HP Propel Service Exchange

Software Version: 1.01
CentOS

Configuration Guide

Document Release Date: October 2014
Software Release Date: October 2014

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: **http://h20230.www2.hp.com/selfsolve/manuals**

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: **http://h20229.www2.hp.com/passport-registration.html**

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: **http://www.hp.com/go/hpsoftwaresupport**

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is **http://h20230.www2.hp.com/sc/solutions/index.jsp**

About this PDF Version of Online Help

This document is a PDF version of the online help. This PDF file is provided so you can easily print multiple topics from the help information or read the online help in PDF format. Because this content was originally created to be viewed as online help in a web browser, some topics may not be formatted properly. Some interactive topics may not be present in this PDF version. Those topics can be successfully printed from within the online help.

Contents

Configuration after VM install

HP Propel Service Exchange (HP SX) is the component that brings the Propel experience together, integrating the portal, catalog, and back-end fulfilment engines to enable functional integration with multiple providers. HP SX features built-in content that exchanges service messages and orchestrates and dramatically simplifies the integration of new and existing products, services and solutions.

To run the complex tasks it is capable of, HP SX needs some configuration. This involves some or all of the following:

- Add any additional HP SM or HP CSA instances you wish connected to HP Propel, see "Adding additional HP CSA instances " on page 9, and "Adding additional HP SM instances ".
- Set HP SX user roles and organizations to allow viewing of the HP SX management UI "Setting User roles and Organizations" on page 30.
- Run HP SX Self-test to verify that components are configured and connected correctly. It alerts administrators to any problems connecting to essential components, or with out-dated versions of customized files. **Self-test** is located on the top level of the HP SX management UI, see "Self-test HP SX configuration" below for details.
- If you are using HP CSA, check through "Connecting to HP CSA " on page 9 for any necessary LDAP and Approval configuration required.
- If you are using HP SM, check for further configuration steps required both in your HP SM instance and in HP SX, see "Connecting to HP SM " on page 11.
- Perform any customization requirements for message flows and operations in "Configuring Case Exchange" on page 45.
- View and manage HP SX content packs ("HP SX Content Management" on page 32) and adapters ("HP SX Adapters" on page 54).

NOTE: Throughout this guide, `[%SX_HOME%]` is used as a shortcut for the path: `/opt/hp/propel/jboss-as/standalone/deployments/sx.war/`

Self-test HP SX configuration

Use Self-test to check for correct configurations, connections and file version details.

From the HP Service Exchange administrator's UI, click **Self-test**. A script will run, testing for correct connections to:

- HP SX url
- HP IdM

- HP Propel catalog
- HP SM and HP CSA instances
- OO mappings
- AMQP configuration

For HP SM instances, Self-test also checks versions of all installed unload files, and lists them in the output file.

Example results:

Check configuration: csa/instances.json
Check instance CSAFTC : **OK**

Check configuration: sx.properties
Check security.idm% : **OK**
Check catalog.notification% : **OK**
Check sx.url : **OK**

Check configuration: sm/instances.json
----------**SMFortCollins**----------
Check instance SMFortCollins is accessible : **OK**
Check instance SMFortCollins - checking unload files ... :
Unload 'SXR2FCustomizations' in version '1.01.1' ... **OK**
Unload 'SXR2FDB' in version '1.01.1' ... **OK**
Unload 'SXR2FExtAccess' in version '1.01.2' ... **OK**
Unload 'SXTicketing' in version '1.01.2' ... **OK**
Unload 'SXCaseExchange' in version '1.01.1' ... **OK**
Unload 'SXBaseCustomizations' in version '1.01.1' ... **OK**
Unload 'SXBaseDB' in version '1.01.4' ... **OK**
Unload 'SXBaseExtAccess' in version '1.01.3' ... **OK**
Unload 'SXUnloadChecker' in version '1.01.1' ... **OK**

Check configuration: operationMappings.json
Check instance.targetInstance : **OK**

Check AMQP configuration
Check AMQP connection : **OK**
Check message listeners : **OK**

Check configuration: oo/properties.json
Check properties : **OK**

Check configuration: infrastructure.json
Check OO.endpoint : **OK**
Check REST.endpoint : **ERROR:** Invalid credentials
Check REST.operationEndpoint : **OK**
Check REST.csaNotificationEndpoint : **OK**
Check SERVICE_CATALOG.requestCallbackEndpoint : **OK**
Check SERVICE_CATALOG.subscriptionCallbackEndpoint : **OK**
Check SERVICE_CATALOG.internalCallbackEndpoint : **OK**

Connecting to HP CSA

Adding additional HP CSA instances

During install, the `installer.properties` file adds one HP CSA instance to the appropriate configuration JSON file. However, it is possible to connect any number of instances of HP CSA to HP SX. To add additional HP CSA instances the following file must be edited manually:

`[%SX_HOME%] /WEB-INF/classes/config/csa/instances.json`

Required fields: endpoint, loginName and password

Assuming CSAEurope1 was added through the `installer.properties` file during install, add additional instances as in the *CSAEurope2* example below.

NOTE: Enter your unique instance names, URLs, loginNames and passwords in place of the values in italics in the CSAEurope2 section below.

Example:

```
{
    "CSAEurope1": {
        "endpoint": "https://example1.com:8444/csa",
        "user": {
            "loginName": "johndoe",
            "password": "mypassword"
        },
        "organization": "CSA_CONSUMER"
    },

    "CSAEurope2": {
        "endpoint": "https://example2.com:8444/csa",
        "user": {
            "loginName": "janedoe",
            "password": "my2password"
        },
        "organization": "CSA_CONSUMER"
    }
```

IMPORTANT: For a specific module to use the correct instance, relevant configuration files must be edited to include the appropriate HP CSA instance names.

LDAP and Approval settings

For HP CSA to integrate with HP SX, LDAP and Approval settings need to be configured. If these are already set, further action is not required. If not, see:

- "Configure HP CSA to use LDAP" below
- "Configure HP CSA Approval settings" below

Configure HP CSA to use LDAP

1. Login to HP CSA.
2. Select **Organizations**.
3. Select **HP CSA Consumer**.
4. Select the **LDAP** section.
5. Fill in your LDAP server information and click **Save**.
6. Select the **Access Control** section.
7. Click **Add On**.
8. Fill in the AC Config and click **Update**.

Configure HP CSA Approval settings

1. Login to CSA.
2. Select **Catalogs**.
3. Create a new catalog.
4. Go to the **Approval Policies** section of the new catalog.
5. Fill in **Name**, select a **Template** (ie. Named Approver Template) and add **Approver**.
6. Save the policy.

Connecting to HP SM

Setting up HP SX to work with HP SM

Adding HP SM instances

During install, the `installer.properties` file adds one HP SM instance to the appropriate configuration JSON file. However, it is possible to connect any number of instances of HP SM to HP SX. To add additional HP SM instances the following file must be edited manually:

`[%SX_HOME%] /WEB-INF/classes/config/sm/instances.json`

Required fields: endpoint, loginName and password

NOTE:

- Previously there was a 'db' section in this file for direct JDBC connection. This is no longer supported and needs to be removed from the configuration file.
- Instances of HP SM with Process Designer installed must have the `withProcessDesigner` parameter set to **true**.
- The example below uses the default port number that HP Propel uses to communicate with HP SM. If you changed the port number, specify yours in the endpoint address in place of *13080.*
- See "Setting up HP SX to use LWSSO" on the next page concerning the `"useLwsso": true` line in the example.

Assuming SMEurope1 was added through the installer.properties file during install, add any additional HP SM instances as in the *SMEurope2* example below. Replace the values in italics with your unique urls, names and passwords.

Example:

```
{
    "SMEurope1":{
        "endpoint":"http://sm1.example.com:13080/SM",
        "user": {
            "loginName": "johndoe",
            "password": ""
        },
        "withProcessDesigner": true
    },
    "SMEurope2":{
        "endpoint":"http://sm2.example.com:13080/SM",
        "user": {
            "loginName": "janedoe",
            "password": ""
        },
        "useLwsso": true
    }
}
```

IMPORTANT: For a specific module to use the correct instance, relevant configuration files must be edited to include the appropriate HP SM instance names.

Setting up HP SX to use LWSSO

The above example shows that HP SX is configured to access the SMEurope2 instance via LWSSO. Note that it is not necessary to supply the password in this case. However, to make the LWSSO communication work, it is necessary that the file `[sx.war]/WEB-INF/classes/config/lwssofmconf.xml` contains proper LWSSO configuration that matches the target HP SM instance. In particular:

- The domain element must contain the common domain for HP SX and the target HP SM instance (default: hpswlabs.adapps.hp.com.)
- The initString attribute of the crypto element must contain the same passphrase as the HP SM instance.

Configure for ticketing

Users of a particular organization are only able to manage tickets on systems configured for that organization.

For ticketing REST API to use a certain instance, edits need to be made to the following file:

`[%SX_HOME%]/WEB-INF/classes/config/tenantInstanceMappings.json`

In `tenantInstanceMappings.json`:

- The `backendSystemType` and `instanceName` field values have to be set for each organization in the file.
- The DEFAULT values need to be added for all users whose organization is not specifically defined elsewhere in the file.

Example:

```
{
    <ORGANIZATION_NAME>: {
        "backendSystemType": "SM",
        "instanceName": "SMInstance1"
    },
    "DEFAULT": {
        "backendSystemType":"SM",
        "instanceName":"SMInstance2"
    }
}
```

In this example, SMInstance1 and SMInstance2 need to be the unique names previously defined in `instances.json`, and used for identifying these HP SM instances in other configuration files.

Configure Case Exchange

To set up Case Exchange functionality, there are a number of configuration steps necessary.

See "Configure HP SX" on page 46 for further details.

