xMatters *(IT)* engine

FOR CA SERVICE DESK MANAGER

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.

Thursday, November 13, 2014

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.

Corporate Headquarters

12647 Alcosta Blvd, Suite 425
San Ramon, CA 94583
Telephone: 925.226.0300
Facsimile: 925-226-0310

Client Assistance:

Email: support@xmatters.com
International: +1 925.226.0300 and press 2
US/CAN Toll Free: +1 877.XMATTRS (962.8877)
EMEA: +44 (0) 20 3427 6333
Australia/APJ Support: +61-2-8038-5048 opt 2

Customer Support Site: http://support.xMatters.com

This integration was designed and tested on an unmodified version of CA Service Desk 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 SDM, 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 xMatters's Professional Services department. For more information, contact your xMatters Sales representative.

Table of Contents

Chapter 1: Introduction

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

About xMatters

xMatters 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.

Through integration modules, xMatters can become the voice and interface of an automation engine or intelligent application (the management system, such as CA Service Desk Manager). When CA SDM 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 Service Desk Manager. Responses are executed immediately on CA SDM, enabling remote resolution of the event.

1.1 Integration Summary

This integration supports ticket notifications (from CA SDM to xMatters) through an included shell script. It also supports inbound actions (from xMatters to CA SDM).

This integration also provides a way to load Group and User data from CA SDM into an xMatters deployment.

You may need to modify the configuration described in this document 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 designing your own integration with xMatters.

1.1.1 Benefits

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

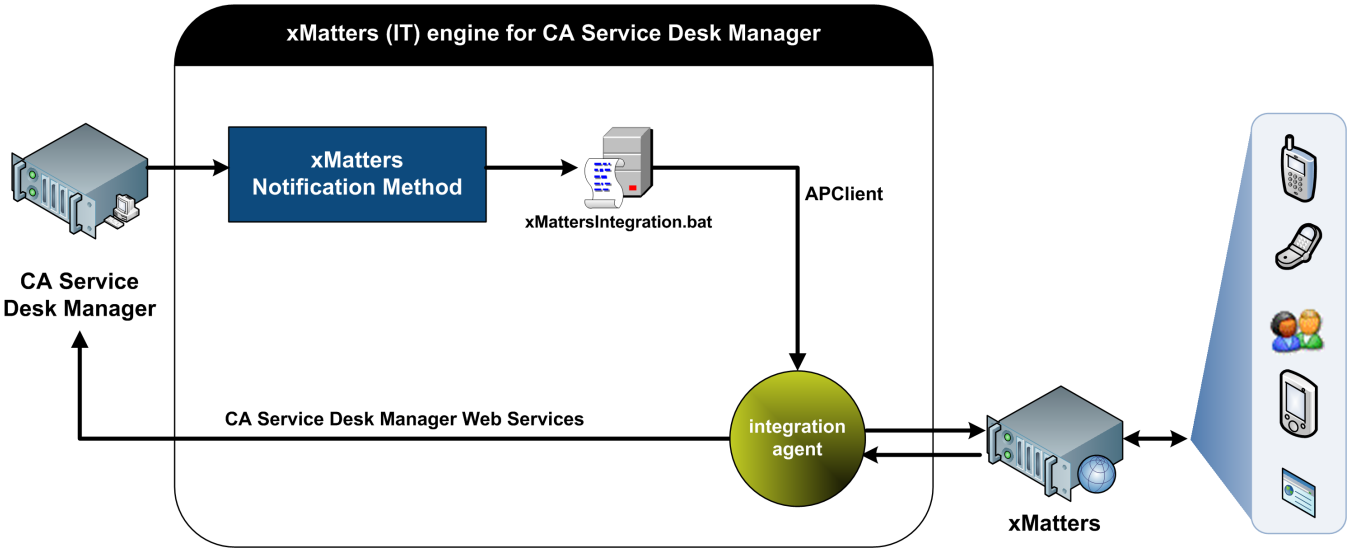
Once a response is selected on the recipient's remote device, xMatters will update the CA SDM ticket in real-time. The benefit is that this process is immediate – significantly faster than the time required for staff to notice the failures or malfunctions, determine who is on call, and manually notify 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 SDM ticket with status information.

The xMatters product features a self-service web user interface to allow accurate assignment of responsible personnel for each ticket.

1.1.2 Integration Architecture

The following diagram illustrates the high-level architecture and components of this integration:



Events injected into xMatters from CA SDM are based on notifications that would have been sent from CA SDM through other methods. As part of the integration, a custom Notification Method must be created in CA SDM. This Notification Method runs a wrapper shell script that executes APClient.bin. When the APXML messages reaches the xMatters integration agent, the integration agent parses a notification file that is created as part of the Notification Method to extract NX_NTF fields which are sent as event tokens to xMatters. A CA SDM administrator or contact can then select the "xMatters Notification Method" as the notification method for Low, Medium, High, or Emergency notifications for specific CA SDM Contacts.

The content of the notifications is based on the HTML content defined by CA SDM users; an xMatters banner is added to the top of the content.

Responses for the integration are dynamic, and based on a query back to CA SDM when the APXML messages reach the integration agent. A web service call (getValidTransitions) looks up the configured Transitions based on the type and current status of the ticket for which the notification was generated. Response handling is performed via web service calls back to CA SDM. The integration updates the state of the ticket to the next transition state.

Authentication can be done either through providing a username/password or through creating a managed session which requires a policy to be configured in CA SDM (this is the preferred method). Impersonation is also supported.

The integration also performs a batch data load. Contact records are extracted from CA SDM and then pushed to xMatters via the web service interface. This process is run by a script installed locally to CA SDM.

1.2 System Requirements

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

xMatters:

- xMatters
- xMatters integration agent (must be installed on the CA Service Desk Manager server)

CA SDM:

- CA Service Desk Manager

1.2.1 Operating Systems

The following component versions are supported by this integration.

Integration Component	Version
xMatters	5.1 patch 002
	On-Demand 5.5.58 (validated)
xMatters integration agent	5.1.3 (validated)
CA Service Desk Manager	12.9

For more information about the supported operating systems for xMatters, refer to the *xMatters installation and administration guide* and *xMatters integration agent guide*.

1.3 Conventions and Terminology

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

1.3.1 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
- code samples

Directory paths

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

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

- On Windows systems, the default is `C:\Program Files\xMatters`
- On Linux systems, the default is `/opt/xmatters`

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

- On Windows systems, the default is `C:\Program Files\xmatters\integrationagent`.
- On Linux systems, the default is `/opt/xmatters/integrationagent`.

1.3.2 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. Whenever possible, these situations are referred to using the management system's preferred terminology (i.e., ticket), but can also collectively be called events.
Management system	A management system is any sort of IT service management software with which xMatters can combine; i.e., a synonym for CA SDM.
Device	The medium through which a recipient is contacted by xMatters; i.e., email, pager, phone, BlackBerry, etc.
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*.

Chapter 2: Installation and Configuration

This chapter provides information about installing the xMatters (IT) engine for CA Service Desk Manager integration. This chapter also contains complete instructions on how to configure xMatters, CA SDM, and the integration components.

2.1 Installing the integration

This section describes the installation process for the xMatters (IT) engine for CA Service Desk Manager integration.

Note that this integration comprises two separate integration services: one for events associated with incoming tickets from CA SDM, and another for batch data load operations.

Components

This integration includes the following components that must be modified for each deployment:

Component Name	Description
caservicedesk.xml, casddataload.xml, configuration.js	The JavaScript and XML service configuration files that defines the integration services on the integration agent. Note that this integration includes two versions of `configuration.js`: one for the notification integration service, and one for the data load integration service.
xmattersDataload, xmattersDataload.bat	Command scripts used to launch the xMatters data load.
conf/deduplicator-filter.xml	The filtering mechanism used to suppress duplicate messages. The filter checks the values of certain parameters within injected events; if they are all the same within a specified timeframe, only the first message will be sent through to xMatters. You can customize these settings by adding or removing predicates in the filter, changing the suppression period or the number of messages that are compared by the integration agent. For more information about this feature, see "Filtering and suppression" on page 38.

2.1.1 Installing the integration services

To install the integration services, you must perform the following steps:

- Copy the folder containing the integration components into the integration agent; this process is similar to patching the application, where instead of copying files and folders one by one, you copy the contents of a single folder directly into the integration agent folder (`<IAHOME>`). The folder structure is identical to the existing integration agent installation, so copying the folder's contents automatically installs the required files to their appropriate locations. Copying these files will not overwrite any existing integrations.
- Modify the integration agent's `IAConfig.xml` file to include the paths for the new integration services.
- Modify the variables in the `configuration.js` files associated with the integration services.

If you have more than one integration agent providing the CA SDM service, repeat the following steps for each one.

Note: *If you have already installed an existing integration, ensure that you back up the* `deduplicator-filter.xml` *file (if one exists) in the* `<IAHOME>\conf` *folder before you install this integration.*

To install the integration service:

1. Copy all of the contents of the `\components\integration-agent` folder from the extracted integration archive to the `<IAHOME>` folder.
2. If you backed up an existing deduplicator file as indicated in the note above, merge the contents of your backup with the newly installed `<IAHOME>\conf\deduplicator-filter.xml` file: open both files in a text editor, and then copy the <filter> node from the backup file into the new deduplicator file after the last </filter> node. Save and close the file.
3. Open the `IAConfig.xml` file found in `<IAHOME>\conf` and add the following line to the "service-configs" section:

```
<path>caservicedesk-1-6-1/caservicedesk-1-6-1/caservicedesk.xml</path>
<path>caservicedesk-1-6-1/casddataload-1-6-1/casddataload.xml</path>
```

4. Open the `configuration.js` file (now located in `<IAHOME>\integrationservices\caservicedesk-1-6-1\caservicedesk-1-6-1\` folder, and set the values for the following variables:

Variable	Description
SERVICE_DESK_URL	The URL of the CA SDM web service.
SERVICE_DESK_USER	The user name of the CA SDM web service user used to access the web services.
SERVICE_DESK_PASSWORD_FILE	Location of the file containing the web service user's password; for instructions on how to set the password for this user, see "Installing the integration services", below.
DEDUPLICATOR_FILTER	The name of the filter used to suppress duplicate notifications for this integration. For more information, see "Filtering and suppression" on page 38.
DELETE_EXISTING_EVENTS	Sends a "delete" prior to creating the new event, which will clear any existing events for the incident ID.

5. Open the `configuration.js` file (now located in the `<IAHOME>\integrationservices\caservicedesk-1-6-1\casddataload-1-6-1` folder, and set the values for the following variables:

Variable	Description
SERVICE_DESK_URL	The URL to the CA SDM web services.
SERVICE_DESK_USER	The user name of the CA SDM web service user used to access the web services.
SERVICE_DESK_PASSWORD_FILE	Location of the file containing the web services user's password; for instructions on how to set the password for this user, see "Installing the integration services", below.
SEND_SYNC_SUMMARY	Determines whether the data load summary should be sent to the xMatters Administrator.
XMATTERS_ADMINISTRATOR	The xMatters User to whom you want to send the data load summary.

