notifications to be informational only (FYI), rather than requiring responses; this overrides the fyi behaviour specified on the injected event. Possible values: **disable**: Nothing is forced. **on**: Notifications are forced to be FYI. **off**: Notifications are forced not to be FYI. Populates the force_fyi parameter.
failsafegroup	CA SPECTRUM FailSafe	The fail-safe recipient to notify, typically a group. The fail-safe group identifies the recipient that will be notified if an event is injected to the xMatters (alarmpoint) engine and no subscriptions exist that match the event. Set this constant if you want to change the failsafe group from CA SPECTRUM FailSafe to another group defined in xMatters.
failsafe	enabled	Controls fail-safe functionality, notifying the fail-safe recipient via EMAIL under certain circumstances; populates the $fail_safe parameter. Possible values: **enabled**: Notify if no subscriptions match or no notifiable recipients. **for-subscriptions**: Notify if subscription functionality is enabled AND no subscriptions match. **for-recipients**: Notify if no notifiable recipients. **disabled**: Disable fail-safe functionality.
overridetimeframes	false	Override Recipients Device Timeframes. Populates the $override_timeframes parameter.
useemergencydevices	false	Force the use of emergency Devices. Populates the $use_emergency_devices parameter.
trackdelivery	true	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 successful or not. Populates the $track_delivery parameter.
annotate	true	Enables submission of annotations back to the management system. Populates the $main.annotate parameter.
subscriptionannotate	true	Enables submission of Subscription annotations back to the management system. Populates the $main.subscription_annotate parameter.

Constant Name	Default Value	Description
tracksubscriptiondelivery	true	Track when each device is delivered to for Subscriptions. Populates the `$track_subscriptionDelivery` parameter.
timeout	259200	Amount of time (in seconds) the event is allowed to run before timing out. 259200 seconds = 72 hours. Populates the `$main.timeout` parameter.
maxinvalidresponses	3	Specifies the maximum number of invalid responses allowed before notification is no longer requeued. Populates the `$main.maxInvalidResponses` parameter.
enablehtmlemail	true	Enables HTML email functionality. Populates the `$main.enable_HTML_Email` parameter.
uselogo	true	Set this if you want the logo displayed within HTML email notifications. Populates the `$main.use_logo` parameter.

Note: *Event Domain Constant values are case-sensitive; Boolean values must be lowercase* "true" *or* "false"*. For more information about the parameters referenced in the Description column, see "Configuration variable reference" on page 31.*

Configuring the default User

By default, this integration uses a default demonstration User named "bsmith". Follow the steps below to ensure that this User has a virtual two-way text phone Device .

To configure the default User:

1. In xMatters, click the **Users** tab.
2. On the Find Users page, click **S**.
3. In the list of returned Users, click **Smith, Bob**.
4. In the Common Tasks pane, click **User Devices**.
5. Verify that a virtual text phone Device 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 the User ID "bsmith", and add a virtual text phone Device. For more information and instructions on how to perform these tasks, refer to the* xMatters User Guide.

Configuring the Subscription panel

This integration includes a Subscription panel which reads list values for Event Domain predicates from CA SPECTRUM using the command-line interface (CLI) via the integration agent. This feature allows administrators to change the source of the content supplied for these lists from CLI calls to predefined predicate value lists.

To allow Users to subscribe to specific criteria on injected events, you must configure the Subscription panel, which 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: *The Subscription panel file is copied to the correct directory during the integration installation, as described in "Installing the integration" on page 6.*

Defining Event Domain predicates

The default Subscription panel provided with the integration requires the following Event Domain predicates:

- IPADDRESS
- ALARMTITLE
- SEV
- PCAUSE
- MNAME
- DTYPE
- MTYPE

Note: *You can also use the following steps to add other predicates that you consider vital and which you plan to add to the integration. For more information, see "Adding new parameters" on page 25.*

To define the Event Domain predicates:

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

Event Domain predicates

Predicate	Type	Important	Values	Description
IPADDRESS	Text	Yes	Manually defined	Network address
ALARMTITLE	Text		Manually defined	Alarm title
SEV	List	Yes	Manually defined	Severity; the default values are: - Critical - Inital - Maintenance - Major - Minor - Suppressed
PCAUSE	Text		Manually defined	Probable cause
MNAME	List	Yes	Automatically generated	Model name

Predicate	Type	Important	Values	Description
DTYPE	Text		Manually defined	Device type
MTYPE	List		Automatically generated	Model type name

Defining a Subscription Domain

The Subscription Domain is the reference point of the optional Subscription panel and allows you to control who can create Subscriptions, how recipients can respond to Subscription notifications, and which Event Domain predicates can be used to create a Subscription. You must create a Subscription Domain before you can create Subscriptions with the new panel.

To create a Subscription Domain:

1. On the Developer tab, in the Developer menu, click **Subscription Domain**s.
2. On the Subscription Domains page, click **Add New**.
3. In the **Event Domain** drop-down list, select **caspectrum**, and then click **Continue**.
4. On the Subscription Domain Details page, in the **Name** field, type caspectrum.
   - By default, Subscriptions are non-FYI (i.e., they support response options). To disable two-way Subscription notifications, select the **One-Way** check box.
5. In the **Type of Management** drop-down list, select **Both**.
   - This allows you to assign Subscriptions to Users, and also allows Users to create their own Subscriptions.
6. In the **Custom Page URL** field, enter the following path:

jsp\subscription\caspectrum\SpectrumSubscriptionForm.jsp

7. Click **Continue**.
8. On the Select Appropriate Response Choices page, specify the available responses for this Subscription, and then click **Continue**.
   - By default, the scripts support the following response choices: "Acknowledge", "Ignore", "Clear", and "Annotate". To enable two-way communications for Subscriptions, define all four response choices on the Select Appropriate Response Choices page. If you require only one-way, informational notifications, do not specify any response choices.
9. On the Select Appropriate Predicates page, add all of the predicates to the **Applied Predicates** list, and then click **Continue**.
10. On the Select Roles page, specify the Roles you want to be able to create Subscriptions on the Domain, and then click **Save**.

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

Configuring the Subscription JSP

You can use one of the following methods to populate the predicate list values on the Subscription Panel:

- Manually specify the predicate list values in the web user interface.
- Using the CLI tool, query CA SPECTRUM for possible values, and automatically populate the predicate lists with the results of the query.

| **Note:** | *Changing Subscription Domains 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* xMatters Installation and Administration Guide. |

Specifying predicate lists manually

To configure the Subscription panel in a demo mode, using predefined predicate list values, you must modify the Subscription JSP.

To manually populate the predicate lists:

1. Open the `SpectrumSubscriptionForm.jsp` found in the `<xMHOME>\webserver\webapps\cocoon\alarmpoint\jsp\subscription\caspectrum` folder on the xMatters web server install.
2. Locate the `<subs:predicate>` tag for each of the predicates that are populated automatically (MNAME and MTYPE).
3. For each predicate, set the `performSearch` attribute to `false`.
4. Save and close the `SpectrumSubscriptionForm.jsp` file.
5. In xMatters, click the **Developer** tab.
6. On the Event Domains page, click **caspectrum**.
7. On the Event Domain Details page, in the Predicates list, click the name of the predicate for which you want to define the values.
8. Add to the predicate list values, and then click **Save**.
9. Repeat steps 6 and 7 for each predicate to which you want to assign values.

The predicate lists on the Subscription will now be populated with the predefined list values instead of the data returned from CA SPECTRUM.

Populating predicate lists automatically

If you want to populate the predicate values lists automatically from CA SPECTRUM rather than by predefining the predicate list values, you must configure the JSP file.

To configure the Subscription JSP to retrieve predicate list values:

1. Open the `SpectrumSubscriptionForm.jsp` found in the `<xMHOME>\webserver\webapps\cocoon\alarmpoint\jsp\subscription\caspectrum` folder on the xMatters web server install.
2. Locate the `<subs:predicate>` tag for the two predicates that you want to have populated automatically (MNAME and MTYPE).
3. For each predicate, set the `performSearch` attribute to `true`.
4. Save and close the JSP.