Setting up HP SM to work with HP SX

- "Create and apply new unload files" on page 24
- "Using Self-test to check unload versions " on page 26
- "Possible conflicts applying unload scripts " on page 27

Import SX Unload scripts

Necessary customizations of HP SM are performed by HP SM unload files. To import unload files into HP SM:

1. In your HP SM instance, go to **System Administration > Ongoing Maintenance > Unload Manager > Apply Unload**.
2. Select the Unload File: e.g. {*path-to-unload-file*}
3. Select **Backup To**: e.g. {*path-to-unload-file*}.backup
4. Click **Next**.

If there is a conflict with an entry, double-click that entry to see details, and look at "Manual configuration for Case Exchange" on page 18 to understand what customizations each HP SM unload pack contains, and "Possible conflicts applying unload scripts " on page 27. Consider checking all the changes made by the unload scripts to verify your HP SM configuration is correct.

HP SX Unload files

The following unload files contain fundamental HP SM customizations that are needed for HP SX to integrate with your HP SM instance.

SXBaseCustomizations.unl

Description: Customize operator "joe.manager":

- Sets role and profiles - to provide administrator level of rights to the joe.manager account.
- Sets max logins - because there can be more calls (logins) from SX concurrently and this SM can also be called from more SX machines.
- Adds "SOAP API" and "RESTful API" Execute Capabilities - to allow ANY remote SOAP and REST calls to joe.manager user.
- Sets password to: changeit

SXBaseDB.unl

Description: The triggers in the following entities:

- **cm3r (changes)**
- **subscription**
    - SX.subscription.delete
- **incidents**
    - SX.incidents.after.add
    - SX.incidents.after.update
    - SX.incidents.after.delete

...use the included scripts:

- SX_EntityChangeV2
- SX_SubscriptionDelete

...to write the triggered changes into these newly created tables:

- SxEntityChangesV2
- SxRegisteredEntitiesV2

SXBaseExtAccess.unl

Description: Provides remote interfaces (SOAP/REST) for the following tables:

- Change detection (see SXBaseDB.unl) - SxEntityChanges (read changes from SxEntityChangesV2 table), SxRegisteredEntities (write into SxRegisteredEntitiesV2 table when its necessary to be informed about changes in SM. These are then written into the SxEntityChangesV2 table.)
- Other functionality that is shared for Quotes, Changes and Ticketing features - for example, providing remote access to the following HP SM objects for HP SX: Relation (screlation), Cart Item (svcCartItem), Interaction (incidents), svcCatalog, Approval, operator and Attachment (SYSATTACHMENTS.)

SXTicketing.unl

Description: Changes for Ticketing feature. Customizes SOAP/REST interfaces:

- **SXGlobalLists** - provides access (add, delete, save) to globallists for HP SX.
- **SXActivityServiceMgt** - provides access (add) to activityservicemgt HP SM item for HP SX - exposes Activity Lines in Interaction Items. It is used for storing comments in Tickets.

- **SXTicketInteraction** - provides remote access to Interaction (incidents) HP SM items for HP SX - this is a duplicate of SXInteraction remote interface (with a line added to Expressions tab: $G.ess=true.) This is needed ONLY to escalate Tickets in HP SM.

SXCaseExchange.unl

Description: Changes for Case Exchange feature support: adding rest endpoints and table triggers.

- Adds new REST endpoint SX/SXCE_Incident - providing remote access to probsummary SM object for SX.
- Adds new REST endpoint SX/SXCE_IncidentActivity - providing remote access to activity SM object for SX.
- Adds triggers for tables:
    - **probsummary** - writes **Incident** changes into SxEntityChangesV2 table (see SXBaseDB.unl.)
    - **activity** - writes **Activity** changes into SxEntityChangesV2 table (see SXBaseDB.unl.)

SXChangesCustomizations.unl

Description: "Changes" feature support:

- Subscription Approval setup (sets joe.manager as the **only** approver, simplifies change category "Subscription" to 2 phases.)
- Change catalog item "Custom desktop Provisioning" (changes connector, adds user options.)

SXChangesDB.unl

Description: Database triggers for SX "Changes" support:

- SX.cm3r.after.update - trigger for writing Change-item updates into the SxEntityChangesV2 table (see SXBaseDB.unl.)

SXChangesExtAccess.unl

Description: SOAP/REST API updates SXSubscription and SXChange endpoints creation which are needed for the "Changes" feature. It provides access to Subscription and Change items from HP SM to HP SX, and includes support for the "force close" of a Change Item. This is needed when a Change item ends in an unexpected state, for example following a non-standard process.

SXQuotesCustomizations.unl

Description: "Quotes" feature support:

- Sets joe.manager as the only approver of "Manager Approval"
- Customizes approval process/lifecycle of Quote:
    - Removes 'Financial department approval' from the first phase (Front Line Management Approval.)
    - Removes 'Manager approval' from the last phase (Customer Follow-up.)
- **Interaction Approval** testing item (Performance Desktop) and Approval Process (SVC Off Catalog Request Approval.)

SXQuotesDB.unl

Description: Database triggers for SX "Quotes" support:

Adds the trigger "SX.ocmq.after.update" to write Quotes Item changes into the SxEntityChangesV2 table (see SXBaseDB.unl.)

SXQuotesExtAccess.unl

Description: SOAP/REST interfaces for SX "Quotes" support:

- Adds "SX_CloseQuote" Script Library - this script does the Quote Item closing.
- Adds "SX.CloseQuote" External Access Action. Adds CloseQuote action into SX/Quote remote interface. This action only executes CloseQuote Script Library.
- Adds SOAP/REST interface for SX/Quote - ocmq (Quote) SM Item access for SX.

SXQuotesTest.unl

Description: Data for Quotes feature testing:

- ScriptLibrary SX_MoveQuoteToCustFollowUpPhase and extractions id="SX.MoveQuoteToCustFollowUpPhase" - Force sets "Customer Follow-up" phase.
- SOAP/REST interface SX/SXQuoteForTests - remote interface for Quotes Testing, which includes Customer Follow-up phase enforcing.

Apply general unloads

Apply the following:

1. SXBaseCustomizations.unl (content-sx-base/src/main/resources/sm/SXBaseCustomizations.unl)
2. SXBaseDB.unl (content-sx-base/src/main/resources/sm/SXBaseDB.unl)

3. SXBaseExtAccess.unl (content-sx-base/src/main/resources/sm/SXBaseExtAccess.unl)
4. SXTicketing.unl (content-sm-ticketing/src/main/resources/sm/SXTicketing.unl)
5. SXCaseExchange.unl (content-sm-case-exchange/src/main/resources/sm/SXCaseExchange.unl)
6. SXUnloadChecker.unl (content-sx-base/src/main/resources/sm/SXUnloadChecker.unl)

Manual configuration for Case Exchange

Add **Add** activity privileges to the user account HP SX will use:

1. Go to **Tailoring > Format Control**.
   a. In the **Name** field add the string `activity` and click **Search**. The activity unload file contents will load.
   b. Open the tab (click the button) **Privileges**.
2. Change "false" to "true" for operation **Add**.
3. Click **Save**.

HP SM Process Designer - additional manual configuration

Configuration for Ticketing (for HP SM with Process Designer only)

Remove the `$G.ess=true` line from the Expressions tab of SXTicketInteraction Web Service:

1. Go to **Tailoring > Web Services > Web Service Configuration**
   a. In the **Object Name** field add the string "SXTicketInteraction" and click **Search**. The SXTicketInteraction settings will load.
   b. Open the tab (click the button) **Expressions**.
2. Remove the string `$G.ess=true`.
3. Click **Save**.

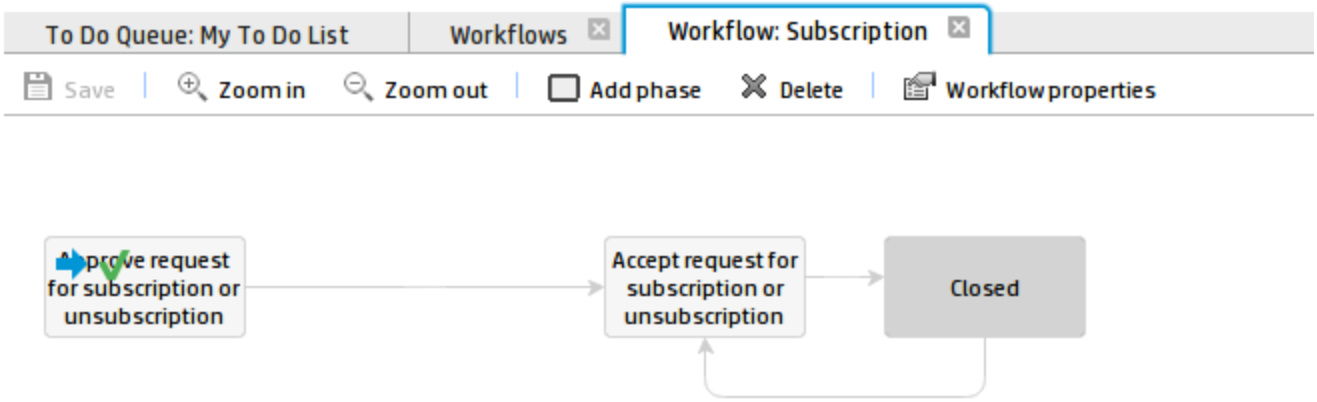
Configuration for Change R2F (for HP SM with Process Designer only)

1. Login to your web client.
2. Go to **Change Management > Configuration > Change Workflows**.
3. Select **Subscription** from the list.

4. Remove the second phase from the diagram.
5. Connect the first and third phases by relation.
6. Click to the new relation.
7. Fill the Command Name with "nextphase".

NOTE: Be careful not to remove anything beyond this.