6. Save and close the file.
7. Restart the integration agent.
   - On Windows, the integration agent runs as a Windows Service; on Linux, it runs as a daemon.

Setting web services user password

This integration includes an encrypted file, located in the <IAHOME>\conf folder, that stores the password for the web services user required for the management system. You will need to update the file with the correct password for the SERVICE_DESK_USER variable specified in the caservicedesk-1-6-1\configuration.js and casddataload-1-6-1\configuration.js files.

Password file name:

- caservicedesk.pwd stores the password for the SERVICE_DESK_USER user used by the caservicedesk-1-6-1 integration service. If you change the name of this file, you must also update the configuration.js files to point to the correct password file.

Note that this file is used for both integration services (caservicedesk-1-6-1 and casddataload-1-6-1). If you want to use a different login for each integration service, create a new password file or copy and rename the existing PWD file. Edit the configuration.js file of the integration service you want to use, and change the SERVICE_DESK_PASSWORD_FILE variable to point to the new password file.

To specify a web service user password:

1. Open a command prompt, and then navigate to <IAHOME>\bin.
2. Run the following command, where <new_password> is the password for the web services user specified in the configuration.js file, <old_password> is the existing password (the default value for a newly installed integration is "password"), and <filename> is the name of the password file (caservicedesk.pwd).

```
iapassword.bat --new <new_password> --old <old_password> --file conf/<filename>.pwd
```

2.1.2 Installing the integration script

Each integration service has a pair of operating-system-specific integration scripts (xmattersIntegration and xmattersIntegration.bat) that are used as a bridge between CA SDM and the event integration service.

Configuring the integration script

These scripts assume that the integration agent has been installed to the default <IAHOME> folder listed in "Directory paths" on page 3. If the integration agent has been installed in a different location, the script files should be updated to reflect the installation path of your integration agent, specifically the apclient.bin.exe file.

Note also that a change in CA SDM also requires that any paths in these script do not contain spaces. This is a common issue on Windows systems, where the default installation path includes the "Program Files (x86)" folder. In this case, you will need to replace all paths in the xmattersintegration.bat file with their "8dot3" formatted equivalents.

To determine the 8dot3 version of a folder name, run the following command in its parent folder:

```
dir /x
```

This command will produce a directory listing of the folder's contents in 8dot3 format. For example, the Program Files (x86) folder is often represented as PROGRA~2. Use these folder names when configuring your integration script as described in the following steps.

To configure the integration script:

1. For Windows systems, determine the "8dot3" representation of the path to your <IAHOME> folder.
    - For example, to determine the 8dot3 representation of the default integration agent installation folder should resemble the following:

```
c:\progra~2\xmatters\integr~1
```

2. Open the <CA SDM>\bin\xmattersintegration.bat (or xmattersintegration file on Linux systems) file in a text editor.
3. Edit the following lines to replace the paths with the location of your integration agent's bin folder.

```
IF EXIST "C:\Program Files (x86)\xmatters\integrationagent\bin\" (
SET IAHOME=C:\Program Files (x86)\xmatters\integrationagent
```

Note that on Windows systems, this involves replacing both instances of C:\program files (x86)\xmatters\integrationagent with the 8dot3 representation. For example, to use the path shown earlier, you would edit the lines as follows:

```
IF EXIST "c:\progra~2\xmatters\integr~1\bin\" (
SET IAHOME=c:\progra~2\xmatters\integr~1
```

4. Save and close the xmattersintegration.bat file.

Note that some versions of the integration script contains two sets of similar lines; for example:

```
IF EXIST "C:\Program Files\AlarmPointSystems\IntegrationAgent\" (
SET IAHOME=C:\Program Files\AlarmPointSystems\IntegrationAgent
)
IF EXIST "C:\Program Files (x86)\xmatters\integrationagent\bin\" (
SET IAHOME=C:\Program Files (x86)\xmatters\integrationagent
)
```

It is recommended that you replace both sets of the lines with a single set as identified in the steps above.

Installing the integration script

To simplify the command path for the xMatters Notification Method that will be created in CA SDM, it is recommended that you copy the integration script files to the CA SDM server.

To install the integration script, do one of the following:

- On Windows systems, copy the components\integration-agent\integrationservices\caservicedesk-1-6-1\caservicedesk-1-6-1\xmattersIntegration.bat file from the extracted integration archive to the <CA SDM Install Folder>\Service Desk\bin folder. (On Windows systems, the default CA SDM installation location is C:\Program Files\CA\Service Desk Manager.)
- On Linux systems, copy the components/integration-agent/integrationservices/caservicedesk-1-6-1/caservicedesk-1-6-1/xmattersIntegration file from the extracted integration archive to the <CA SDM Install Folder>/Service Desk/bin folder.

Note that while each of the integration services has its own version of the integration script, you only need to install the script for the caservicedesk-1-6-1 integration service into the CA SDM bin folder.

2.1.3 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*.

This integration provides two sets of English voice files; one for the notification integration, and one for the data load portion.

To install the voice files:

1. Determine the value of the File Identifier associated with your Company.
   - To find your Company's File Identifier, log into the xMatters web user interface as the Super Administrator, and view the target Company's Details page (**Admin** tab **> Companies > Company name**).
2. Copy the contents of the \components\xmatters\vox\ folder from the extracted integration archive to the following node installs folder:

```
<xMHOME>\node\phone-engine\Datastore\<FILE_IDENTIFIER>\
```

For example, if you were installing the integration for the Default Company on an out-of-the-box deployment, the installation paths for the voice files would be as follows:

```
<xMHOME>\node\phone-engine\Datastore\1\caservicedesk-1-6-1\recordings\english\phrases
<xMHOME>\node\phone-engine\Datastore\1\casddataload-1-6-1\recordings\english\phrases
```

Note that if this is the first custom Event Domain you have created, the <FILE_IDENTIFIER> directory will not have been created yet. You can create it manually or log into xMatters and use the web user interface to add a new voice recording. If the Phone Device Engine is running, xMatters will create the directory structure and place the new voice recording in it.

2.2 Configuring CA Service Desk Manager

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

- Configure a new notification method to notify Users via xMatters.
- Optionally, configure CA SDM to use managed login and allow impersonations (recommended).

Note that you may also want to perform periodic maintenance on the CA SDM temporary files, as described in "Purging temporary files" on page 34.

2.2.1 Configuring a notification method

To configure a notification method for this integration, create a new notification for xMatters, and then set CA SDM to use the new notification method.

To configure a notification method:

1. In the CA Service Desk Manager Web Client interface, click the **Administration** tab, expand **Notifications**, and then click **Notification Methods**.
2. Click **Create New**.
3. In the Create New Notification Method dialog box, enter the following information:
   - **Symbol**: xMatters
   - **Write to File**: Yes
   - **Supports SMTP**: No
   - **Record Status**: Active
   - **Description**: Notify via xMatters
4. In the **Notification method** field, type the command appropriate for your operating system:

Windows:

```
launchit.exe -b xmattersIntegration.bat
```

Linux:

```
launchit -b xmattersIntegration
```

5. Set the xMatters notification method for contacts.
   - CA SDM will now use the xMatters to notify users.
6. Configure the notification rules in CA SDM.

For more information on how to automatically notify key personnel about ticket activities and events, refer to the *CA Service Desk Manager Administration Guide*.

2.2.2 Configuring CA SDM to use managed login

It is recommended that you configure CA Service Desk Manager to support managed login so that interactions from xMatters to CA SDM can be executed by impersonating the user responding to the notification.

Note: *Note that this requires each CA SDM user who you want to receive notifications from xMatters to have the* ***Impersonate*** *check box selected.*

The Managed Login functionality in CA SDM performs the user authentication by locating the defined security policy through the plain text policy code, finding the policyholder's public key associated with the policy, decrypting the encrypted policy code, matching decrypted content with the policy code, and finally, opening a session with a back-end server.

To configure Managed Login on CA SDM, please refer to the *CA Service Desk Manager Technical Reference Guide* on how to create the Manager Certificate for use in this process.

Note: *Ensure that you create a copy of the Manager Certificate file and store it in a location outside the CA SDM sub-directory. Upgrading CA Service Desk Manager will overwrite the Manager Certificate.*

Once the Manager Certificate has been created, open the `caservicedesk-1-6-1\caservicedesk-1-6-1\configuration.js` file and ensure that the ACCESS_POLICY_FILE_PATH matches the location of the Manager Certificate (ending in a forward slash) and the ACCESS_POLICY matches the name of the Manager Certificate (with no file extension). The values in the `configuration.js` file are the default values that are applicable if you are following the CA Service Desk Manager Technical Reference Guide.

If the integration is not being configured to use managed login, the MANAGED_LOGIN and ENABLE_IMPERSONATE variables should be set to false. In this case, the SERVICE_DESK_USER will be used to update the tickets from xMatters responses.

The managed login functionality applies only to the notification integration; the data load service does not use managed login.

2.2.3 Configuring a contact for system notifications

If you are configuring the integration to use managed login as described in "Configuring CA SDM to use managed login" on page 10, it is recommended that you configure a separate CA SDM contact to use for xMatters system notifications (such as delivery notifications).

You can use any Access Type that has Functional Access for issue_mgr set to "Modify" in the CA SDM Security and Role Management details. For more information, refer to your CA SDM documentation.

To configure the contact:

1. In the CA SDM interface, click the **Administration** tab, expand **Security and Role Management**, then click **Contacts**.
2. Click **Create New**.
3. In the Create New Contact dialog box, enter the following information:
   - **Last Name**: xMatters
   - **User ID**: xMatters
   - **Access Type**: Administration
   - **Status**: Active
4. Click **Save**.

Note: *If you enter a different User ID, you must also update the update the ANNOTATION_USER value in the* `<IAHOME>\integrationservices\caservicedesk-1-6-1\caservicedesk-1-6-1\configuration.js` *file.*

2.3 Configuring xMatters

Configuring xMatters to combine with CA Service Desk Manager requires the following steps:

- Import the Event Domains (one for the notification service and another for data load), and configure the integration service and Event Domain Constants for each
- Configure the default User
- Add or configure a Web Services User for each integration service
- Create a Subscription (optional)

2.3.1 Importing Event Domains and scripts

The integration package includes two XML files that were created using the xMatters "Export Integration" feature; this greatly simplifies the xMatters configuration process by enabling you to create the integration Event Domain, configure the predicates and Event Domain Constants, and import the integration script package in a single step.