Creating a Subscription

You can now use the custom Subscription Panel to subscribe to CA SPECTRUM events that match specific criteria. For example, you could configure a subscription that would send a notification to a specific User each time an event entered the system that was of critical severity.

To create a Subscription:

1. On the Alerts tab, in the Alerts menu, click **My Subscribed Alerts**.
2. Select the caspectrum 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 tabs.
4. When you are satisfied with the criteria, click **Save** to create the Subscription.

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

- Click **Select Model Name** to open a search window that allows you to search for model names. This popup makes it easier to define your subscription when dealing with large data sets.

Search Model Name

To find a list of available Model Names, specify your search criteria, and then click "Get Model Name."

Name	Operator	Value
Landscape Handle	MATCHES	[⬍]
Model Name	CONTAINS ⬍	[ ]

[Get Model Name]

Available Model Names:

Selected Model Names:

[Add >]
[< Remove]

[Save]   [Cancel]

- Click **Select Model Type Name** to open a search window that allows you to search for model type names. This popup makes it easier to define your subscription when dealing with large data sets.

Search Model Type Name

To find a list of available Model Type Names, specify your search criteria, and then click "Get Model Type Name."

Name	Operator	Value
Landscape Handle	MATCHES	[⬍]
Model Type Name	CONTAINS ⬍	[ ]

[Get Model Type Name]

Available Model Type Names:

Selected Model Type Names:

[Add >]
[< Remove]

[Save]   [Cancel]

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

The Summary tab:

Creating a fail-safe Group

If an event is submitted to xMatters when the fail-safe functionality is enabled, and there is no subscription that matches the event, xMatterssends 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 xMatters, click the Groups tab.
2. Create a new Group named CA SPECTRUM FailSafe, with at least one User as a Team member to receive notifications.

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

Note:	*If you want to use an existing Group or a different Group name, modify the value for the* `failsafegroup` *Event Domain Constant; for more information, see "Defining Event Domain Constants" on page 10. You can also choose to not notify a fail-safe Group by setting the* `failsafe` *Event Domain Constant to* `disabled`*.*

Integration validation

After configuring xMatters and CA SPECTRUM, you can validate that communication is properly configured. It is recommended that you start the components in the following order:

- CA SPECTRUM Network Fault Manager
- CA SPECTRUM Alarm Notifier
- xMatters (alarmpoint) engine
- xMatters integration agent

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

The following sections will test the combination of xMatters and CA SPECTRUM for notification delivery and response, and Subscription Panel functionality.

Triggering a notification

To trigger a notification, create an alarm in CA SPECTRUM that matches the details of your subscription.

Responding to a notification

This section describes how to respond to a notification from xMatters. In the following example, the notification is received on a BlackBerry Device, but the process is similar for all Devices.

To respond to a notification:

1. When a notification arrives for the User, the Device indicates the number of messages received:
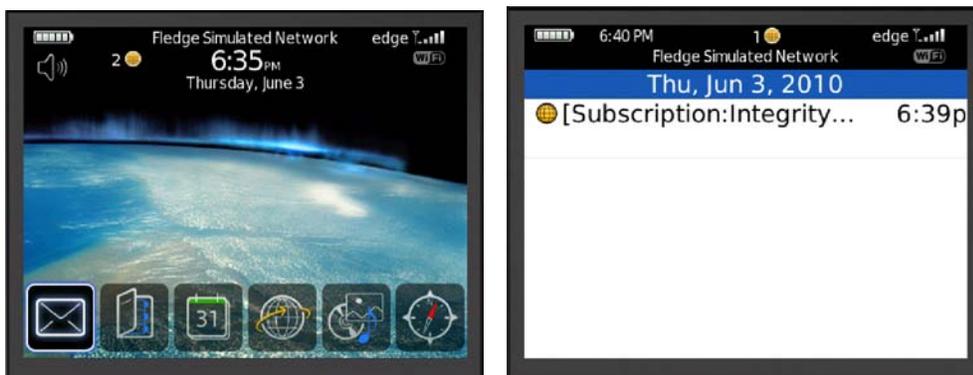
2. Opening the notification displays its details:
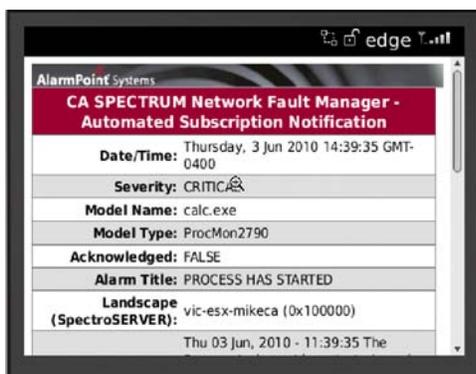
3. Scrolling down will display the remainder of the details, and the list of possible replies:
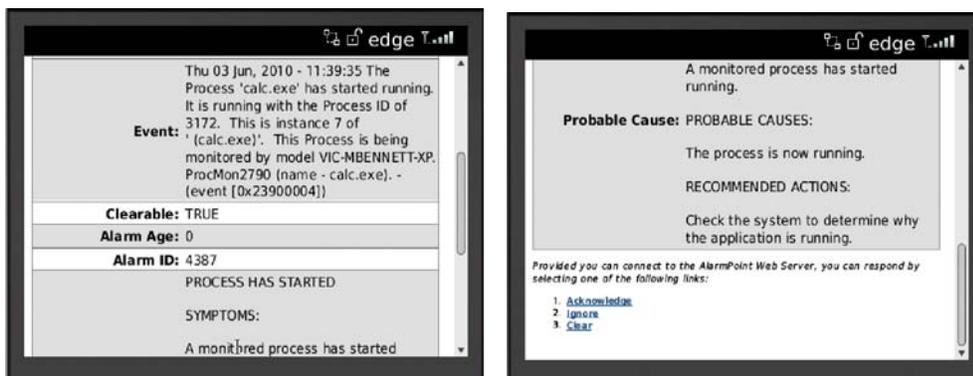
4. To respond to the notification, the User clicks a response choice, and the response is sent to CA SPECTRUM:
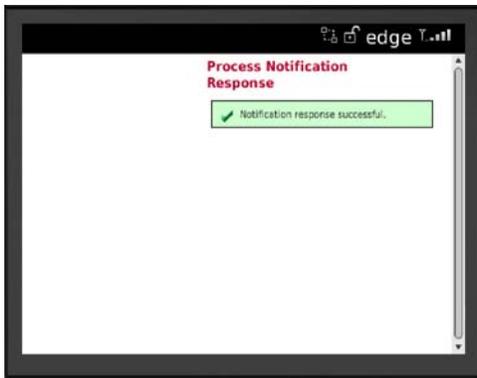
Viewing response results

To view the results of the notification response, view the Status field on the Alarm Details tab in CA SPECTRUM:

# Optimizing and extending the integration

This section describes some of the available methods you can use to optimize or extend the xMatters (alarmpoint) engine for CA SPECTRUM Network Fault Manager integration.

Modifying severity level

By default, the CA SPECTRUM Alarm Notifier only injects alarms of CRITICAL or MAJOR severity into the xMatters (alarmpoint) engine. To include other severity levels, modify the SetScript and ClearScript scripts.

To change the severity level of injected events:

1. Navigate to the `C:\win32\SPECTRUM\Notifier` directory, and open the SetScript and ClearScript scripts.
2. Locate the xMatters (alarmpoint) engine section at the end of the scripts that were added during installation.
3. Modify the test condition on the line to specify the severity levels you want to inject:

```
if [ "$SEV" = "CRITICAL" -o "$SEV" = "MAJOR" ]
```

Possible severity levels include: CRITICAL, MAJOR, MINOR, MAINTENANCE, SUPPRESSED, and INITIAL.

Note: *Increasing the number of injected severity levels may impact system performance.*

Adding new parameters

You can choose additional data parameters (or alarm properties) to include in injected events. The following steps explain how to configure the integration components to include a new event token.