The result of this step should look like this:



Apply R2F unload scripts

Apply the following request-to-fulfillment (R2F) unload scripts:

- **SXR2FCustomizations.unl** (content-sm-r2f/src/main/resources/sm/SXR2FCustomizations.unl)
- **SXR2FDB.unl** (content-sm-r2f/src/main/resources/sm/SXR2FDB.unl)
- **SXR2FExtAccess.unl** (content-sm-r2f/src/main/resources/sm/SXR2FExtAccess.unl)
- **SXTest.unl** (content-sm-test/src/main/resources/sm/SXTest.unl)

Manual configuration

1. Customize the approval process/lifecycle of Quotes

1. Go to **Request Management > Quotes > Quote Categories**, click **Search** and select the **Customer** record.

2. Click the first phase box (**Front Line Management Approval**) and remove 'Financial Approval' on the **Approvals** tab. Click **OK**. If the tab "Select Event For New Phase" opens, click the **Back** button.

3. Click the last phase (**Customer Approves Delivery of Item**) and remove 'Manager Approval' on the **Approvals** tab. Click **OK**. If the tab "Select Event For New Phase" opens, click the **Back** button.

2. Rebuild the "Extaccess Actions" Global List

NOTE: Use the HP SM client directly.

1. Go to **System Definition > Tables > globallist** and open it.

2. Click **View all records in the table**.

3. Select the line **Extaccess Actions**.

4. Mouse right click anywhere in the bottom part of the screen (the Item View panel), and select **Rebuild Global List**.

5. Click **Save**.

3. Modify the DEFAULT profile

NOTE: This step will not work if you have Process Designer installed.

1. Go to **System Administration > Ongoing Maintenance > Profiles > Service Desk Profiles** (Request Management Profiles on SM with PD.)

2. Click **Search** and select the **DEFAULT** profile.

3. Check the **Close** check-box.

4. Click **Save**.

Manual configuration - Approvals

NOTE: The following configuration is **not needed** if the HP SM instance is accessed via LWSSO.

Modify Change and Request profiles used by your Approvers

1. Login as Admin.

2. Go to **System Administration > Ongoing Maintenanace > Operators**.

3. Enter the login name and click **Search**.

4. Click the magnifying glass icon beside **Change Profile**.

5. Select the **Approvals/Groups** tab.
6. Check **Can Delegate Approvals**.
7. Click **OK**.
8. Click the magnifying glass icon beside **Request Profile**.
9. Change to **Alert/Approval** tab.
10. Check **Delegate Approvals**.
11. Click **OK**.
12. Select the **Startup** tab.
13. Change the parameter values in the first table in this way:
    1. name = MAIN MENU
    2. prompt =
    3. string1 = HOME
14. Click **OK**.

Delegate Change approving

NOTE: This step is only necessary if you have Process Designer installed.

1. Go to **System Administration > Operators**.
2. Fill **Login Name**: as "joe.manager", and click **Search**.
3. Add the "change approver" **Security Role** to joe.manager.
4. Click **Save**.
5. Go to **System Administration > Security > Roles**.
6. Select the change approver and click **Search**
7. Click the **Change** row.
8. Check **Can Delegate Approvals** under **Settings**.
9. Click **Save**.

Setup approval delegation for each Approver

1. Login as the Approver.
2. Go to **Miscellaneous > Approval Delegation**.
3. Click **Add New Delegation**.
4. Select **Delegate Selected Approvals**.
5. Click **Next**.
6. Select the module **Request Management**.
7. Click **Next**.
8. Move "jane.doe" to the right column.
9. Click **Next**.
10. Delegate to: johndoe. Fill in the Start and End dates.
11. Click **Next**.
12. Click **Finish**.
13. Repeat for the the **Change Management** module.

Manual configuration for Ticketing

NOTE: This is only necessary for Process Designer-enabled HP SM.

1. Login as Admin.
2. Go to **Tailoring > Tailoring Tools > Display Options**.
3. Enter `db.view_add` into the **Unique ID** field.
4. Change the condition from

   ```
   evaluate(add in $L.env) and filename($L.filed)~="dbdict" and
   nullsub($L.io.cond.flag, true)
   ```

   to

   ```
   (evaluate(add in $L.env) or evaluate(new in $L.env)) and
   filename($L.filed)~="dbdict" and nullsub($L.io.cond.flag, true)
   ```

5. Click **Save**.

Import Certificates

On the HP SX Machine

1. Download the HP SM certificate, for example by using the following command:

   ```
   openssl s_client -connect sm_host:8443 </dev/null | sed -ne '/-BEGIN
   CERTIFICATE-/,/-END CERTIFICATE-/p' >sm.crt
   ```

2. `keytool --import -file sm.crt -keystore /usr/lib/jvm/java-1.7.0-openjdk-1.7.0.55.x86_64/jre/lib/security/cacerts -alias sm_host`

3. Enter the password *changeit*.

4. Enter *yes*.

Setup LDAP

When LDAP has been configured in HP SM:

1. Identify the operator template used, by:

   - Go to **System Administration > Base System Configuration > Miscellaneous > System Information Record**, and note the **Operator Template** field. Follow this method if you used the second approach from the *LDAP Configuration Guide*.

   - Otherwise, go to **System Administration > Ongoing Maintenance > Operators** and search for the operator template, for example in LDAP.template or Operator.General

   If there is a value for **Contact ID**, remove it.

2. For non-Process Designer installations: add **Change manager** to the **Change Profiles**.
   For Process Designer-enabled installations: add **Change manager** to the **Security Roles**.

3. Switch to the **Startup** tab.

4. Into **Execute Capabilities** add **SOAP API** and **RESTful API**.

5. Click **Save**.

Temporarily, for every LDAP user perform the following:

1. Login into HP SM.

2. Logout.

3. Login as Admin.

4. Go to **System Administration > Ongoing Maintenance > Operators** and find your user.

5. Click **Create Contact**.
6. Select a contact to clone.
7. Click **Finish**.

Create and apply new unload files

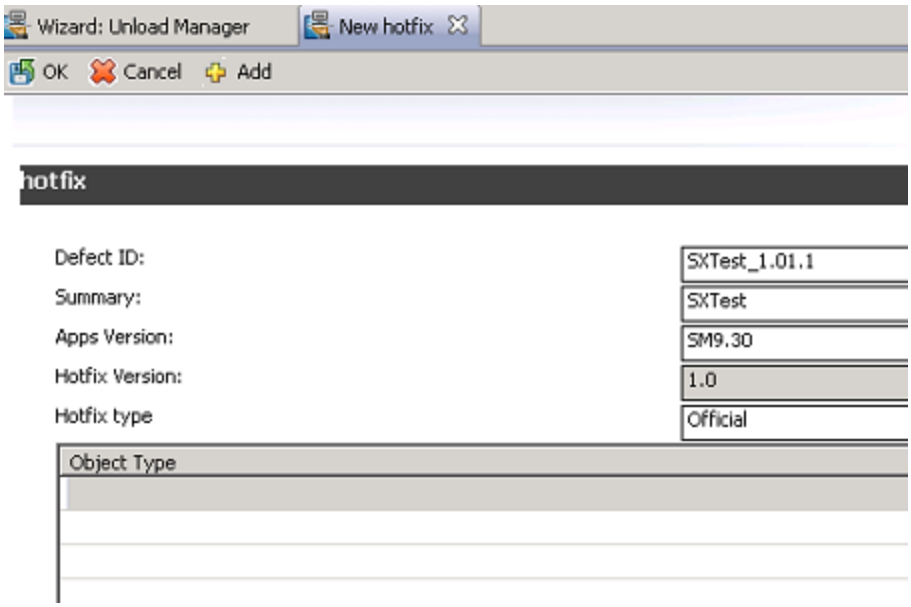
To enable HP SX to communicate with your HP SM instance, you may need to customize the out-of-the-box unload files, or create new ones.

If you develop your own HP SX content and make modifications to your HP SM instance, you will need to export these modifications into your own new unload file, or create a new version of an unload file. Using unload files enables you to backup changes and transfer them to other HP SM instances.

NOTE: After creating a new unload file, edit and then run **Self-test** to check that you have all the latest versions of files installed, see "Using Self-test to check unload versions " on page 26.

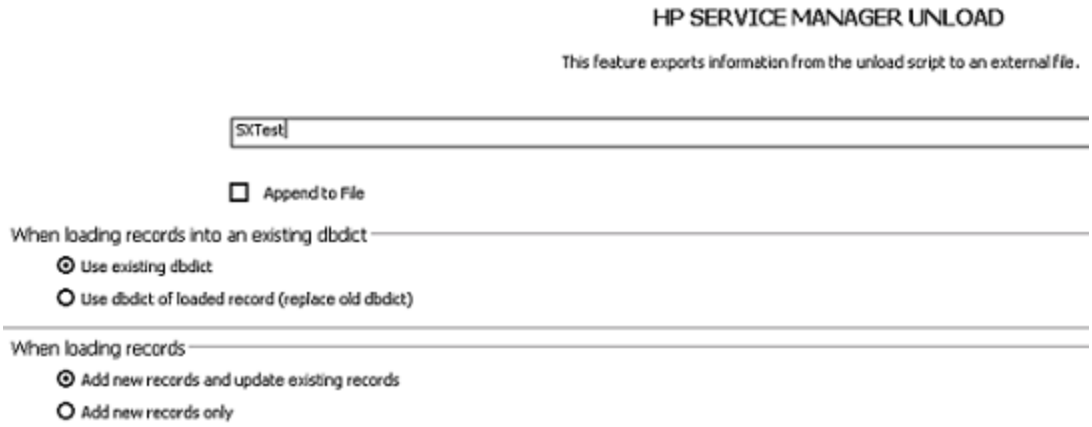
Creating unloads in HP SM Unload manager

1. In **System Administration > Ongoing maintenance > Unload manager** select **Create Unload**.
   - **Defect ID** - Use this field to hold the version of the unload file. Versions must be unique and follow this format: <*unl_file_name*>_<*version*>. For example: *SXBaseCustomizations_1.01.1*.
   - **Summary** - Enter a name for the unload file, without file extension.
   - **Apps version** and **Hotfix type** are not currently used. *SM9.30* and *Official* are chosen in the example below.