To import the Event Domain packages:

1. Log in to xMatters as a Company Administrator, and click the **Developer** tab.
2. On the Event Domains page, click **Import New**.
3. On the Import Integration page, click **Browse**, and then locate the `\components\xmatters\event-domain\xM-CA-ServiceDesk.xml` file extracted from the integration archive.
4. Click **Open**, and then click **Upload**.
5. On the Import Integration page, click **Browse**, and then locate the `\components\xmatters\event-domain\xM-CA-ServiceDesk-DL.xml` file extracted from the integration archive.
6. Click **Open**, and then click **Upload**.

xMatters imports the integration configuration settings and displays the new "caservicedesk-1-6-1" and "casddataload-1-6-1" Event Domains. xMatters also creates the predicates for the Event Domains, and assigns common values as defaults. For more information about the created predicates, and instructions on how to modify them, see "Defining Event Domain predicates" on page 32.

Defining the integration service

For the installation to be successful, the integration service names must match the names specified in the `caservicedesk.xml` and `casddataload.xml` files installed on the integration agent.

To define an Integration Service:

1. In xMatters, on the Event Domains page, click the **caservicedesk-1-6-1** 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**: caservicedesk-1-6-1
   - **Description**: CA Service Desk Integration Service
   - **Path**: *Not required.*
4. Click **Save**.
5. On the Event Domains page, click the **casddataload-1-6-1** Event Domain.
6. On the Event Domain Details page, in the Integration Services area, click **Add New**.
7. Enter the following information into the form:
   - **Name**: casddataload-1-6-1
   - **Description**: CA Service Desk Data Load Integration Service

- **Path**: *Not required.*

8. Click **Save**.

Specifying connection parameters

Once you have imported the Event Domain packages and configured the Integration Services, you must specify an xMatters address that is reachable from within a notification so that responses can be processed, and other values for the Event Domain Constants.

Note: *A known issue in xMatters version 5.0 requires that all Event Domain Constants be defined in UPPERCASE.*

To specify the connection constants:

1. On the Event Domains page, in the Domains menu, click **Event Domain Constants**.
2. In the **Event Domain** drop-down list, select **caservicedesk-1-6-1**, and then click **Continue**.
   - xMatters displays the pre-configured Event Domain Constants for the integration:
3. In the Event Domain Constants list, specify the correct values for the following constants (click the name of a constant to edit its value and description).
4. Repeat the above steps for the **casddataload-1-6-1** Event Domain Constants.

Event Domain Constants

Constant Name	Default Value	Description
XMATTERSURL	http://localhost:8888	Used to specify the address of the xMatters web server. The links provided in notification content use this 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.
MAINLOGO	/static/images/logos/xmatters_email.gif	Specifies the location of the xMatters logo displayed in email notifications.
The following constants are used by the casddataload-1-6-1 integration service only; they are used in a message that is sent to a user of the integration if, while processing the user's response to a notification, the integration encounters a serious error that indicates there may be a problem with the configuration of the integration.		
ERRORCONTACT	xMatters/CA Service Desk Integration Team	Adminstrator or Admin Group for the xMatters for CA Service Desk integration.
ERORCONTACTEMAIL	admin@defaultcompany.com	Email address for the error contact group.
ERRORCONTACTPHONE	555-555-5555	Phone number for the error contact group.

Note: *For more information about the Event Domain Constants included in the integration and how to configure them to suit your deployment, see "Defining Event Domain Constants" on page 29.*

2.3.2 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.

2.3.3 Adding the Web Service Users

This integration requires a Web Service User for the CA SDM events to be injected to xMatters using web services, and a separate Web Services User for the data load portion of the integration. The following steps describe how to configure the default Web Service User, IAUser, for the event injection, and how to add another Web Service User for data load.

To set up Web Service Users:

1. In xMatters, click the **Users** tab, and then click **Find Web Service Users**.
2. On the Find Web Service Users page, click **All**.
3. In the returned search results, click **IA_User**.
4. On the Details for IA_User page, select all of the web services in the **Denied Web Services** list (Ctrl-click to select more than one, or select one and press Ctrl-A to select them all), and then click **Add**.
5. Click **Save**.
6. In the Web Service Users menu on the left side of the screen, click **Add Web Service User**.
7. Fill in the fields to create a new Web Service User.
   - Remember the User ID and password you assign to this User.
8. Select all of the web services in the **Denied Web Services** list (Ctrl-click to select more than one, or select one and press Ctrl-A to select them all), and then click **Add**.
9. Click **Save**.

2.3.4 Subscribing to Alerts

You can use the Subscriptions feature in xMatters to subscribe to CA SDM tickets that match specific criteria. For example, you could configure a subscription that would send an informational notification to a specific User each time an event entered the system that was of "Immediate" urgency, or whenever an event's status was changed to "Resolved". These notifications, and their responses, do not affect the normal progression of an event through the system.

To allow Users to subscribe to specific criteria on injected events, you must configure the Subscription using the following steps:

- Define a Subscription Domain
- Create a Subscription

- Create a Fail-Safe Group

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.

Note: *For more information about the predicates used by this integration, see "Defining Event Domain predicates" on page 32.*

To create a Subscription Domain:

1. On the Developer tab, in the Developer menu, click **Subscription Domains**.
2. On the Subscription Domains page, click **Add New**.
3. In the **Event Domain** drop-down list, select **caservicedesk-1-6-1**, and then click **Continue**.
4. On the Subscription Domain Details page, in the **Name** field, type `CA Service Desk`.
5. Select the **One-Way** check box.
   - For this integration, responses are dynamically created for each notification; this makes defining precise response choices very difficult. It is recommended that you create only One-Way Subscriptions for this integration.
6. Click **Continue**.
7. On the Select Appropriate Predicates page, add all of the predicates to the **Applied Predicates** list, and then click **Continue**.
8. 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.

Creating a subscription

You can use the subscriptions feature in xMatters to subscribe to CA SDM events that match specific criteria. For example, you could configure a subscription that would send an informational notification to a specific user each time an event entered the system that was of "Immediate" urgency, or whenever an event's status was changed to "Resolved". These notifications, and their responses, do not affect the normal progression of an event through the system.

To create a Subscription:

1. Do one of the following:
   - In xMatters 5.x, on the Alerts tab, in the Alerts menu, click **My Subscribed Alerts**. Select the **CA Service Desk** subscription domain, and click the **Add New** link.
   - In xMatters On-Demand, click your user name to open your profile page, click **Subscriptions**. Click the **Add New** button (the plus symbol), and then select **CA Service Desk**.
2. On the Subscription Details page, specify a name for the subscription, and set the subscription criteria using the tabs.
3. When you are satisfied with the criteria, click **Save** to create the subscription.

Creating a fail-safe Group

If an event is submitted to xMatters when the fail-safe functionality is enabled, and there is no Device or User that matches the event, xMatters 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 xMatters, click the Groups tab.
2. Create a new Group named CA Service Desk 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, as explained in "Configuring the Event Domain" on page 29.*

2.4 Configuring data load

This integration supports one-way batch loading (adding and updating only) of Groups, Users, and Devices from CA SDM into xMatters. To configure the data load according to your desired business behavior, modify the included configuration files; the following sections provide an overview of the configuration options.

2.4.1 Data load configuration files

The data load configuration files are installed to `<IAHOME>\integrationservices\caservicedesk-1-6-1\casddataload-1-6-1\` as described in "Installing the integration services" on page 5, and consist of the following files:

- `configuration.js`: defines the default values for objects loaded into xMatters, allows the configuration of filters that use CA SDM contact attributes to determine which contacts are transferred to xMatters, and controls the logging of data load result.
- `dataSyncList.js`: defines whether the data load will update existing objects or only add new objects, and specifies the list of included or excluded objects (referred to as the "sync list").

These files define the default values that control the behavior of data load operations and determine which users and groups are transferred from CA SDM to xMatters; you can modify the behavior of the data load process by specifying the parameters in the following tables.

Data load settings: configuration.js file

The following settings can be modified or adjusted in the `configuration.js` file:

Variable	Description	Default
SERVICE_DESK_URL	Connection information and credentials used to access the CA SDM web services that support the integration; for more information, see "Installing the integration services" on page 5.	
SERVICE_DESK_USER		
SERVICE_DESK_PASSWORD_FILE		
SEND_SYNC_SUMMARY	Determines whether the data load summary should be sent to the User specified in XMATTERS_ADMINISTRATOR Note that this summary is written to the integration agent logs.	true
DATA_LOAD_EXCEPTION_LOG_MESSAGE	Defines the message that is written to the integration agent log file when a data load operation terminates abnormally due to an error.	"Data Load operation terminated with exception"

Variable	Description	Default
XMATTERS_ADMINISTRATOR	The xMatters User ID to which you want to send the data load summary.	companyadmin
WHERE_CLAUSE_USERS	Selects a subset of CA SDM users for transfer to xMatters.	Selects only active CA SDM Contacts whose Contact type is one of Analyst, Help Desk, Manager, Operator or Technician.
WHERE_CLAUSE_GROUPS	Selects a subset of CA SDM groups for transfer to xMatters.	Selects only active CA SDM Contacts whose Contact type is Group.
WHERE_CLAUSE_GROUP_ MEMBERS	Selects the CA SDM users who are members of a specified group. **This clause must be changed if WHERE_ CLAUSE_USERS is altered.**	Selects members of the specified group that are active CA SDM Contacts whose Contact type is one of Analyst, Help Desk, Manager, Operator or Technician.
ATTRIBUTES_CNT_OBJECT **ATTRIBUTES_GRPMEM_OBJECT** **ATTRIBUTES_CONTACT_ROLES**	Determine which attributes are retrieved when contacts, group members and contact roles are selected from CA SDM.	
DEFAULT_USER_FIRST_NAME	Determines the first name of any users created in xMatters when the corresponding CA SDM Contact does not have a first name.	"Undefined"
DEFAULT_SUPERVISOR	The xMatters User ID of the default supervisor for Users and Groups in xMatters.	companyadmin
DEFAULT_XMATTERS_ROLES	The default Roles assigned to Users in xMatters.	Standard User
MAP_USER_ROLES	If this variable is *false*, all xMatters Users will be assigned the Role or Roles defined by DEFAULT_ XMATTERS_ROLES, irrespective of what CA SDM Contact Type the individual users have. If *true*, the xMatters Roles assigned to individual Users will be based on their types inCA SDM and the role mappings defined by the roleMap variable.	false
DEFAULT_USER_SITE	Determines the Site to which any new xMatters User should belong.	Default Site
DEFAULT_PROVIDER_EMAIL	Specifies the default user service provider for email devices,	(x)matters Email Gateway
DEFAULT_PROVIDER_TEXT	Specifies the default user service provider for text devices.	(x)matters SMS Gateway