To add an event token:

1. Open the `C:\win32app\SPECTRUM\Notifier\SetScript` file.
2. Add the parameter value, in quotes, to the end of the map-data line in the section added in "Modifying the CA SPECTRUM Alarm Notifier scripts" on page 8. Note that the maximum number of characters allowed in the map-data command is 255.
   - For more information about modifying these script files, refer to the CA SPECTRUM documentation.
3. Save and close the SetScript file.
4. Navigate to the `<IAHOME>\integrationservices\caspectrum\` folder and open the caspectrum.xml file in a text editor.
5. At the end of the list of parameters in the mapped-input node, add a new parameter using the same syntax as the other parameters in the list.

Note: *The parameters to APClient.bin.exe are positional so the parameter added in step 2 must be in the same position as the mapped-input element.*

6. The name of the parameter you add must match the name of the alarm property added to the `SetScript` file.
7. Save and close the caspectrum.xml file.
8. In the xMatters web user interface, add a new Event Domain predicate that matches the name of the parameter you added to the SetScript and caspectrum.xml files. For instructions on how to add predicates, see "Defining Event Domain predicates" on page 13. (Note that this step is optional.)
9. Add the predicate to the Subscription Domain, and apply it to your Subscriptions as necessary, as described in "Defining a Subscription Domain" on page 14. (Note that this step is optional.)

Adding new tokens to notification content

Once you have injected the new data elements, you can add the token as a 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 a new token to email notification content:

1. Open the xMatters Developer IDE and check out the CA SPECTRUM Network Fault Manager (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.

Response choices

The integration allows recipients to respond to notifications with several default choices, some of which are injected back to the CA SPECTRUM 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 event.

The following is a list of the default response choices available with the integration and their associated actions on the xMatters event and the CA SPECTRUM incident.

Default response choices

Response	xMatters Action	CA SPECTRUM Update	Availability
Acknowledge	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 acknowledges the event, and then later clears the event.	The status for the alert is changed to "Acknowledged by <User Name>". Any additional notes added to the Acknowledge response are logged to the AlarmPoint_Notification.txt file of the integration agent as an Annotate entry.	Email, BES, and browser; for other non-FYI mobile Devices, an Acknowledge is represented as "Ack".

Response	xMatters Action	CA SPECTRUM Update	Availability
Ignore	Signifies that the User ignores the notification. The event is escalated to the next recipient.	The user's response is logged to the `AlarmPoint_Notication.txt` file of the integration agent as "Ignored by <User Name>". Any additional notes added to the Ignore response are logged to the AlarmPoint_Notification.txt file of the integration agent as an Annotate entry.	Email, BES and Browser. For other non-FYI mobile Devices an Ignore is represented as "Ign".
Clear	Delinks all Users from the event, not allowing them to submit responses.	The status for the alert is changed to "Cleared by <User Name>", and the alarm is cleared. Any additional notes added to the Clear response are logged to the AlarmPoint_Notification.txt file of the integration agent as an Annotate entry.	Email, BES and Browser. For other non-FYI mobile Devices a Clear is represented as "Clr".
Annotate	Halts delivery of notifications to any other Devices the responding User may have configured.	This response does not affect the status of the alert. Any additional notes added to the response are logged to the AlarmPoint_Notification.txt file of the integration agent as an Annotate entry.	Email, BES and Browser. For other non-FYI mobile Devices an Annotate is represented as "Ann".

Altering the duration of events

You can modify the amount of time xMatters will send out notifications for a particular event before it times out by changing the **timeout** Event Domain Constant, as described in "Defining Event Domain Constants" on page 10. This variable stores the number of seconds the notifications will be allowed to continue before timing out.

The default value is 259200, which is the number of seconds in a 72-hour period. You can change the delay to a two-hour timeout by changing the value for the constant to 7200.

FYI notifications

You can make all notifications informational only (i.e., the user is not offered any response choices) by modifying the Event Domain Constants, as described in "Defining Event Domain Constants" on page 10. Setting the **forcefyi** Event Domain Constant to "on" makes all normal and Subscription notifications one-way (FYI).