NOTE: the unload object does not need to be in the unload when using Unload manager.

2. Click **Add**.
3. The unload is created. The next step is to export it into a file. Click **Proceed**.
4. Export the unload to a file with the same value as entered in **Summary**.



5. Click **Proceed** and your new unload file is complete.

Apply unload in HP SM Unload manager

In **System Administration > Ongoing maintenance > Unload manager** select **Apply Unload** and follow the wizard instructions. After you finish you will see your unload with its version under **View Unload**.

NOTE: The table of applied unloads under **View Unload** is not updated automatically. Close and then reopen Unload manager to view your new unload.

Creating and updating version numbers

After implementing a change you may want to create a new version number for your unload.

To do so:

1. Double click your unload in **Unload manager > View Unload**.
2. Increment the version in the **Defect ID** field.
3. Export the unload into a file, see Step 4 above.
4. Apply the unload, see "Apply general unloads" on page 17.

Using Self-test to check unload versions

As well as checking for correct connections and configurations, HP SX Self-test checks the current versions of unload files, listing them in the output file. For this to happen, `SXUnloadChecker.unl` must be applied, and `metadata.json` edited to include all current unload file versions.

Before using the SX Self-test to check versions:

1. Apply `SXUnloadChecker.unl` to your HP SM instance.
2. Edit your `metadata.json` file, specifying any new unload files, and any new version numbers of unload files.

   For example:

```
{
    "id": "sx-base"
    "name": "Service Exchange base content",
    "description": "",
    "version": "1.0.0",
    "adapter": "SX",
    "features": [
    ],

    "files": [
       {
            "path": "sm/SXBaseCustomizations.unl",
            "version": "1.01.1",
            "type": "sm_unload"
        },
```

```
        {
            "path": "sm/SXBaseDB.unl",
            "version": "1.01.1",
            "type": "sm_unload"
        },

        {
            "path": "sm/SXBaseExtAccess.unl",
            "version": "1.01.1",
            "type": "sm_unload"
        }

    ]

}
```

3. Run the HP SX Self-test, see "Self-test HP SX configuration" on page 7.
4. Verify that the versions of unload files listed are your latest.

Possible conflicts applying unload scripts

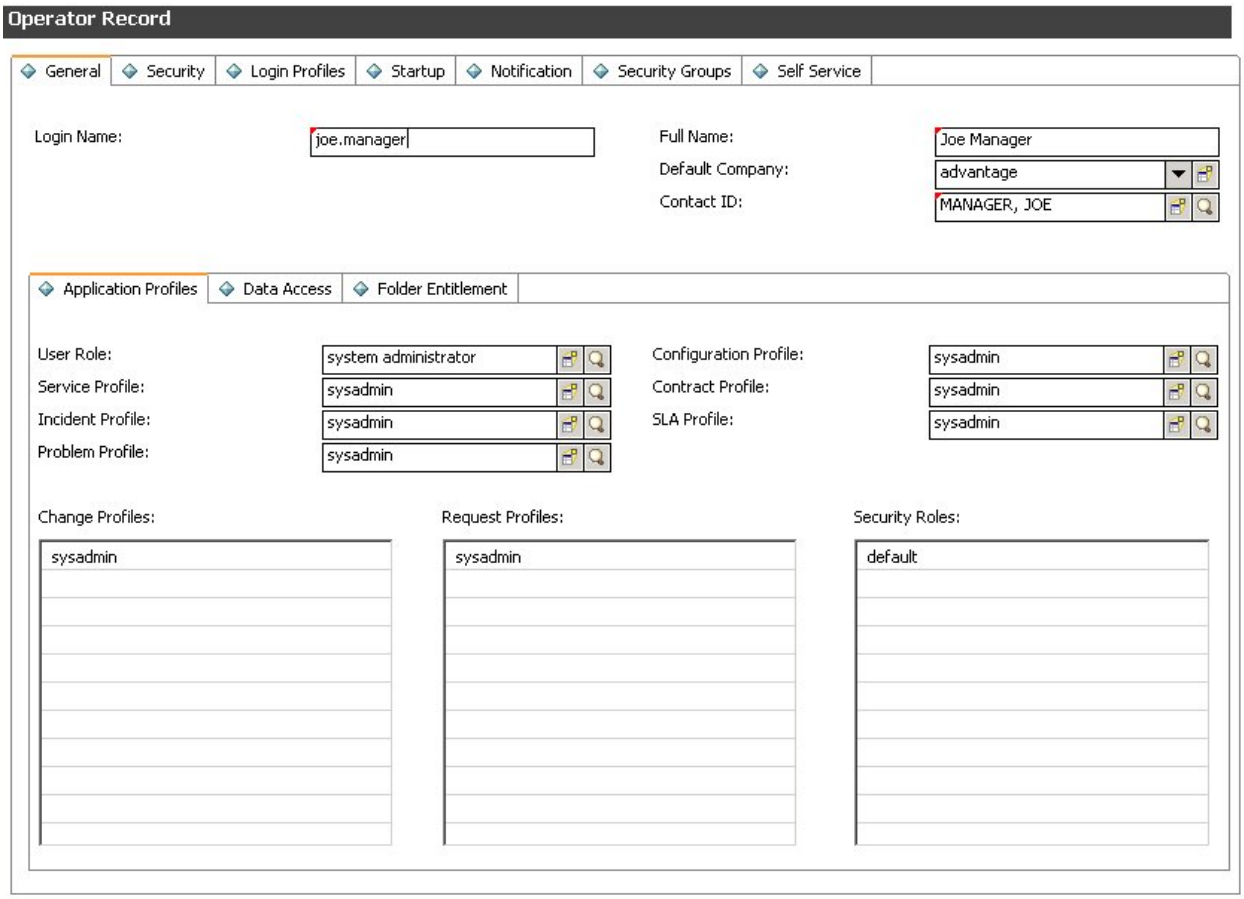
Applying any of the following three unload scripts can lead to conflicts with already existing HP SM settings:

- SXBaseCustomizations.unl
- SXChangesCustomizations.unl
- SXQuotesCustomizations.unl

Look at the parameters set by each of the HP SX customization files detailed here. This will help you avoid or resolve any conflicts with previously existing HP SM settings.

SXBaseCustomizations.unl

CAUTION: This unload script initiates a number of customizations. As they might conflict with already existing HP SM settings, HP recommends you review all changes made after running the script.



- The **Login Name** sets to joe.manager.
- On the **General** tab the **User Role** and each of the various **Profiles** are set (see screenshot above.)
- On the **Security** tab the **Max Logins** in the **User Session Information** section is set.
- On the **Startup** tab, in the **Execute Capabilities** section the words SOAP API and RESTful API are added.
- The password is set to changeit.

SXChangesCustomizations.unl

CAUTION: This unload script contains a number of customizations. As they might conflict with already existing HP SM settings, HP recommends you review all changes made after running the script.

This unload script initiates the following customizations:

- Sets joe.manager as the only approver of **Subscription Approval**, using **Change Management > Maintenance > Approvals**.

- Simplifies the change category **Subscription** to just 2 phases - **Approval** and **Acceptance**.
- Changes catalog item **Custom desktop Provisioning**:
    - Changes connector. Creates a change of type Subscription.
    - Adds user options (memory - number and model - Model 1 (model1), Model 2 (model2) Model 3 (model3) with prices $100,$200,$300.)

SXQuotesCustomizations.unl

CAUTION: This unload script contains a number of customizations. As they might conflict with already existing HP SM settings, HP recommends you review all changes made after running the script.

This unload script initiates the following customizations:

- Sets joe.manager as the only approver of **Manager Approval**.
  Change Management > Maintenance > Approvals
- Customizes approval process/lifecycle of Quote.
  Requests **Management > Quotes > Quote Categories**
    - Click on the phase box to get to phase details.
    - Removes Financial department approval from the first phase (**Front Line Management Approval**.)
    - Removes Manager approval from the last phase (**Customer Follow-up**.)

Setting User roles and Organizations

HP SX contains a set of management pages, for example the Content Management UI and the Testing UI, that are only accessible by users having certain roles. Users can have the following three possible roles:

- ADMINISTRATOR – An administrative User.
- UI – A general testing UI User.
- CONSUMPTION – This role is used for all API calls *from* and *to* the consumption component.

A user needs to be assigned an ADMINISTRATOR or UI role to access any of the HP SX UI pages.

A user needs to be assigned an ADMINISTRATOR role to access the HP SX management UI pages.

To assign or change HP SX user roles, see "Role association" below.

Role association

Users are associated with roles in the `users.json` configuration file, located at `[%SX_HOME%]/WEB-INF/config/users.json`

An example configuration:

```
{
    "Provider": {
        "sxCatalogTransportUser": {
            "roles": [
                "CONSUMPTION"
            ]
        },
        "admin": {
            "roles": [
                "ADMINISTRATOR"
            ]
        }
    }
}
```

The file structure reflects the organization structure. In this example the Provider is at the top level, with two users with their roles underneath. A user name can also be * which means all users within the organization will have the same roles. When multiple entries match a user all their roles are merged together.

Another example:

```
{
            "CONSUMER": {
        "admin" : {
            "roles": [
                "ADMINISTRATOR"
            ]
        },
        "*": {
            "roles": [
                "UI"
            ]
        }
    },
    "Provider": {
        "sxCatalogTransportUser": {
            "roles": [
                "CONSUMPTION"
            ]
        }
    }
}
```

In this example all the users of the CONSUMER organization have access to the testing UI for creating orders, and the admin user also has access to the administration section.