Variable	Description	Default
DEFAULT_PROVIDER_VOICE	Specifies the default user service provider for voice devices.	(x)matters Voice Gateway
EXTERNALLY_OWN_USERS	Determines whether Users created in xMatters should be marked as "Externally Owned".	false
EXTERNALLY_OWN_GROUPS	Determines whether Groups created in xMatters should be marked as "Externally Owned".	false
WEB_LOGIN_TYPE	Specifies the method of web login to use for Users created in xMatters; possible values are "NATIVE" (uses xMatters web login credentials) or "LDAP".	NATIVE
WEB_LOGIN_LDAP_DOMAIN	Specifies the LDAP domain to use if WEB_ LOGIN_TYPE is set to "LDAP".	company.com
roleMap	If MAP_USER_ROLES is set to true, this setting specifies how to map CA SDM support groups functional roles to xMatters Roles.	
countryCodes	Associates international country dialing prefixes with ISO 3166-1 alpha-2 codes required by xMatters.	The default mapping converts a dialing code of "1" to the country code "US"

Data load settings: dataSyncList.js file

The following settings can be modified or adjusted in the `dataSyncList.js` file; for more information about these settings and how they interrelate, see the following section, "Data load process":

Variable	Description
SYNC_ACTION	Defines whether the Users and Groups specified in the syncList parameter (defined below) should be included or excluded in the data load.
USER_SEED_ONLY	If set to true, user objects will be added to xMatters only when they are initially loaded, and not updated. If set to false, any modifications to the object in CA SDM will be synchronized with the object in xMatters, overwriting any changes that may have been made to the object in xMatters.
GROUP_SEED_ONLY	If set to true, group objects will be added to xMatters only when they are initially loaded, and not updated. If set to false, any modifications to the object in CA SDM will be synchronized with the object in xMatters, overwriting any changes that may have been made to the object in xMatters. Note that the update process will preserve existing xMatters Team information, such as type and rotation settings, and the rotation order and delay settings for existing members.
syncList	An XML document defining user and group names that should be excluded or included, as explained in the following sections. **Note**: The list of users must not exceed 50 entries, due to a limitation in CA SDM. Similarly, the list of groups must also not exceed 50 entries. For more information and strategies on handling larger lists, see "Specifying longer lists of users or groups to include/exclude" on page 43.

2.4.2 Data priority and sources

The data load integration service uses the following rules for creating and updating the individual properties of Users, Groups, and Teams in xMatters:

1. New objects are created from a combination of CA SDM object information and configured defaults provided by the integration service file.
2. When an object in CA SDM is changed and the corresponding User or Group in xMatters is updated via the batch data load:
   - Any properties that are populated with information from CA SDM will be updated based on CA SDM information even if the property has been changed in xMatters.
   - Any properties that are populated with configured defaults will not be updated, and any changes in xMatters will be preserved.

The following tables describe the source for the data used to populate the fields Users, Groups and Devices in xMatters.

User data

The data for user details in xMatters that is obtained from CA SDM is provided by the XM_CTM_People_WS web service which exposes fields belonging to CTM:People.

xMatters Field	Source
Active	CA SDM: deleteFlag
User ID	CA SDM: userid
First Name	CA SDM: first_name (or configuration file if not found)
Last Name	CA SDM: last_name
User Devices	CA SDM
Roles	Configuration file OR CA SDM user roles
User Site	Configuration file
Supervisors	Configuration file
Externally Owned	Configuration file
Web Login Type	Configuration file
LDAP Domain	Configuration file

Group data

The data for group details in xMatters that is obtained from CA SDM is provided by the CA Unicenter web service.

xMatters Field	Source
Group Name	CA SDM: last_name
Active	CA SDM: deleteFlag
Externally Owned	Configuration file

User Devices

The data for device details in xMatters is provided by the CA Unicenter web service.

Note: *For these Devices to be loaded, you may need to add the Device Names into xMatters. For more information about adding Device Types and Device Names, see the xMatters installation and administration guide.*

xMatters Device Name	xMatters Device Type	CA SDM Field
Work Email	Email	email_address
Pager Email	Email	pemail_address
Work Phone	Voice	phone_number
Home Phone	Voice	alt_phone
SMS Phone	Text Phone	beeper_phone
Other Phone	Voice	fax_phone

The data load integration service supports phone numbers with the following formats:

- Local 7-digit numbers: 555 6473, 555-6473
- 10-digit numbers: 212 555 6473, (212) 555 6473, 212.515.6473
- 10-digit numbers preceeded by a 1: 1 212 555 6473, 1 (212) 555 6473

The non-digit separators can be any number of non-digit characters, including white space, brackets, dashes, or periods; i.e., " (", ")", "-", ".", etc

Note: *Phone extensions are not supported in any form.*

The default integration behavior is to map the dialing prefix 1 to the Country Code Override "US". To change this, or to add additional mappings, see "Phone numbers and country code mapping" on page 43.

2.4.3 Data load process

The data load integration transfers user and group information between CA SDM and xMatters via batch data load when a CA SDM user manually initiates the process. You must run the `xmattersDataload` script as described in "Validating User and Group Data Load" on page 23. This instructs the integration to retrieve lists of qualifying users and groups from CA SDM and transfer them to xMatters .

The following sections explain the data load process in more detail.

Batch user data load

The USER_SEED_ONLY variable determines how users are treated when loading them into xMatters:

If the USER_SEED_ONLY variable is set to *true*:

- If (User ID **does not exist** in xMatters) AND (User meets Batch Qualification criteria) AND [(User ID is not in excluded list) OR (User ID is in included list)]:
  - New User is created in xMatters
  - User information is populated
  - User Devices are added to xMatters

- IF (User ID **already exists** in xMatters):
  - No changes are made to any Users or Devices

If the USER_SEED_ONLY variable is set to *false*:

- If (User ID **does not exist** in xMatters) AND (User meets Batch Qualification criteria) AND [(User ID is not in excluded list) OR (User ID is in included list)]:
  - New User is created in xMatters
  - User information is populated
  - User Devices are added to xMatters
- IF (User ID **already exists** in xMatters):
  - User information is updated
  - User Devices are added to match settings in CA SDM

Usage notes:

The integration is not able to change the User ID of xMatters Users. If a Login ID of a user in CA SDM is changed, the resulting update to xMatters will create a new User and leave the original xMatters User unchanged.

Batch group data load

The GROUP_SEED_ONLY variable determines how users are treated when loading them into xMatters:

If the GROUP_SEED_ONLY variable is set to *true*:

- If (Group name **does not exist** in xMatters) AND (Group meets Batch Qualification criteria) AND [(Group name is not in excluded list) OR (Group name is in included list)]:
  - New Group is created in xMatters
  - Group attributes are added
  - Coverage is created
  - Team is created
- IF (Group name **already exists** in xMatters):
  - No changes are made

If the GROUP_SEED_ONLY variable is set to *false*:

- If (Group name **does not exist** in xMatters) AND (Group meets Batch Qualification criteria) AND [(Group name is not in excluded list) OR (Group name is in included list)]:
  - New Group is created in xMatters
  - Group attributes are added
  - Coverage is created
  - Team is created
- IF (Group name **already exists** in xMatters):
  - Group attributes are updated
  - Team membership is updated (see "Team data load" on page 21)
  - Group Coverage is **not** updated

Notes:

- Group updates may modify team memberships, but will maintain existing xMatters Team information, such as type and rotation settings, and the rotation order and delay settings for existing members.

- If a CA SDM group has no members at the time it is transferred to xMatters, a Group will be created with a Coverage and a Team, but the Team will have no members.
- Renaming Groups via data load is not supported: if you change the name of a CA SDM group after the group has been transferred to xMatters, subsequent synchronization will create a new Group in xMatters matching the new name of the CA SDM group. The data load will not modify the original xMatters Group, and the new Group will not have any Coverages that may have been added to the original Group.

Team data load

Team membership is updated in xMatters only as part of a batch update of CA SDM Support groups; the following sections are included to provide detail on the process.

If the GROUP_SEED_ONLY variable is set to *true*:

- If (Group name **does not exist** in xMatters):
  - New Team will be created in xMatters with the name "<GroupName> - Default Team"
  - The Team is associated with a default 24x7 Coverage
  - New Team will comprise only those Users in xMatters who are members of the group in CA SDM AND for whom "Assignment Availability" equals "Yes"
- IF (Group name **already exists** in xMatters):
  - No changes are made

If the GROUP_SEED_ONLY variable is set to *false*:

- If (Group name **does not exist** in xMatters):
  - New Team will be created in xMatters with the name "<GroupName> - Default Team"
  - The Team is associated with a default 24x7 Coverage
  - New Team will comprise only those Users in xMatters who are members of the group in CA SDM AND for whom "Assignment Availability" equals "Yes"
- IF (Group name **already exists** in xMatters) AND ("<GroupName> - Default Team" **already exists** in xMatters):
  - Team with the name "<GroupName> - Default Team" will be updated
  - The Team is not associated with any Coverage
  - Team will comprise only those Users in xMatters who are members of the group in CA SDM AND for whom "Assignment Availability" equals "Yes"
  - Rotation and escalation settings, and ordering of existing team members in xMatters will be reset
- IF (Group name **already exists** in xMatters) AND ("<GroupName> - Default Team" **does not exist** in xMatters):
  - New Team will be created in xMatters with the name "<GroupName> - Default Team"
  - The Team is **not** associated with any Coverage
  - New Team will comprise only those Users in xMatters who are members of the group in CA SDM AND for whom "Assignment Availability" equals "Yes"

2.4.4 Data load notification and logging

Following each data load operation, a summary of successful, successful with warning, and failed actions is written to the integration agent logs. You can also use the XMATTERS_ADMINISTRATOR and SEND_SYNC_SUMMARY variables in the `configuration.js` file to send a notification containing the summary to a User or Group within xMatters.

Note that the summary notification is FYI-only. No user responses to the notification are supported and the integration does not support any annotations from the recipients of the notification or from xMatters concerning the delivery of the notification.

If the data load operation terminates before completion, a message is written to the integration agent logs indicating what has happened. The contents of this string can be configured via the DATA_LOAD_EXCEPTION_LOG_MESSAGE in the `configuration.js`. In this case, a summary notification is also sent to the configured xMatters User or Group.

Chapter 3: Integration Validation

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

- CA SDM
- xMatters integration agent
- xMatters

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

The following sections will test the combination of xMatters and CA SDM for notification delivery and response, and data load configuration.

3.1 Validating User and Group Data Load

The following tests the communication between CA SDM and xMatters to ensure that the data load is properly configured.