Generating FYI notifications for Subscriptions

To ensure that all notification sent from a specific Subscription are FYI, select the **One Way** check box on the Subscription Domain details page for the associated Subscription Domain.

Manual installation

The following sections explain how to perform a manual install of the components required for this integration. You should only need these instructions if the installer file described in "Installing the integration" on page 6

Extracting the integration components

Extract the xM-CA-SPECTRUM_<version> archive (ZIP or tar.gz) to access the integration components. The following shows the notable files and folders (in bold) in the archive:

```
|-- components
| |-- alarmpoint
| | |-- lib
| | | |-- com.alarmpoint.spectrum.jar
| | | |-- wstx-asl-3.2.1.jar
| | | `-- XmlSchema-1.3.2.jar
| | |-- scripts
| | | `-- xM-CA-SPECTRUM.aps
| | |-- sub_panel
| | | |-- caspectrum
| | | | |-- SpectrumSubscriptionForm.jsp
| | | | `-- SpectrumSubscriptionSubSearch.jsp
| | | |-- SubscriptionCssInc.jsp
| | | `-- SubscriptionJsInc.jsp
| | `-- vox
| `-- alarmpoint-integration-agent
|     `-- caspectrum
|        |-- caspectrum.js
|        |-- caspectrum.xml
|        |-- lib
|        | `-- com.alarmpoint.spectrum.jar
|        |-- log4j.xml
|        `-- version.properties
|-- documentation
| `-- xm-ca-spectrum_v1.0.1.pdf
`-- version.properties
```

Installing the Subscription Panel library

To enable calls to the CLI from xMatters to CA SPECTRUM, you must copy the JAR files from the `lib` folder in the extracted integration archive into the web server library folders:

Source files:

- xM-CA-SPECTRUM_<version>\components\alarmpoint\lib\com.alarmpoint.spectrum.jar

- xM-CA-SPECTRUM_<version>\components\alarmpoint\lib\wstx-asl-3.2.1.jar

- xM-CA-SPECTRUM_<version>\components\alarmpoint\lib\XmlSchema-1.3.2.jar

Web server destination directory:

- <xMHOME>\webserver\webapps\cocoon\WEB-INF\lib

Note: *If you have installed more than one web server; install the JAR files into the web server library folders on each one.*

Installing the subscription files

To use the optional subscription panel, you must copy the JSP files into the xMatters installation folder. If you have more than one web server, repeat the following steps for each one.

To install the JSP files:

1. Copy the `xM-CA-SPECTRUM_<version>\components\alarmpoint\sub-panel\caspectrum` folder from the extracted integration archive into
`<xMHOME>\webserver\webapps\cocoon\alarmpoint\jsp\subscription`.
2. Restart the web server.

Installing the integration service

To enable the CA SPECTRUM integration service, you must copy the folder containing the integration agent files into the xMatters integration services folder and modify the caspectrum.js and `IAConfig.xml` files. If you have more than one integration agent providing the CA SPECTRUM service, repeat the following steps for each one.

To install the integration service:

1. Copy the following folder to the `<IAHOME>\integrationservices` folder:

`xM-CA-SPECTRUM_<version>\components\alarmpoint-integration-agent\caspectrum`

2. Open the `IAConfig.xml` file found in `<IAHOME>\conf` and add the following line to the "service-configs" section:

`<path>caspectrum/caspectrum.xml</path>`

3. Open the caspectrum.js file (now located in `<IAHOME>\integrationservices\caspectrum\`) and modify the paths within the `initvariables()` method to match your CA SPECTRUM installation.
4. To enable the logging of user annotations and notification delivery annotations to `<IAHOME>\logs \AlarmPoint_Notification.txt`, backup the `<IAHOME>\log4j.xml` file and then replace it with the file provided at:

`<IAHOME>\integrationservices\caspectrum\log4j.xml`.

5. Restart the integration agent.
   - On Windows, the integration agent runs as a Windows Service; on Unix, it runs as a Unix daemon.