Selecting the Organization

When a user tries to access the HP SX UI using `http://sx_host:8444/sx` they need to log in, but are unable to select an organization as only one organization can be specified in the HP SX configuration.

To change the default organization:

- Open the `sx.properties` file at `[%SX_HOME%]/WEB-INF/sx.properties`
- Change the `security.idmTenant` property to match the organization name.

HP SX Content Management

Content packs are extension points to HP SX. A typical role of a content pack in HP SX is - in collaboration with SX Adapters - to enable HP SX to communicate with other remote systems (for example HP SM or HP CSA.)

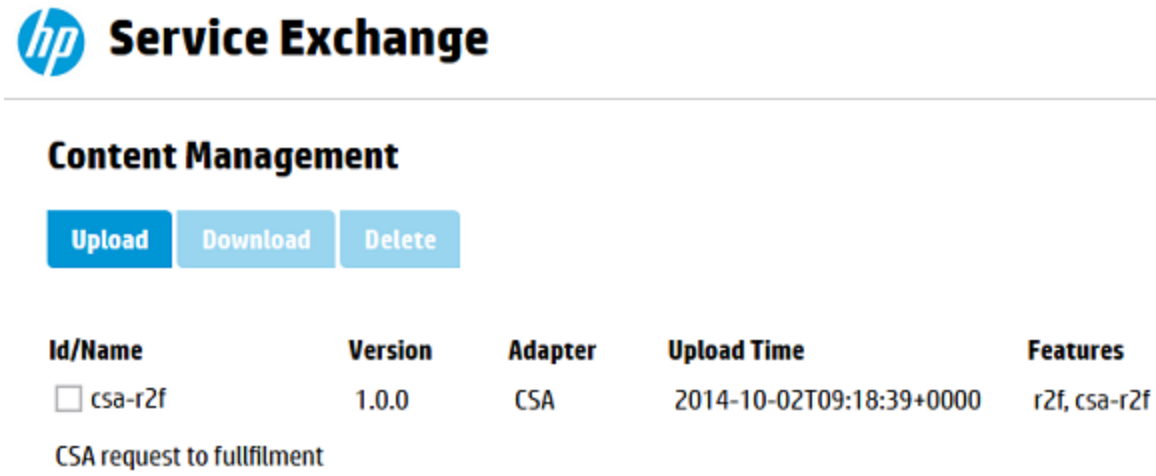
Technically a content pack is a zipped file containing operation definitions, Freemarker templates, OO flows and optionally other configuration files. HP SX offers out-of-the-box functionality through content packs that can be used as-is or customized. The HP SX Content Management UI provides an easy interface to view, download, upload and remove available content packs. Access to this UI is limited to users with the appropriate user roles, see "Setting User roles and Organizations" on page 30.

Upload and removal operations include the automatic upload or removal of relevant OO jar files, and the merging of HP SX customizations into the running HP SX server.

Using the Content Management UI

1. Click the Content Management link from the HP SX UI.
2. In the Content Management UI, view the available content packs and their details:
    - Version numbers
    - Which adapter they connect to
    - When last uploaded
    - Their high level features
    - The relevant OO content pack name.

Example section of the Content Management UI:



Downloading content packs

1. To download a content pack, check the appropriate content pack in the **Id/Name** column.
2. Click the **Download** button.
3. When prompted, **Save** the <*contentpack*>.zip. Depending on your browser settings, select the location through **Save As…** or copy the <*contentpack*>.zip from the **Downloads** folder to another location.
4. View and customize the files.

Deleting content packs

To delete one or more content packs:

1. Check the appropriate content packs in the **Id/Name** column.
2. Click the **Delete** button.
3. A confirmation with the number of content packs deleted appears, below the buttons on the top of the Content Management UI.

Uploading content packs

To upload a content pack:

1. Click the **Upload** button.
2. Locate the .zip or .jar to be uploaded, for example, the sm-case-exchange.jar containing a customized case-exchange.json file.
3. Select **Open**.
4. It takes a moment for the upload to process. When it is complete a confirmation appears below the buttons on the top of the Content Management UI and the Upload Time for the relevant content pack is updated.

NOTE: HP SX will automatically locate the appropriate content pack to upload to, it does not need to be selected or specified. The upload creates a new content pack, or if there is one with a matching ID already it replaces the existing one.

Content Packs and their contents

HP SX contains the following out-of-the-box content packs:

- **sx-base** - the base content for HP SX. This content pack is required and cannot be removed.
- **csa-r2f** - the content pack providing files for HP CSA requests to fulfillment (r2f.)
- **sm-r2f** - the content pack providing files for HP SM requests to fulfillment.
- **sm-ticketing** - the content pack providing files for HP SM ticketing.
- **sm-case-exchange** - the content pack providing files for HP SX Case Exchange customizations.
- **sm-test-ui-support** - the content pack providing files for HP SM related functions of HP SX UI.
- **csa-test-ui-support** - the content pack providing files for HP CSA related functions of HP SX UI.
- **mock-r2f** - an empty content pack.
- **email-r2f** - files to enable Email requests to fulfillment of native offerings.
- **sm-problem** - SM problem demo content.

Content packs contain the following folders and files:

- **[oo]** – folder containing HP OO content packs of custom OO flows.
- **[sx]** – folder containing HP SX specific configuration files.

    - [templates] - folder containing Freemarker templates
    - operations.json - file containing SX operation definitions
    - flows.json - file containing mapping of message type to OO flow
- **[sm]** – folder containing HP SM-specific unload (UNL) and customization files.
- **metadata.json** - the content pack description file.

You can create your own content packs, or modify the provided content packs to fit particular needs (for example, unique HP SM naming and process flows), and then upload them.

Using Case Exchange

Case Exchange is a feature of HP SX that offers easy management of underlying connections between entities within different HP SM (or other application) instances, ensuring that updates propagate appropriately. It is only necessary to integrate an application instance once with SX, thereafter all exchange of messages/cases are performed automatically through SX. This eliminates point-to-point integration and greatly simplifies IT management.

When configured, Case Exchange can listen out for entity changes where an entity in one HP SM instance is referring to an entity in another HP SM instance. If a referring entity is changed in one HP SM instance, HP SX is notified and registers a listener for entity changes in the other (referred) HP SM instance.

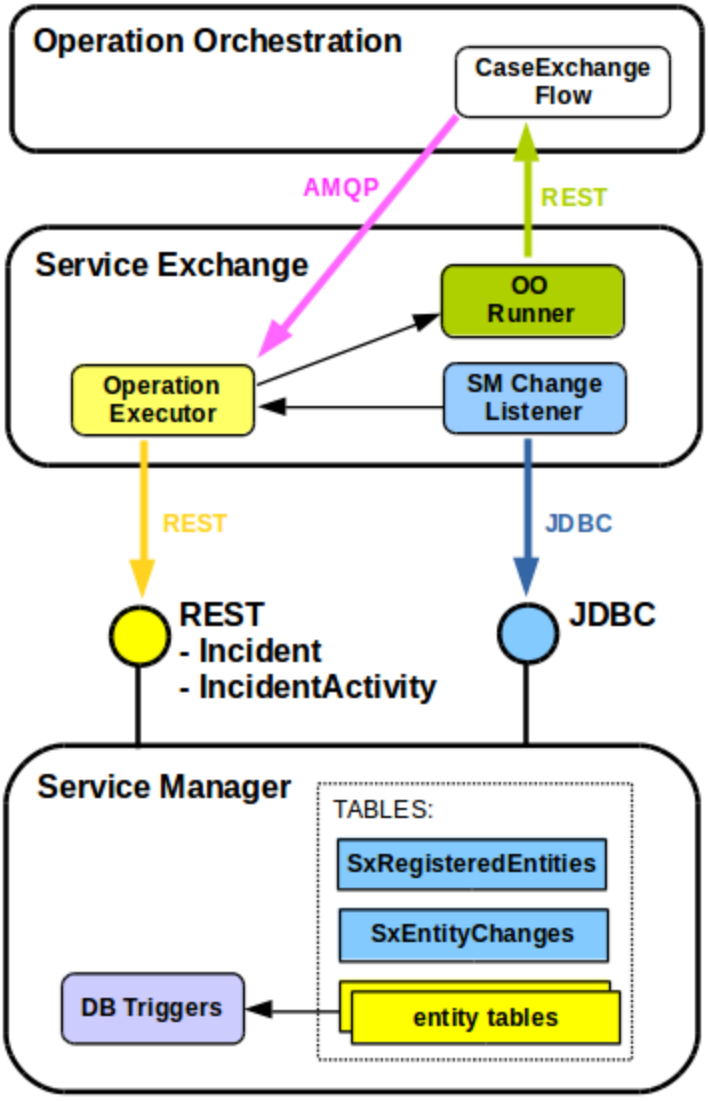
Examples of SX Case Exchange supported use cases:

1. An administrator can configure SX to be notified:
    - if an entity of type {entityType} complying with filter condition {entityFilterExpr} in external system {instance} is created/updated/deleted
    - if an entity of type {entityType} with id {entityId} in external system {instance} is created/updated/deleted.
2. An administrator can configure SX to be notified about an entity change in an external system then:
    - execute a custom OO flow
    - execute a custom SX operation.
3. OO flows can call custom (Case Exchange specific) SX operations to:
    - register/unregister new entity change listener in external systems
    - store mapping from entity {instanceTypeA}:{instanceA}:{entityTypeA}:{entityIdA} to {instanceTypeB}:{instanceB}:{entityTypeB}:{entityIdB}
    - remove mapping from/to entity {instanceType}:{instance}:{entityType}:{entityId}

See the following sections:

- "Components" on the next page
- "Implementing Case Exchange messaging" on page 41
- "Configuring Case Exchange" on page 45
- "Customizing Case Exchange" on page 47