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

To test the User load:

1. Review the contents of `integrationservices\caservicedesk-1-6-1\casddataload-1-6-1\dataSyncList.js` and the value of WHERE_CLAUSE_USERS in `integrationservices\caservicedesk-1-6-1\casddataload-1-6-1\configuration.js`.
   - The default configuration will only process updates to active CA SDM Contacts that have specific contact types.
2. In CA SDM, create a new active user that meets the criteria of the configuration files, and assign them an email address.
3. Update the synchronization lists (as described in "Configuring data load" on page 15) to include and add the User ID of the user you created in step 2.
   - This will limit the test to only the user you added.
4. Navigate to the `<IAHOME>\integrationservices\caservicedesk-1-6-1\casddataload-1-6-1` folder, and confirm that the value of IAHOME in the xmattersDataload or xmattersDataload.bat file (depending on your platform) is correct.
   - If you installed the integration agent to a location other than the default installation folder, you will need to specify is location within the file.
5. Run one of the following commands:

Windows:

```
xmattersDataload.bat load
```

Linux:

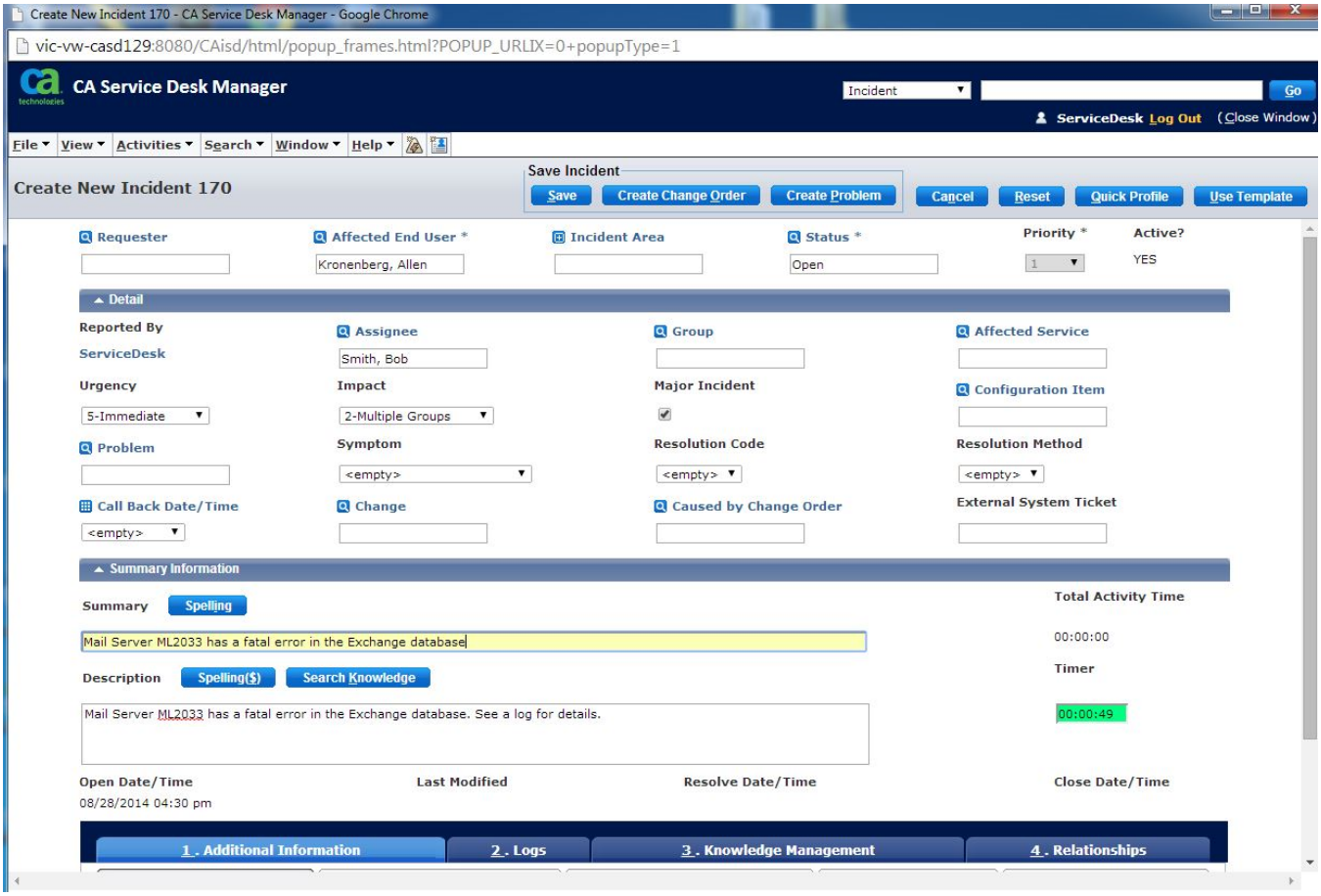
```
xmattersDataload load
```

6. Log in to xMatters, and confirm that the user has been added, and has an Email Device with the correct address.

- For additional information on the data load process, consult the integration agent's `AlarmPoint.txt` file, which includes a summary detailing any failures or warnings and their reasons.

3.2 Triggering a notification

To trigger a notification, create a new ticket or incident in CA SDM, and assign it to "Bob Smith", or a User that exists in both CA Service Desk Manager and xMatters.

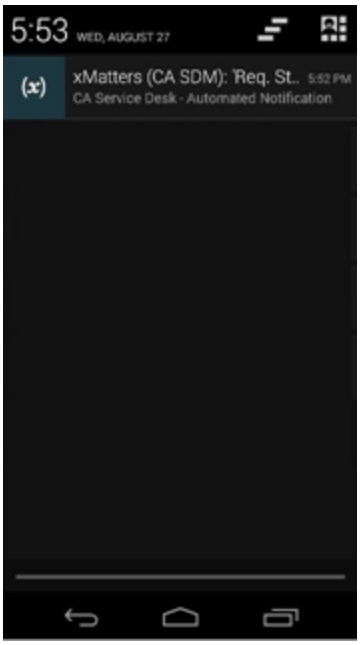


3.3 Responding to a notification

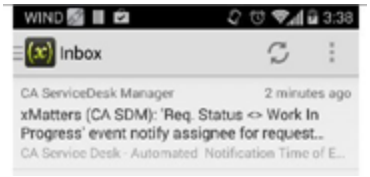
This section describes how to respond to a notification from xMatters. In the following example, the notification is received via the xMatters mobile app on an Android phone, but the process is similar for all devices.

To respond to a notification:

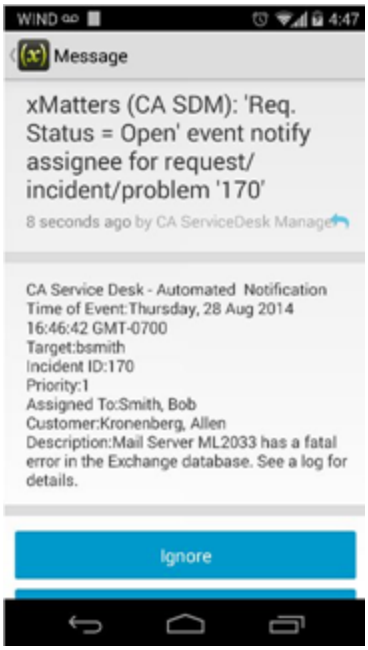
1. When the notification arrives, the mobile app uses the subject of the notification as a push message to let the user know that a new alert is available:



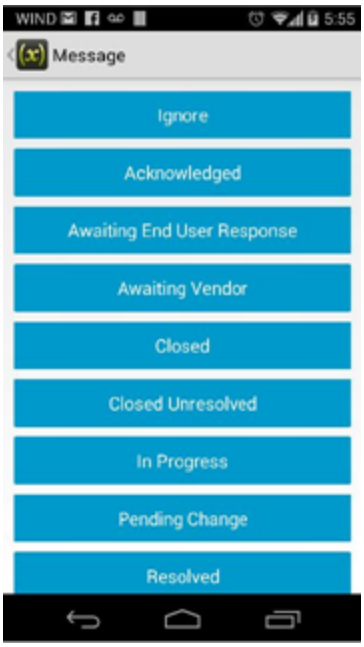
2. When the user opens the mobile app, they can see the new alert waiting in their inbox:



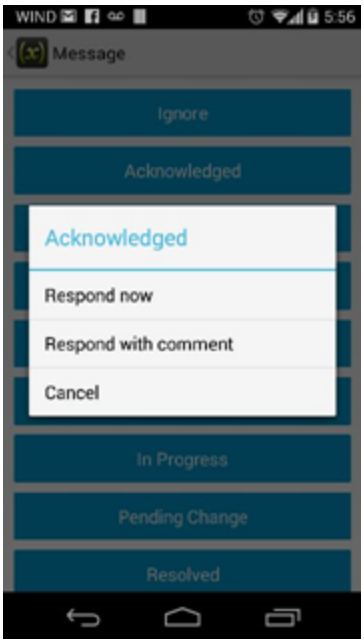
3. The user can open the alert to see its contents and the details of the problem:



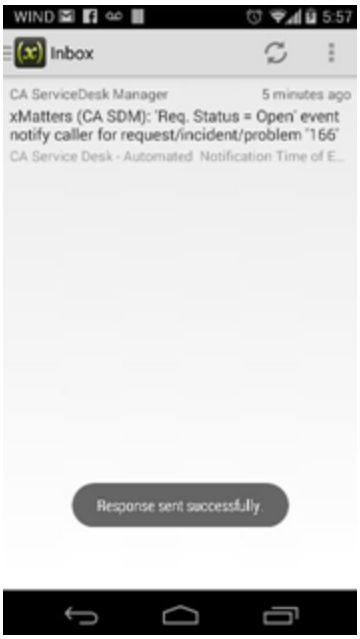
4. To respond to the notification, the user scrolls down to view the available options, and then clicks a response choice:



5. On some systems, the user can also choose to add a comment or annotation to the response.
    - Note that these annotations are visible only in the event's Log report in xMatters, and not sent back to CA SDM.