Installing voice files

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

To install the voice files:

1. Copy all of the files in the `xM-CA-SPECTRUM_<version>\components\alarmpoint\vox\english` folder from the extracted integration archive to the following node installs folder:

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

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

Uninstalling

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

Configuration variable reference

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

Global configuration variables

These variables are available throughout the script package, and are parameters of the "main" object. The value assigned to each variable in the "Variable" column is its default value within the script. The value in the "Constant" column is the name of the variable's associated Event Domain Constant, as described in "Defining Event Domain Constants" on page 10.

Gobal variables		
Variable	**Constant**	**Description**
$main.timeout = 259200	timeout	Amount of time (in seconds) the event is allowed to run before timing out. (259200 seconds = 72 hours.)
$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		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/"		Defines the file used to log debugging information (only if $main.use_logfile is set to true).
$main.maxInvalidResponses = 3	maxInvalidResponses	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 re-notified with the same content, prefixed with a message indicating that their response was invalid.

Variable	Constant	Description
$main.annotate = true	annotate	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.subscription_annotate = false	subscriptionannotate	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	enablehtmlemail	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.
$main.AlarmPoint_URL = "http://localhost:8888"	alarmpointURL	Identifies the xMatters URL used for the HTML response form and xMatters 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 xMatters web server's Process Notification Response JSP form, used by HTML email and BES to inject responses through the system.
$main.use_logo = true	uselogo	Specifies whether HTML email notifications will display the xMatters (or custom) logo.
$main.logo = $AlarmPoint_URL & "/static/images/logos/alarmpoint/UNKNOWN.png"		Specifies the path to the graphic displayed on HTML (email and BES) notifications.

Variable	Constant	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.]"		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"		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 xMatters event has occurred.
$main.bes_pushurl = "http://localhost:8888/static"	bespushurl	Specifies the URL of the BES server. (Optional.)

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 *xMatters Online Developer's Guide*.

FYI and Subscription notification variables

The following variables configure the behavior of informational-only, or FYI, notifications. The value assigned to each variable in the "Variable" column is its default value within the script. The value in the "Constant" column is the name of the variable's associated Event Domain Constant, as described in "Defining Event Domain Constants" on page 10.

Note:	*For more information on the behavior associated with informational-only notifications, see "FYI notifications" on page 27.*

FYI and Subscription variables

Variable	Constant	Description
$force_fyi = "disable"	forcefyi	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.

Variable	Constant	Description
$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.

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 in the "Variable" column is its default value within the script. The value in the "Constant" column is the name of the variable's associated Event Domain Constant, as described in "Defining Event Domain Constants" on page 10.

Note: *For instructions on how to set up a fail-safe recipient, see "Creating a fail-safe Group" on page 18.*

Fail-safe variables

Variable	Constant	Description
$fail_safe = "enabled"	failsafe	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 = "CA SPECTRUM FailSafe"	failsafegroup	Identifies the fail-safe recipient, which is typically a Group, but may be a User.

Notification configuration variables

The following variables configure notification behavior. The value assigned to each variable in the "Variable" column is its default value within the script. The value in the "Constant" column is the name of the variable's associated Event Domain Constant, as described in "Defining Event Domain Constants" on page 10.

Notification variables

Variables	Constant	Description
$override_timeframes = false	overridetimeframes	Overrides any Device Timeframes that have been configured for a User for this notification.
$use_emergency_devices = false	useemergencydevices	Forces the use of emergency Devices as part of the Device resolution processing.
$track_delivery = true	trackdelivery	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.

Integration agent configuration variables

The `<IAHOME>\integrationservices\caspectrum\caspectrum.js` file installed on the integration agent contains the following configuration variables:

Integration agent variables

Variable	Description
cmdPath = "C:\\win32app\\SPECTRUM\\Notifier";	Defines the directory path to the CA SPECTRUM Alarm Notifier scripts and executables.
showPath = "C:\\win32app\\SPECTRUM\\vnmsh";	Defines the directory path to the CA SPECTRUM CLI executables.