Components



As depicted above, Case Exchange involves the following components:

- "Operation Orchestration" on the next page
- "HP Service Exchange" on page 39
- "HP Service Manager" on page 39

Operation Orchestration

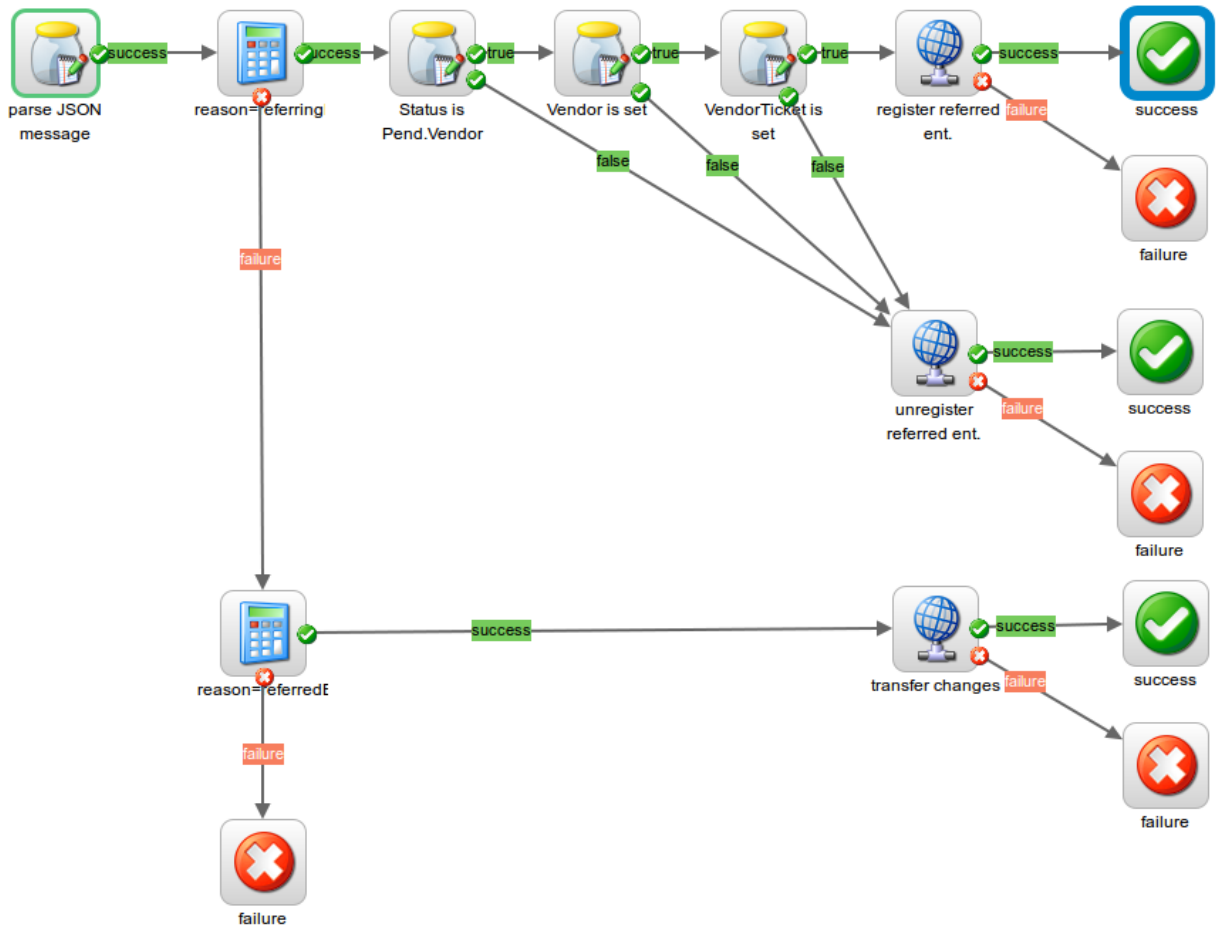
OO flows can be used when your change handling requires further logic to determine which actions (operations) should be performed.

The Flows available for execution are defined in the `flows.json` configuration file that is located in the `sx` directory in the relevant content pack.

A Flows configuration file defines the translation between the logical flow name (the name used in `case-exchange.json`) and the real flow ID. For each flow you can define parameters that will be passed to flow execution. All the given parameters in a flow can be used.

Each flow execution is given a complete message, which contains the full entity retrieved by the **retrieveSmEntity** operation.

Example Case Exchange OO Flow



This is a custom flow in OO containing the following logic:

if change reason is "referringEntity"

- if Status is “Pending Vendor” and Vendor and VendorTicket is set, then call operation registerEntityChangeListener
- else call operation unregisterEntityChangeListener

if change reason is "referredEntity" then call operation transferReferredEntityChanges

HP Service Exchange

Operation Executor

This component executes Case Exchange related operations:

- entityChangeOf_referringEntity
- entityChangeOf_referredEntity
- registerEntityChangeListener
- unregisterEntityChangeListener
- transferReferredEntityChanges

For details, see "Customizing Operations" on page 48.

OO Runner

This is the common component that runs the flow in Operation Orchestration.

HP SM Change Listener

This component periodically checks table SxEntityChanges in HP SM and if a new entity change is detected it executes operation "entityChangeOf_{changeReason}".

HP Service Manager

Table SxRegisteredEntities

This is the database table containing entity registrations for HP SX. If an HP SM entity registered in this table is changed, this change will be reported to HP SX.

The content of this table is written exclusively by HP SX.

Entity registration can be defined using:

- entity type
- entity ID
- filter expression

The table has the following structure:

idNotUsed (varchar 60)

entityType (varchar 60)

entityId (varchar 100)

entityFilterExpr (varchar 300)

reason (varchar 60)

subReason (varchar 60)

sxId (varchar 60)

Table SxEntityChanges

This is a database table containing changes of entities which were previously registered in SxRegisteredEntities. New entries to this table are added by DB Triggers and processed entries are deleted by HP SX.

The table has the following structure:

id (varchar 60)

entityType (varchar 60)

entityId (varchar 100)

changeType (varchar 30, values={create/update/delete})

status (varchar 60)

updatedOn (datetime)

reason (varchar 60)

subReason (varchar 60)

sxId (varchar 60)

DB Triggers

If an HP SM entity which is registered in SxRegisteredEntities is changed (created/updated/deleted), then a database trigger creates a new entry in SxEntityChanges.

Implementing Case Exchange messaging

Register entity reference in SM-A

In the following examples *SMInstance-demo-sm01* and *SMInstance-demo-sm02* would be the names of your HP SM instances.

Message created if a referring entity is changed in HP SM:

```
{
 "messageHeader" : {
    "messageType" : "entityChange",
    "backendSystemType" : "SM",
    "targetInstance" : " SMInstance-demo-sm01"
  },
  "@type" :

"urn:x-hp:2014:software:cloud:data_model:sx:entityChange",
  "entityType" : "probsummary",
  "entityId" : "IM10136",
  "reason" : "referringEntity",
  "subReason" : "vendorIsSet",
  "changeType" : "update"
}
```

Message after it is enhanced by entity details of operation entityChangeOf_referringEntity:

```
{
  "messageHeader" : {
    "messageType" : "entityChange",
    "backendSystemType" : "SM",
    "targetInstance" : "SMInstance-demo-sm01"
  },
  "@type" :


"urn:x-hp:2014:software:cloud:data_model:sx:entityChange",
  "entityType" : "probsummary",
  "entityId" : "IM10136",
  "reason" : "referringEntity",
  "subReason" : "vendorIsSet",
  "changeType" : "update",
  "entity" : {
    "Incident" : {
```

```
      "Area" : "performance",
      "Assignee" : "Incident.Coordinator",
      "AssignmentGroup" : "Field Support (Europe)",
      "Category" : "incident",
      "Description" : [ "d13" ],
      "Impact" : "3",
      "IncidentID" : "IM10136",
      "JournalUpdates" : [ ... ],
      "OpenTime" : "2011-06-02T08:17:32+00:00",
      "OpenedBy" : "johndoe",
      "ProblemType" : "incident",
      "Service" : "Intranet / Internet (Asia)",
      "Status" : "Pending Vendor",
      "Subarea" : "system or application hangs",
      "TicketOwner" : "johndoe",
      "Title" : "TEST 1",
      "UpdatedBy" : "johndoe",
      "UpdatedTime" : "2011-06-03T15:16:46+00:00",
      "Urgency" : "3",
      "Vendor" : "SMInstance-demo-sm01",
      "VendorTicket" : "IM10135"
    },
    "Messages" : [ ],
    "ReturnCode" : "0"
  }
```


Message constructed by CaseExchangeFlow to register the referred entity in HP SM:


```
{
 "messageHeader" : {
    "messageType" : "registerEntityChangeListener",
    "backendSystemType" : "SM",
    "targetInstance" : "SMInstance-demo-sm01"
  },
  "@type" :


"urn:x-hp:2014:software:cloud:data_model:sx:registerEntityChangeListener",
  "referringEntities" : [ {
    "targetInstance" : "SMInstance-demo-sm01",
    "entityType" : "probsummary",
    "entityId" : "IM10136"
  } ],
  "referredEntity" : {
    "targetInstance" : "SMInstance-demo-sm02",
    "entityType" : "probsummary",
    "entityId" : "IM10135"
  },
  "entityRegistration" : {
    "targetInstance" : "SMInstance-demo-sm02",
```

```
    "reason" : "referredEntity",
    "subReason" : "statusChange",
    "entityType" : "probsummary",
    "entityId" : "IM10135",
    "entityFilterExpr" : "ISUPDATE && OLDRECORD
['problem.status'] != NEWRECORD['problem.status']"
  }
}
```