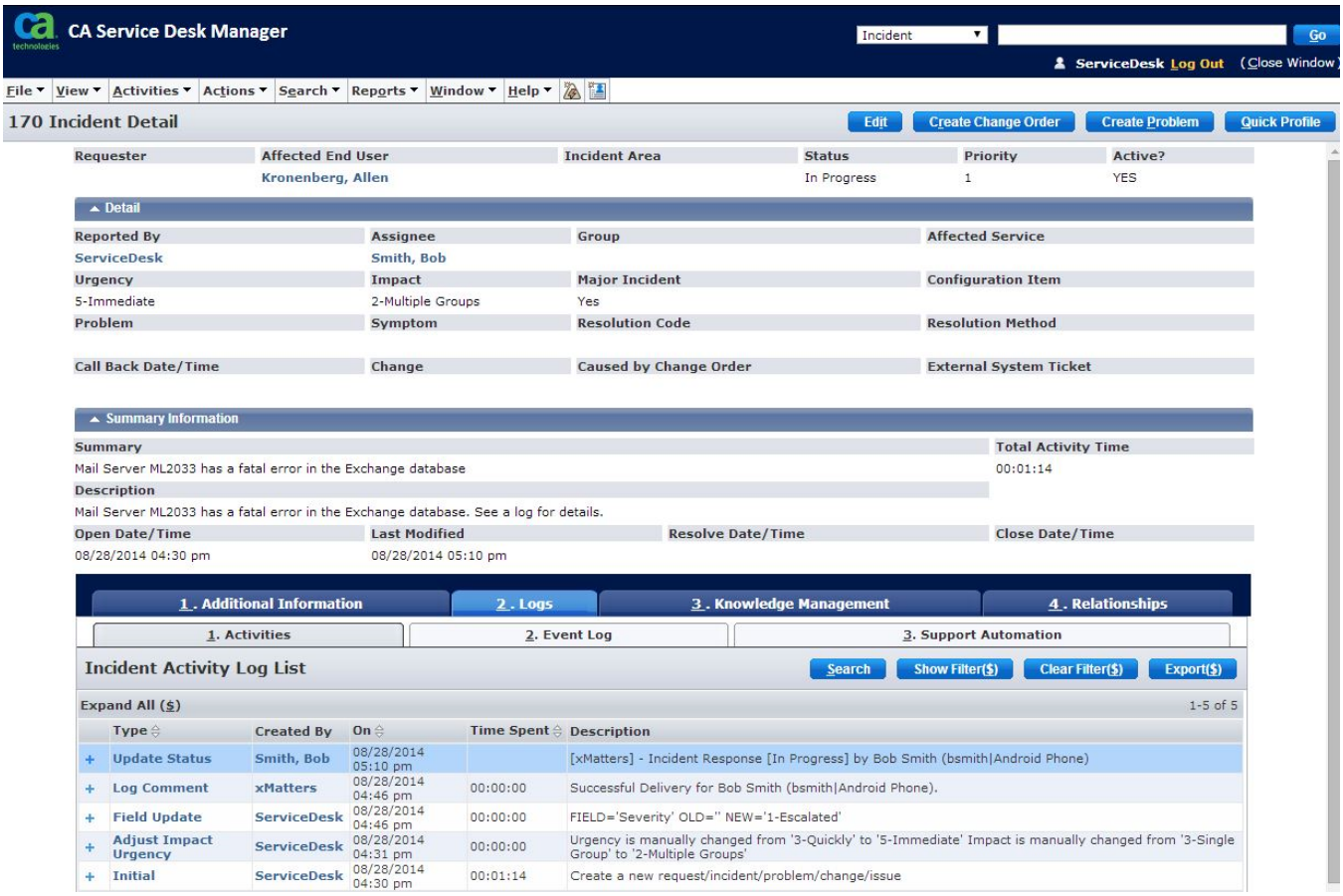
6. When the user submits their response, the app will update the event in CA SDM:



3.4 Viewing response results

To view the results of the response, view the ticket in CA SDM.

In the following example, the "Assignee" field has been updated to the name of the responding User, the "Status" has been updated to "Acknowledged", and the "Incident Activity Log List" has been updated with additional entries to show the changes resulting from the User's response:


CA Service Desk Manager
Incident
Go
ServiceDesk Log Out (Close Window)
File View Activities Actions Search Reports Window Help
170 Incident Detail
Edit
Create Change Order
Create Problem
Quick Profile
Requester
Affected End User
Incident Area
Status
Priority
Active?
Kronenberg, Allen
In Progress
1
YES
Detail
Reported By
Assignee
Group
Affected Service
ServiceDesk
Smith, Bob
Urgency
Impact
Major Incident
Configuration Item
5-Immediate
2-Multiple Groups
Yes
Problem
Symptom
Resolution Code
Resolution Method
Call Back Date/Time
Change
Caused by Change Order
External System Ticket
Summary Information
Summary
Total Activity Time
Mail Server ML2033 has a fatal error in the Exchange database
00:01:14
Description
Mail Server ML2033 has a fatal error in the Exchange database. See a log for details.
Open Date/Time
Last Modified
Resolve Date/Time
Close Date/Time
08/28/2014 04:30 pm
08/28/2014 05:10 pm
1. Additional Information
2. Logs
3. Knowledge Management
4. Relationships
1. Activities
2. Event Log
3. Support Automation
Incident Activity Log List
Search
Show Filter($)
Clear Filter($)
Export($)
Expand All ($)
1-5 of 5
Type
Created By
On
Time Spent
Description
+ Update Status Smith, Bob 08/28/2014 05:10 pm [xMatters] - Incident Response [In Progress] by Bob Smith (bsmith|Android Phone)
+ Log Comment xMatters 08/28/2014 04:46 pm 00:00:00 Successful Delivery for Bob Smith (bsmith|Android Phone).
+ Field Update ServiceDesk 08/28/2014 04:46 pm 00:00:00 FIELD='Severity' OLD='' NEW='1-Escalated'
+ Adjust Impact Urgency ServiceDesk 08/28/2014 04:31 pm 00:00:00 Urgency is manually changed from '3-Quickly' to '5-Immediate' Impact is manually changed from '3-Single Group' to '2-Multiple Groups'
+ Initial ServiceDesk 08/28/2014 04:30 pm 00:01:14 Create a new request/incident/problem/change/issue

Chapter 4: Optimizing and Extending the Integration

This section describes some of the available methods you can use to optimize or extend the xMatters (IT) engine for CA Service Desk Manager integration.

4.1 Manually configuring xMatters

This integration includes an exported version of the xMatters Event Domain that includes event domain constants and predicates. If you do not want to use this to create and configure the required event domain, the following sections describe how to manually configure xMatters.

4.1.1 Configuring the Event Domain

By default this integration is set up to use the Event Domains “caservicedesk-1-6-1” and "casddataload-1-6-1"; it is strongly recommended that you use these default Event Domains.

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

To define the Event Domains:

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**: caservicedesk-1-6-1
   - **Description**: CA SDM Integration
   - **Script Package**: CA Service Desk
5. Click **Save**.
6. On the Event Domains page, click **Add New**.
7. Enter the following information into the form:
   - **Name**: casddataload-1-6-1
   - **Description**: CA SDM Data Load Integration
   - **Script Package**: CA Service Desk Manager Data Load
8. Click **Save**

Once you have defined the Event Domain, you can add the integration service, as described in "Defining the integration service" on page 11, and the Event Domain predicates, as described in "Defining Event Domain predicates" on page 32.

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.

Note that while the constants listed in this section are present in both of the integration's Event Domains, most of them will have no impact on the behavior of the data load integration as it sends only FYI notifications.

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, or speak to an xMatters client assistance representative before changing these values.

Note: *The values for the XMATTERS and BESPUSHURL constants should be modified to specify the address of the xMatters web server (to enable the HMTL 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 **caservicedesk-1-6-1**.
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.
   - Note that Event Domain Constant names in xMatters version 5.0 MUST be defined in uppercase.
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 Values column of the following table.

Note: *Shaded rows indicate **mandatory** settings that are specific to your deployment. You must change the default settings to match your instance.*

Constant Name	Default Value	Description
XMATTERSURL	http://localhost:8888	Used to specify the address of the xMatters web server. The links provided in notification content use the XMATTERSURL 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. Populates the `$main.xmatters_url` variable.
BESPUSHURL	http://localhost:8888/static	Used to specify the address of the BES device server. Populates the `$main.bes_pushurl` parameter.
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.

Constant Name	Default Value	Description
FAILSAFEGROUP	CA Service Desk 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 xMatters and no subscriptions exist that match the event. Set this constant if you want to change the failsafe group from CA Service Desk FailSafe to another group defined in xMatters.
FAILSAFE	enabled	Controls fail-safe functionality, notifying the fail-safe recipient via EMAIL under certain circumstances; possible values are: • **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. Populates the `$fail_safe` parameter.
OVERRIDETIMEFRAMES	false	Override Recipients Device Timeframes. Populates the `$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.
TRACKSUBSCRIPTIONDELIVERY	true	Track when each Device is delivered to for Subscriptions. Populates the `$track_subscriptionDelivery` parameter.

Constant Name	Default Value	Description
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.
MAINLOGO	/static/images/logos /xmatters_email.gif	Indicates the location of the image that will be displayed in HTML notification.
USEURLALIAS	false	Indicates how Response Choices are presented to xMatters to ensure that the user is authenticated in the correct company so the notification can be updated.; set to *true* for xMatters On-Demand integrations.
DEBUG	false	Indicates whether to use the debug level for logging messages. Populates the `$main.debug` variable.
ENABLESUBSCRIPTIONS	true	Indicates whether to enable processing of Subscriptions on incoming events.
The following constants are used in a message that is sent to a user of the integration if, while processing the user's response to a notification, the integration encounters a serious error that indicates there may be a problem with the configuration of the integration.		
ERRORCONTACT		Adminstrator or Admin Group for the xMatters for CA Service Desk integration.
ERORCONTACTEMAIL		Email address for the error contact group.
ERRORCONTACTPHONE		Phone number for the error contact group.

Defining Event Domain predicates

This section describes how to add and configure the default Event Domain predicates used by this integration. You can also use the following steps to add other predicates that you consider important and which you plan to add to the integration.

To define the Event Domain predicates:

1. In xMatters, click the **Developer** tab.
2. On the Event Domains page, click caservicedesk-1-6-1.
3. On the Event Domain Details page, click the **Add New** link beside the **Predicates** heading.
4. Add the following predicates to the Event Domain:

Event Domain predicates

Predicate	Type	Important	Description
assignee_name	Text		Name of the User to which the event was assigned.
impact	List		The perceived impact of the event, as defined in CA SDM; default values are: • 1-Entire organization • 2-Multiple Groups • 3-Single Group • 4-Small Group • 5-One person
requestor_name	Text		Name of the User who submitted the request.
priority	List		The severity of the event as defined in CA SDM; default values are: • None • 1 • 2 • 3 • 4 • 5
status	List		The status of the event in CA SDM; default values are the same as those defined in "Response choices" on page 35.
urgency	List		The urgency assigned to the event in CA SDM; default values are: • 1-When Possible • 2-Soon • 3-Quickly • 4-Very Quickly • 5-Immediate
ticket_factory	List		The type of ticket created in CA SDM; default values are: • chg • iss • in • cr • pr
subject	text		The subject of the notification.

Data Load predicates

The "casddataload-1-6-1" Event Domain contains the following default predicates:

Event Domain predicates

Predicate	Type	Important	Description
has_errors	List		Indicates whether the data load encountered any errors; possible values are "true" and "false". Note that the has_errors and has_warnings predicates are provided to enable administrators and other Users to create subscriptions in xMatters that will notify them about data load errors or warnings.
has_warnings	List		Indicates whether the data load encountered any warnings; possible values are "true" and "false".

Note: *For more information about predicates and how they work in xMatters, see the* xMatters installation and administration guide.

4.2 Purging temporary files

This integration relies on temporary files created by CA SDM to store notification data that will be sent to the integration agent. Files generated for the xMatters integration are not needed after the incident has been injected to the integration agent, but the temporary files are not automatically deleted.