Change referred entity in SM-B

Message created if a referred entity is changed in HP SM:

```
{
  "messageHeader" : {
    "messageType" : "entityChange",
    "backendSystemType" : "SM",
    "targetInstance" : "SMInstance-demo-sm02"
  },
  "@type" :
"urn:x-hp:2014:software:cloud:data_model:sx:entityChange",
  "entityType" : "probsummary",
  "entityId" : "IM10135",
  "reason" : "referredEntity",
  "subReason" : "statusChange",
  "changeType" : "update"
}
```

Message once it is enhanced by the entity details coming from operation entityChangeOf_ referredEntity:

```
{
  "messageHeader" : {
    "messageType" : "entityChange",
    "backendSystemType" : "SM",
    "targetInstance" : "SMInstance-demo-sm02"
  },
  "@type" :
"urn:x-hp:2014:software:cloud:data_model:sx:entityChange",
  "entityType" : "probsummary",
  "entityId" : "IM10135",
  "reason" : "referredEntity",
  "subReason" : "statusChange",
  "changeType" : "update",
  "referringEntities" : [ {
    "entityId" : "IM10135",
    "targetInstance" : "SMInstance-demo-sm02",
    "entityType" : "probsummary"
```

```
  }, {
    "entityId" : "IM10135",
    "targetInstance" : "SMInstance-demo-sm02",
    "entityType" : "probsummary"
  } ],
  "entity" : {
    "Incident" : {
      "Area" : "hardware",
      "AssignmentGroup" : "Hardware",
      "Category" : "incident",
      "Description" : [ "hhj" ],
      "Impact" : "3",
      "IncidentID" : "IM10135",
      "JournalUpdates" : [ ... ],
      "OpenTime" : "2011-06-02T07:46:38+00:00",
      "OpenedBy" : "johndoe",
      "ProblemType" : "incident",
      "Service" : "Intranet / Internet (South America)",
      "Status" : "Accepted",
      "Subarea" : "hardware failure",
      "TicketOwner" : "johndoe",
      "Title" : "SIMON - Referred entity",
      "UpdatedBy" : "johndoe",
      "UpdatedTime" : "2011-06-03T15:17:28+00:00",
      "Urgency" : "3"
    },
    "Messages" : [ ],
    "ReturnCode" : "0"
  }
}
```

Message constructed by CaseExchangeFlow to transfer changes from referred entity to referring entity:

```
{
  "messageHeader" : {
    "messageType" : "transferReferredEntityChanges",
    "backendSystemType" : "SM",
    "targetInstance" : "SMInstance-demo-sm02"
  },
  "@type" :

"urn:x-hp:2014:software:cloud:data_model:sx:transferReferredEntityChanges",
  "referredEntity" : {
    "targetInstance" : "SMInstance-demo-sm02",
    "entityType" : "probsummary",
    "entityId" : "IM10135",
    "Status" : "Accepted",
    "UpdatedTime" : "2011-06-03T15:17:28+00:00",
    "UpdatedBy" : "johndoe"
  },
```

```
  "reason" : "referredEntity",
  "subReason" : "statusChange"
}
```

Configuring Case Exchange

This requires two procedures:

- "Configure HP SM" below
- "Configure HP SX" on the next page

Configure HP SM

1. Apply unload script SXCaseExchange.unl

Find the SXCaseExchange.unl unload script inside the sm-case-exchange content pack.

To apply the SXCaseExchange.unl script into each of your HP SM instances, follow these steps:

1. Go to **System Administration > Ongoing Maintenance > Unload Manager > Apply Unload**.
2. Select **SXCaseExchange.unl**.
3. Select **Backup To:** and enter or select a backup location.
4. Click **Next**.
5. If there is a conflict with an entry, double-click that entry and manually resolve the conflict based on the description of what the unload script should do.
6. Click **Next**.

The unload script contains the following customizations:

- Adds new REST endpoint SX/SXCE_Incident
- Adds new REST endpoint SX/SXCE_IncidentActivity
- Adds triggers for the following tables:
    - probsummary
    - activity

2. Configure Activity privileges

Add activity privileges:

1. Open the HP SM client.
2. Go to **Tailoring > Format Control > <Name: activity> > Privileges**.
3. Change false to true for operation **Add**.
4. Click **Save**.

Configure HP SX

To set up Case Exchange functionality, one or more entries have to be added to this file:

`[content-sm-case-exchange]\src\main\resources\sm\case-exchange.json`

Configure case-exchange.json

1. Download the content pack `sm-case-exchange.zip` from sx/contentManagement.
2. Open the **sm** folder in the **sm-case-exchange** content pack.
3. Open the `case-exchange.json` file, and using the format used in the `case-exchange.json.example` file, replace *SMInstance1* with the name of your HP SM instance (SM-A) that contains referring entities.

IMPORTANT: Make sure that your HP SM instances have the same names as those entered in your HP SX `instances.json`file.

For example:

```
{
    " SMInstance1" : {
        "registeredEntities" : [
            {
                "reason" : "referringEntity",
                "subReason" : "vendorIsSet",
                "entityType" : "probsummary",
                "entityFilterExpr" : "(RECORD['vendor'] != null && RECORD
['reference.no'] != null) || (ISUPDATE && OLDRECORD['vendor'] != null && OLDRECORD
['reference.no'] != null)"
            }

      ]
},
```

Where SMInstance1 must be the name assigned to your HP SM instance in your `instances.json` file.

5. Add any additional HP SM instances (which contain referring entities) to the `case-exchange.json` file, using a comma to separate each instance. Use the example format in the `case-exchange.json.example` file.
6. Save and close the `case-exchange.json` file.

7. Upload the updated `sm-case-exchange.zip` back to HP SX through the Content Management UI, for details see "Uploading content packs" on page 33.

8. When the upload is finished HP SX will automatically register all entity change listeners defined in `case-exchange.json`

Contents of case-exchange.json

The content of the `case-exchange.json` configuration file above defines that HP SX is interested in entity changes occurring in HP SM instance SMInstance1:

- where the changed (create/update/delete) entities are of type "probSummary" (Incidents)
- where the changed (create/update/delete) entities have (or had before update) set fields "vendor" (Vendor) and "reference.no" (Vendor Ticket) fields
- and if such a change occurs then this change will be reported to HP SX with reason/subreason set to "referringEntity"/"vendorIsSet"

The filter expression in "entityFilterExpr" has a JavaScript format and the following variables can be used in them:

- ISCREATE - Boolean variable, set to TRUE, if the change operation was a creation of a new entity.
- ISUPDATE - Boolean variable, set to TRUE, if the change operation was an update of an existing entity.
- ISDELETE - Boolean variable, set to TRUE, if the change operation was a delete of an existing entity.
- RECORD - Map, containing the created/updated/deleted entity's fields. To access these fields use syntax RECORD['fieldName'].
- OLDRECORD - Map, if the operation is update, then it contains the entity's fields before the change.
- NEWRECORD - Map, if the operation is update, then it contains the entity's fields after the change.

Customizing Case Exchange

The reference Case Exchange implementation in the **sm-case-exchange** content pack demonstrates a working example, where Incidents (probsummary in SM) references Incidents in other (or even in the same) SM instance. If the referred Incident's status is changed then a new Activity line is added to the referring Incident.

This is just one example from the wide range of use cases you can manage using Case Exchange. To write your own Case Exchange implementation you can either modify the existing reference implementation, in the sm-case-exchange content pack, or create your own content pack with your own implementation.

Customizing Operations

HP SX Case Exchange reference functionality implements the following operations:

1. **Operations of type "entityChangeOf_<*changeReason*>"**

Where *changeReason* is the specific changeReason you specify.

Generally, if HP SX is notified of an entity change of reason <*changeReason*> then the operation with name "entityChangeOf_<*changeReason*>" is called.

The purpose of these "entityChangeOf_" operations is to load entity details from HP SM before the CaseExchangeFlow is invoked.

HP SX has two "entityChangeOf_<*changeReason*>" operations:

OPERATION "entityChangeOf_referringEntity".

- This operation is implemented by FTL templates inside the content pack.
- It is invoked if a referring entity in HP SM is changed. This operation queries HP SM through REST to load the changed entity's details.

OPERATION "entityChangeOf_referredEntity"

- This operation is implemented purely using FTL templates inside the content pack.
- Invoked if a referred entity in HP SM is changed. This operation queries HP SM through REST to load the changed entity's details.

2. **OPERATION "registerEntityChangeListener"**
   - This operation is an internal operation implemented in HP SX.
   - It registers a new entity change listener in HP SM.
   - It is invoked from CaseExchangeFlow OO Flow if a referring entity in HP SM is changed and the referring fields are filled out.
3. **OPERATION "unregisterReferredEntityListener"**
   - This operation is an internal operation implemented in HP SX.
   - It removes a registration entry from SxRegisteredEntities table in HP SM.

- It is invoked from CaseExchangeFlow OO Flow if a referring entity in HP SM is changed and the referring fields are cleared.

4. **OPERATION "transferReferredEntityChanges"**

   - This operation is an internal operation implemented in Service Exchange, but additionally it has parts implemented in FTL templates (in the content pack.)
   - The FTL template part of the operation adds a new Activity entry on the referring entity.
   - It is invoked from CaseExchangeFlow OO Flow if a referred entity in HP SM is changed.