If these files are not purged occasionally, injections into xMatters may be delayed while CA SDM searches for the next available file name; for example, after restarting CA SDM.

By default, these files are stored in the `temp` folder on the CA SDM server, and are named sequentially, e.g. `c:\temp\1`, `c:\temp\2`, etc.

The integration (as of version 1.4) creates a folder into which it moves the notification files after they have been processed. This folder is named "xmatters", and is located within the original file location.

For example, if CA SDM creates a notification file called `C:\temp\1`, the integration renames the file and moves it to `c:\temp\xmatters\1.<date>`, where `<date>` is the current date and time in a "`yyMMddHHmmss`" format. (The directory and filename scheme can be customized by editing the `caservicedesk.js` integration script.

This process is intended to reduce the delay presently encountered after CA SDM is restarted, before notifications are sent to the integration agent.

It is recommended that you occasionally purge the files from the `temp\xmatters` folder to reduce congestion of the CA SDM server's file system.

4.2.1 Customizing the handling of temporary files

The temporary file handling feature can be disabled or configured by modifying the following sections within the `caservicedesk.js` integration script.

```
// markProcessed() will attempt to move the parsed notification file to a subdirectory called
xmatters.
// This is bound by file system access so if you don't want this feature just comment out
fileParser.markProcessed().
fileParser.setCopyAfterProcessing(false);
fileParser.markProcessed();
//fileParser.markProcessed('<custom directory>');
// If you want a different directory, use full qualified path.
```

```
//fileParser.markProcessed('<custom directory>', '<custom filename>');
// If you want to control directory and file name.
```

If setCopyAfterProcessing is set to "true", then the integration will put a copy of the CA SDM temporary file into the `xmatters` folder after processing; otherwise, the file will be moved.

If you want the file to be moved to a folder other than `%temp%\xmatters`, then specifiy the desired location by setting the "fileParser.markProcessed('<custom directory>');" parameter. For example:

```
fileParser.markProcessed('c:\deleteme');
```

If you edit the script to specify "fileParser.markProcessed('<custom directory>', '<custom filename>');", then the files will be moved to a custom folder and will be renamed. Note that this will cause the previous notification files to be overwritten, and so it will effectively prevent the hard drive from filling up with temporary files (since each file will replace the previous one when it is moved and renamed.)

For example:

```
fileParser.markProcessed('c:\deleteme', 'latest-notification-file.txt');
```

4.3 Response choices

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

Note that the configuration options described in this section apply only to the caservicedesk-1-6-1 integration service; the data load portion of this integration does not support response choices.

The response choices presented on a notification will vary according to the ticket type that the notification represents, and the current status of the incident. xMatters presents the response choices that would be available in the ticket if the user were to perform the update directly in CA SDM.

Which the exception of "Ignore" and "Annotate", response choices are essentially ticket statuses and will update the status of the ticket to the specified response choice. CA SDM also adds an activity log entry to the ticket with details of who changed the status, and from which Device. "Ignore" and "Annotate" do not modify the ticket status; they only update the activity log.

For short text (SMS) and non-HTML email Devices, the responder must respond with the response code as spaces in the statuses would cause the response handler to take the text after the first word as annotation text.

Note: *To remove the "Ignore" option, edit the* `configuration.js` *file in a text editor, and set the* `ADD_IGNORE_RESPONSE` *option to* false.

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 SDM incident. For a definition of the job control terms, see the list below the table.

Default response choices

Response	Response Code	Job Control
Ignore	IGN	Notify next, delink responder.
Acknowledge	ACK	Delink all except responder.
Analysis Complete	PRBANCOMP	Delivered
Approval In Progress	APP	Delink all except responder.
Approved (Request/Incident/Problem)	PRBAPP	Delink all except responder.

Response	Response Code	Job Control
Approved (Change)	APR	Delink all except responder.
Avoided	AVOID	Notify next, delink responder.
Awaiting End User Response	AEUR	Delink all except responder.
Awaiting Vendor	AWTVNDR	Delink all except responder.
Backed Out	BACK	Delink all except responder.
Cancelled	CNCL	Delink all.
Close Requested	CLREQ	Delink all except responder.
Closed	CL	Delink all.
Closed Unresolved	CLNRSLV	Delink all.
Customer Hold	CSTHLD	Delink all except responder.
Fix In Progress	FIP	Delink all except responder.
Fixed	FXD	Delink all except responder.
Hold	HLD	Delink all except responder.
Implementation in Progress	IMPL	Delink all except responder.
Implemented	IMPD	Delink all.
In Progress	WIP	Delink all except responder.
Known Error	KE	Delink all.
Not Approved	NOAP	Delink all.
Open	OP	Delink all except responder.
Pending Change	PNDCHG	Delink all except responder.
Problem Closed	PC	Delink all.
Problem Fixed	PF	Delink all except responder.
Problem Open	PO	Delink all except responder.
Reject Solution	REJSAP	Delink all except responder.
Rejected (Request/Incident/Problem)	PRBREJ	Delink all except responder.
Rejected (Change)	REJ	Delink all except responder.
Researching	RSCH	Delink all except responder.
Resolved	RE	Delink all except responder.
RFC	RFC	Delink all except responder.

Response	Response Code	Job Control
SA-Abandon	SAABND	Delink all.
SA-Resolved	SARES	Delink all.
Scheduled	SCHDLD	Delink all except responder.
Suspended	SUSPEND	Delink all except responder.
Vendor Hold	VNDHLD	Delink all except responder.
Verification in Progress	VRFY	Delink all except responder.
Verify Solution	VERSOL	Delink all.

Job control definitions

The job controls defined in the above table are implemented as follows:

- **Delivered**: marks the notification as delivered.
- **Notify next**: notifies the next recipient in the Group according to the defined escalation in xMatters.
- **Delink responder**: marks the notification as delivered. Stops any further action on the notification for the Responder ONLY.
- **Delink all except responder**: marks the notification as delivered, and stops any further action on the notification for all recipients of the notification EXCEPT for the responder.
- **Delink all**: marks the notification as delivered, stops any further action on the notification for all recipients, and terminates the event in xMatters

The job control defined for each response choice is the default configuration for this integration; for more information about job control, and how to modify these actions in the scripts, see the *xMatters Online Developer's Guide*.

4.3.1 Adding annotation messages

Two-way email Device notifications (not FYI) can add extra annotations that will be added to the CA SDM incident. To add an extra annotation, respond to an email notification with the following format in the subject line:

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

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

4.3.2 Changing and adding response choices

You can change the response choices by adding new statuses and transitions in CA SDM. For each new status you add, you must update the Response Handler section in the Action Script to handle the new response choice.

A new ELSE-IF block must be inserted into the handler to capture the response code that will be returned by the notification response. If the status change is to be reflected in CA SDM, the first call must be to sendExternalServiceRequest. Subsequent calls are for job control and can be configured as required using the job control descriptions above.

Response script

```
ELSE-IF ( $token == "newstatus" )       ### New Status
  CALL sendExternalServiceRequest
  CALL sendAPDelinkAllExceptResponderResponse
```

4.4 Delivery Annotations

This integration extensively annotates the originating CA SDM ticket for each Device to which a notification is delivered, but this may not be desirable in all environments. To prevent the delivery annotation of an incident, change the "annotatedelivery" Event Domain Constant to *false*. For more information, see "Configuring the Event Domain" on page 29.

4.5 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. This constant stores the number of seconds the notifications will be allowed to continue before timing out.

For example, if you wanted to change the event duration to two hours, you could change the value for the timeout constant to **7200**.

For more information about working with Event Domain Constants, see "Configuring the Event Domain" on page 29.

4.6 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 "Configuring the Event Domain" on page 29. Setting the **forcefyi** Event Domain Constant to "on" makes all normal and Subscription notifications one-way (FYI).

4.7 Filtering and suppression

The xMatters integration agent's Portable Filtering and Suppression Module is a built-in module that maintains a rolling record of previously injected events, and allows for the suppression of duplicates (also referred to as "deduplication"). This helps avoid disruption of traffic due to inadvertent loads that can result when, for example, improperly configured management systems inject duplicated events.

The `deduplicator-filter.xml` file is installed in the `<IAHOME>\conf` folder and is configured to suppress duplicate events for 30 minutes (up to a maximum of 100 events in that period).

This filter can be modified to extend the time period over which an event is considered to be a duplicate, the number of events in that period and the tokens that are used to determine what makes the event unique.

For example, to add REQUESTOR_NAME to the tokens, open the `deduplicator-filter.xml` file in a text editor and add the following line to the <predicates> collection:

```
<predicate>REQUESTOR_NAME</predicate>
```

Save the file and restart the integration agent for the changes to take effect.

Note: *To see a complete list of predicates available in the integration, reviewing the Event Data in the Event Summary Report in the xMatters web user interface.*

4.8 Configuring SSL

This integration supports SSL communication between the integration agent and CA SDM and between the integration agent and xMatters.

4.8.1 Using self-signed certificates

The SSL support has been configured out of the box to support self-signed certificates. This is not recommended for production systems due to security reasons, unless you are aware and accepting of the security implications of self-signed certificates.

To modify the SSL configuration:

1. Open the `<IAHOME>\integrationservices\caservicedesk-1-6-1\lib\lib_1_2_6\javascript\webservices\wsutil.js` file and modify the ACCEPT_ANY_CERTIFICATE variable as follows:
   - Set to *true* to use SSL but trust any certificate (including self-signed ones).
   - Set to *false* to accept only Certificate Authority (CA) certified certificates (recommended in production environments).

4.8.2 Importing certificates

The next step required to enable SSL support is to import the certificate used by the CA SDM web server to the cacerts keystore of the Java Virtual Machine (JVM) bundled with the integration agent.

Using the keytool executable located at `<IAHOME>\jre\bin`, execute the following command on the integration agent to import the certificate, replacing the variables with the appropriate values as described in the list below:

```
keytool -import -alias <your.alias> -file <path>/<certificate>.cer -keystore
<dir>/jre/lib/security/cacerts -storepass <password>
```

- **<your.alias>**: an identifier for the certificate within the keystore; for example, you can use the string "caservicedesk-1-6-1".
- **<path>**: path to the certificate
- **<certificate>**: the certificate's file name
- **<dir>**: the directory in which the integration agent is installed.
- **<password>**: the password for the cacerts keystore; the default password is "changeit".

If you want to configure SSL support between the integration agent and xMatters, use the above command to import the trusted certificate for xMatters into the integration agent keystore (for information on setting up SSL in xMatters, consult the xMatters Community site at http://support.xMatters.com

4.8.3 Updating HTTP to HTTPS

The configuration of HTTPS requires changes to three files. Each integration service has its own copies of these files, which are in the following locations within the `integrationservices/caservicedesk-1-6-1/` folder:

caservicedesk-1-6-1:

```
caservicedesk-1-6-1/configuration.js
lib/lib_1_2_6/javascript/webservices/wsutil.js
lib/lib_1_2_6/javascript/xmatters/xmattersws.js
```

casddataload-1-6-1:

```
casddataload-1-6-1/configuration.js
lib/lib_1_2_6/javascript/webservices/wsutil.js
lib/lib_1_2_6/javascript/xmatters/xmattersws.js
```

In the remainder of this section these files will be referred to by name without their path; the instructions can be applied to either or both of the integration services.

The next step is to update the SERVICE_DESK_URL in the `configuration.js` file to use the HTTPS protocol instead of HTTP.

The modified value should resemble the following:

```
var SERVICE_DESK_URL = "https://localhost:8443/axis/services/USD_R11_WebService";
```

Note: *For trusted certificates, "localhost" should be replaced with the COMMON NAME (CN) specified in the certificate and the port should be set to the port specified in the SSL configuration for CA SDM.*

To configure the integration agent to use HTTPS when communicating with xMatters:

1. In a text editor, open the `<IAHOME>\conf\IAConfig.xml` file.
2. Modify the URL for the <primary-servers> and <secondary-servers> elements to use the HTTPS protocol instead of HTTP; the section should resemble the following:

```
<primary-servers>
<!--
| 0 or more URL elements that specify the primary location of each xMatters server's
| RegisterIntegrationAgent Web Service.  The URLs must begin with either http:// or https://
| and cannot have a query or fragment component.  The URLs must be resolvable from this IA.
+-->
<url>https://localhost:8443/api/services/AlarmPointWebService</url>
</primary-servers>

<!--
| These servers are assumed to be connected to the same xMatters database,
| which can be different than the primary servers' database.
+-->
<secondary-servers>
<!--
| 0 or more URL elements that specify the secondary location of each xMatters server's
| RegisterIntegrationAgent Web Service.  The URLs must begin with either http:// or https://
| and cannot have a query or fragment component.  The URLs must be resolvable from this IA.
+-->
<url>https://localhost:8443/api/services/AlarmPointWebService</url>
</secondary-servers>
```

Note: *For trusted certificates, "localhost" should be replaced with the COMMON NAME (CN) specified in the certificate and the port should be set to the port specified in the SSL configuration for the xMatters server.*

3. Modify the value for the <service-gateway> element to use SSL; note that the service-gateway host IP must be resolvable from the xMatters servers:

```
<service-gateway ssl="true" host="localhost" port="8081"/>
```

4. Restart the integration agent.

4.8.4 Optional Configuration

The following scenarios illustrate the common configuration options available when using SSL.

Scenario 1

- CA SDM certificate: CA-certified
- xMatters certificate: CA-certified

In `wsutil.js`, set the variable ACCEPT_ANY_CERTIFICATE to *false*.

This will ensure ALL communication between the integration agent and CA SDM and the integration agent and xMatters uses the appropriate CA certified certificates

Scenario 2

- CA SDM certificate: CA-certified
- xMatters certificate: self-signed

In `wsutil.js`, set the variable ACCEPT_ANY_CERTIFICATE to *false*.

In `xmatterws.js`, add the following line at the end of the init() method:

```
this.ACCEPT_ANY_CERTIFICATE = true;
```

This will allow communication between the integration agent and xMatters to use self-signed certificates while maintaining more complete security between the integration agent and CA SDM.

Scenario 3

- CA SDM certificate: self-signed
- xMatters certificate: CA-certified

In `wsutil.js`, set the variable ACCEPT_ANY_CERTIFICATE to *true*.

In `xmatterws.js`, add the following line at the end of the init() method:

```
this.ACCEPT_ANY_CERTIFICATE = false;
```

This will allow communication between the integration agent and CA SDM to use self-signed certificates while maintaining more complete security between the integration agent and xMatters.

Scenario 4

- CA SDM certificate: self-signed
- xMatters certificate: self-signed

In `wsutil.js`, set the variable ACCEPT_ANY_CERTIFICATE to *true*.

This will allow ALL communication between the integration agent and CA SDM and between the integration agent and xMatters to use self-signed certificates.

4.9 Optimizing the data load integration

The following sections identify some of the ways you can adjust or modify the behavior of the data load integration to best suit your deployment.

Note that the data load integration has its own Event Domain, integration service, and script package. These must be installed using the instructions in the Installation and Configuration chapter, or manually installed as described in "Manually configuring xMatters" on page 29.

4.9.1 Mapping user roles

The default behavior of the data load integration is to assign the Role defined in the configuration.js file to all xMatters Users that it creates. For example, the default file uses the following code to assign the "Standard User" Role to all new xMatters Users:

```
var DEFAULT_XMATTERS_ROLES = ["Standard User"];
```

To assign a different Role, or to assign multiple Roles to all new or updated Users, you can modify the value of DEFAULT_XMATTERS_ROLES to include a comma-delimited list:

```
var DEFAULT_XMATTERS_ROLES = ["Role 1", "Role 2];
```

The integration also supports the more flexible assignment of xMatters Roles based on the roles associated with CA SDM users.

To map CA SDM roles to xMatters Roles:

1. In the configuration.js file, modify the value of the MAP_USER_ROLES variable to true; i.e.:

```
var MAP_USER_ROLES = true;
```

2. In the roleMap section, edit the roleMap lines to reflect the mapping you want to implement; e.g.:

```
1| var roleMap = [];
2| roleMap["Service Desk Manager"] = ["Group Supervisor", "Person Supervisor"];
3| roleMap["CA SDM Role"] = ["xMatters Role 1", "xMatters Role 2"];
```

Note how line 3, above, allows you to map a single CA SDM role to multiple Roles in xMatters. Any number of these one to many mappings can be specified, but the configuration does not allow you to assign multiple CA SDM roles to a single xMatters Role in a single roleMap entry.

Note: *Lines 2 and 3 can be modified, removed, or added to as needed, but do not remove or modify Line 1.*

4.9.2 Changing data load default values

The dataload integration service sets the values of some of the properties of xMatters Users, Groups, and Devices to hard-coded defaults. For example, Users are assigned by default to the "US/Pacific" time zone, and to the "Default Company".

The default values for any xMatters properties that are assigned in this way are defined in integrationservices\caservicedesk-1-6-1\casddataload-1-6-1\configuration.js and assigned to the associated xMatters object in the files and functions indicated below.

Object	Integration Service File	Function	Details
User	processUsers.js	makeUserForAddUpdate()	Sets User properties after a new User objects is created by calling new User()
Device	processUsers.js	makeUserForAddUpdate()	Sets property values for Devices other than Voice and Pager
	phonenumber.js	setVoicePhoneOrPager()	Sets property values for Voice and Pager Devices
Group	processGroups.js	makeGroupForAddUpdate()	Sets Group properties after a new Group is created by calling new Group()
Team	processGroups.js	makeGroupForAddUpdate()	Sets the property values for the Team created for Group objects
Team Member	processGroups.js	makeGroupForAddUpdate()	Sets the properties for Team members after new GroupMember() is called

4.9.3 Changing user device mapping

The makeUserForAddUpdate() function within the `processUsers.js` file examines the "udsObject" object that describes the user details in CA SDM, and maps the user devices to xMatters. The code that does this for a given device is a code block that typically resembles the following:

```
casdDeviceInfo = udsObject.email_address;
syncDevice = new EmailDevice(user.targetName, "Work Email");
syncDevice.isDelete = isEmpty(casdDeviceInfo);
syncDevice.address = casdDeviceInfo;
syncDevice.externallyOwned = user.externallyOwned;
```

```
user.devices.add(syncDevice);
```

To change the mappings between devices in CA SDM and Devices in xMatters, including a change to the name of the xMatters Device, add to or edit the relevant code block for the Device as needed.

4.9.4 Phone numbers and country code mapping

The code in setVoicePhoneOrPager() in `phoneNumber.js` maps the information in a phone number attribute in CA SDM to the phone number fields used by xMatters. It handles typical cases for 7-, 10-, and 11-digit North American phone numbers using simple regular expressions.

In the case of 11-digit numbers using a leading '1' as a long-distance dialing prefix, the prefix is mapped to an ISO 3166-1 two-character country code that xMatters requires.

The mapping from dialing prefix to Country Code Override is defined by the countryCodes variable in the `configuration.js` file:

```
var countryCodes =
  <countryCodes>
    <country>
      <dialingPrefix>1</dialingPrefix>
      <ISO_3166_1>US</ISO_3166_1>
    </country>
  </countryCodes>
```

You can extend the default configuration by adding <country> elements to the above definition. Note that each <country> element defines a one-to-one mapping. If more than one mapping uses the same dialing prefix (e.g., US and CA are both associated with the dialing prefix "1", the integration will use the first mapping it finds.

4.9.5 Specifying longer lists of users or groups to include/exclude

As described in "Data load configuration files" on page 15, the data load integration allows lists of users and groups to be excluded or included in the data load. However, if the list of users (or list of groups) contains more than approximately 50 entries, the resulting SOAP request to CA SDM will likely fail.

An alternative method to limiting the lists is to add a parameter to the "WHERE_CLAUSE_USERS" variable in the `configuration.js` file; for example:

```
" ... AND (notify_method = 12345)";
```

A clause similar to the above would cause users to be included only if their CA SDM user record included the "xmatters" notification method.

Note: *Your CA SDM database administrator would need to identify the actual clause that identifies applicable user records.*

Alternately, some clients elect to add a custom element to their CA SDM user interface. For example, a check box can be added to set a database flag in the user record, and the above "where" clause can then be modified to include or exclude user records according to this flag. You can engage an xMatters client services representative to assist in implementing the appropriate customization.

4.10 Uninstalling

For instructions on removing an xMatters deployment, refer to the *xMatters installation and administration guide*.

Chapter 5: Configuration Variable Reference

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

5.1 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.

Note that many of the configuration variables are configurable using the Event Domain Constants described in "Configuring the Event Domain" on page 29. Those variables are not listed here.

Gobal variables

$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.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.logo = $AlarmPoint_URL & "/static/images/logos/alarmpoint/UNKNOWN.png"	Specifies the path to the graphic displayed on HTML (email and BES) notifications.
$main.logo_alt_text = "[If the logo does not appear you may be blocking images or you may be outside a firewall. If the latter, the links will not work for responding and you should respond by replying to this email as described below.]"	The alternate text to display if the HTML email logo is unavailable. **Note**: If the logo does not display, it is unlikely that the HTML_ form_url 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.

www.xMatters.com

12647 ALCOSTA BLVD., SUITE #425 SAN RAMON CA 94583 USA | P: 1-877-xMattrs + 1.877.962.8877
CENTRAL COURT 25 SOUTHAMPTON BUILDINGS, LONDON WC2A 1AL UK | P: +44 (0) 800 652 7711