5. **Operation "registerEntityChangeListener"**

An example message showing the message format (the values need to be edited to your unique values):

```
{
 "messageHeader" : {
    "messageType" : "registerEntityChangeListener",
    "backendSystemType" : "SM",
    "targetInstance" : "SM-B"
  },
  "@type" : "urn:x-hp:2014:software:cloud:data_
model:sx:registerEntityChangeListener",
  "entityRegistration" : {
    "targetInstance" : "SM-B",
    "reason" : "referredEntity",
    "subReason" : "statusChange",
    "entityType" : "probsummary",
    "entityId" : "IM10135",
    "entityFilterExpr" : "ISUPDATE && OLDRECORD['problem.status'] !=
NEWRECORD['problem.status']"
  }
}
```

or

```
{
 "messageHeader" : {
    "messageType" : "registerEntityChangeListener",
    "backendSystemType" : "SM",
    "targetInstance" : "SM-A"
  },
  "@type" : "urn:x-hp:2014:software:cloud:data_
model:sx:registerEntityChangeListener",
  "entityRegistration" : {
    "targetInstance" : "SM-B",
    "reason" : "referredEntity",
```

```
    "subReason" : "statusChange",
    "entityType" : "probsummary",
    "entityId" : "IM10135",
    "entityFilterExpr" : "ISUPDATE && OLDRECORD['problem.status'] !=
NEWRECORD['problem.status']"
  },
  "referringEntities" : [ {
    "targetInstance" : "SM-A",
    "entityType" : "probsummary",
    "entityId" : "IM10136"
  } ],
  "referredEntity" : {
    "targetInstance" : "SM-B",
    "entityType" : "probsummary",
    "entityId" : "IM10135"
  }
}
```

The message tells HP SX to:

- Register an entity change listener in SM-B with the following parameters:
  - reason=referredEntity
  - subReason=statusChange
  - entityType=probsummary
  - entityId=IM10135
  - entityFilterExpr="ISUPDATE && OLDRECORD['problem.status'] != NEWRECORD['problem.status']"
- (If referringEntities and referredEntity are set then) also save an entity mapping from (SM-A/probsummary/IM10136) to (SM-B/probsummary/IM10135) into the SX internal database.

6. **OPERATION "unregisterEntityChangeListener"**

An example message showing the message format (the values need to be edited to your unique values):

```
{
  "messageHeader" : {
    "messageType" : "unregisterEntityChangeListener",
    "backendSystemType" : "SM",
    "targetInstance" : "SM-A"
  },
  "@type" : "urn:x-hp:2014:software:cloud:data_
model:sx:unregisterEntityChangeListener",
  "entityRegistration" : {
```

```
    "targetInstance" : "SM-A",
    "reason" : "customReason",
    "entityType" : "probsummary",
    "entityId" : "IM10111"
  }
}
```

This message tells HP SX to:

- Unregister entity change listener SM-A/probsummary/customReason/IM10111 from SM-A
- (If property entityId is missing then) unregister all entity change listeners SM-A/probsummary/customReason from SM-A

7. **OPERATION "unregisterReferredEntityListener"**

An example message showing the message format (the values need to be edited to your unique values):

```
{
  "messageHeader" : {
    "messageType" : "unregisterReferredEntityListener",
    "backendSystemType" : "SM",
    "targetInstance" : "SM-A"
  },
  "@type" : "urn:x-hp:2014:software:cloud:data_
model:sx:unregisterReferredEntityListener",
  "referringEntities" : [ {
    "targetInstance" : "SM-A",
    "entityType" : "probsummary",
    "entityId" : "IM10136"
  } ],
  "entityRegistration" : {
    "reason" : "referredEntity"
  }
}
```

The message tells HP SX to

- Find existing entity mappings in the HP SX internal database from entity SM-A/probsummary/IM10136, and for each mapping found, to unregister the entity change listener from SM of its referred entity.

8. **OPERATION "transferReferredEntityChanges"**

An example message showing the message format (the values need to be edited to your unique values):

```
{
  "messageHeader" : {
    "messageType" : "transferReferredEntityChanges",
    "backendSystemType" : "SM",
    "targetInstance" : "SM-B"
  },
  "@type" : "urn:x-hp:2014:software:cloud:data_
model:sx:transferReferredEntityChanges",
  "referredEntity" : {
    "targetInstance" : "SM-B",
    "entityType" : "probsummary",
    "entityId" : "IM10135",
    "Status" : "Accepted",
    "UpdatedTime" : "2011-06-03T15:17:28+00:00",
    "UpdatedBy" : "johndoe"
  },
  "reason" : "referredEntity",
  "subReason" : "statusChange"
}
```

The message tells HP SX to

- Find all entity mappings in the HP SX internal database where the referred entity is (SM-B/probsummary/IM10135), and for each mapping found, to invoke operation transferReferredEntityChanges.

Customizing the OO Flow

See the example diagram "Example Case Exchange OO Flow" on page 38.

The CaseExchangeFlow from the reference functionality implements the following logic:

- if the change reason is "referringEntity"
  - if Status is Pending Vendor and Vendor and VendorTicket is set then call operation registerEntityChangeListener
  - else call operation unregisterReferredEntityListener
- if the change reason is "referredEntity" then call operation transferReferredEntityChanges

You can modify the OO Flow, for example:

- Which fields contain the reference to another entity (Vendor, VendorTicket.)
- In the case of a referred entity change, which fields are forwarded to the

transferReferredEntityChanges message (Status, UpdateTime, UpdatedBy.)

- Calls to other custom operations.

HP SX Adapters

HP SX adapters interact with underlying (backend) systems, making them accessible to HP SX processes. Examples of backend systems are HP SM, HP CSA, and SAP. In order for a backend system to be accessed by HP SX an adapter is required. In this way the adapters make the functionality of the backend systems available to HP SX clients, for example the HP Propel Catalog or Portal.

HP SX has a few out-of-the-box adapters:

- SX adapter - the internal SX adapter. This is always the first adapter and implements the HP SX case-exchange functionality.
- SM adapter - specifically for HP SM backend systems.
- CSA adapter - specifically for HP CSA backend systems.
- EMAIL adapter - this adapter enables the fulfillment by email of offerings created independent of third party products.
- MOCK adapter - for testing.

SX Internal Adapter

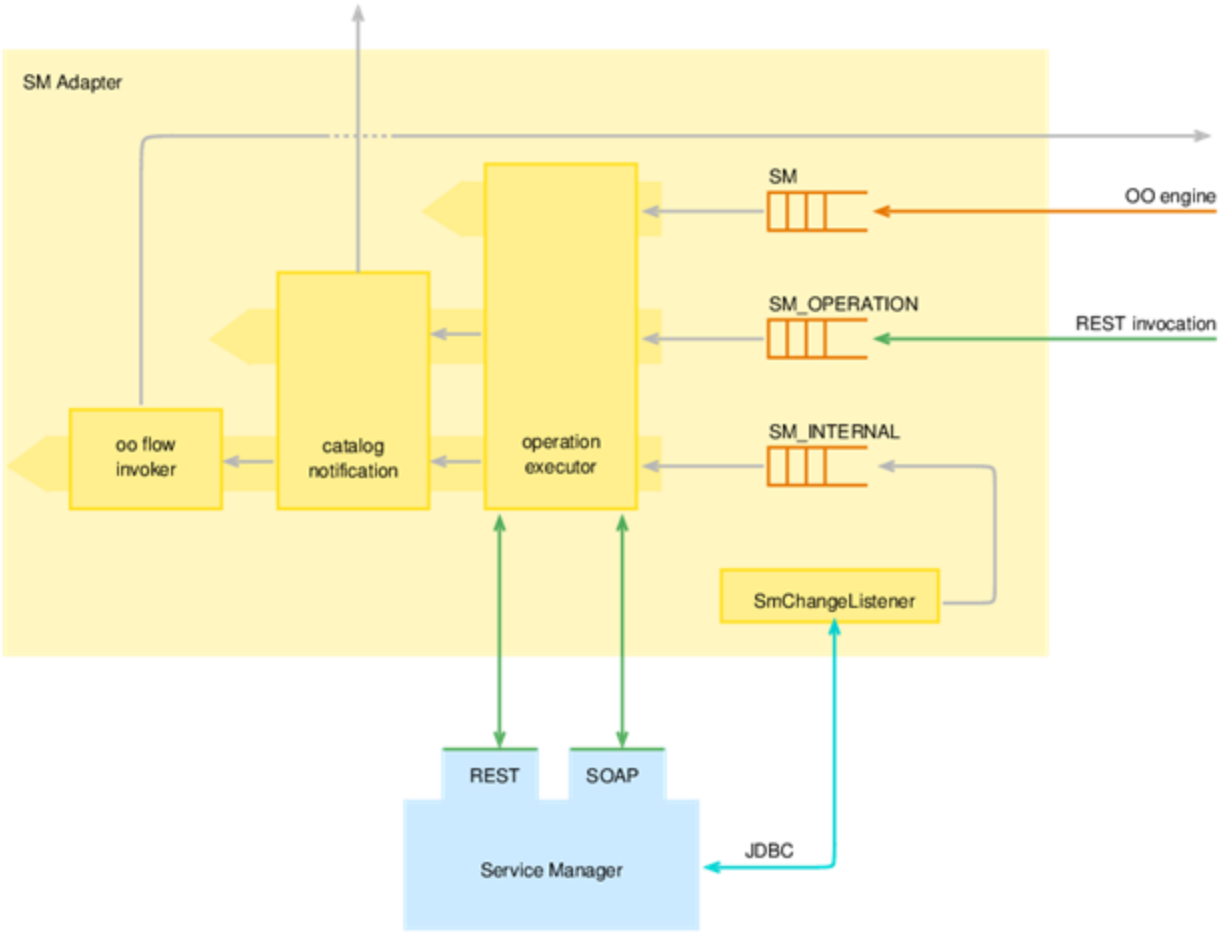
This adapter implements Case Exchange functionality which does not belong to any other existing adapter. It performs Case Exchange operations. The one pipeline (CASEEXOP), executes Case Exchange operations through CaseExchangeOperationExecutor.

See the following:

- "SX SM adapter" below
- "SX CSA adapter" on page 56
- "SX Email adapter" on page 57

SX SM adapter

The SM adapter enables HP SX to interact with HP SM. This diagram shows its architecture:



Inputs come from RESTful endpoint and OO engine into three 'pipelines' in the adapter:

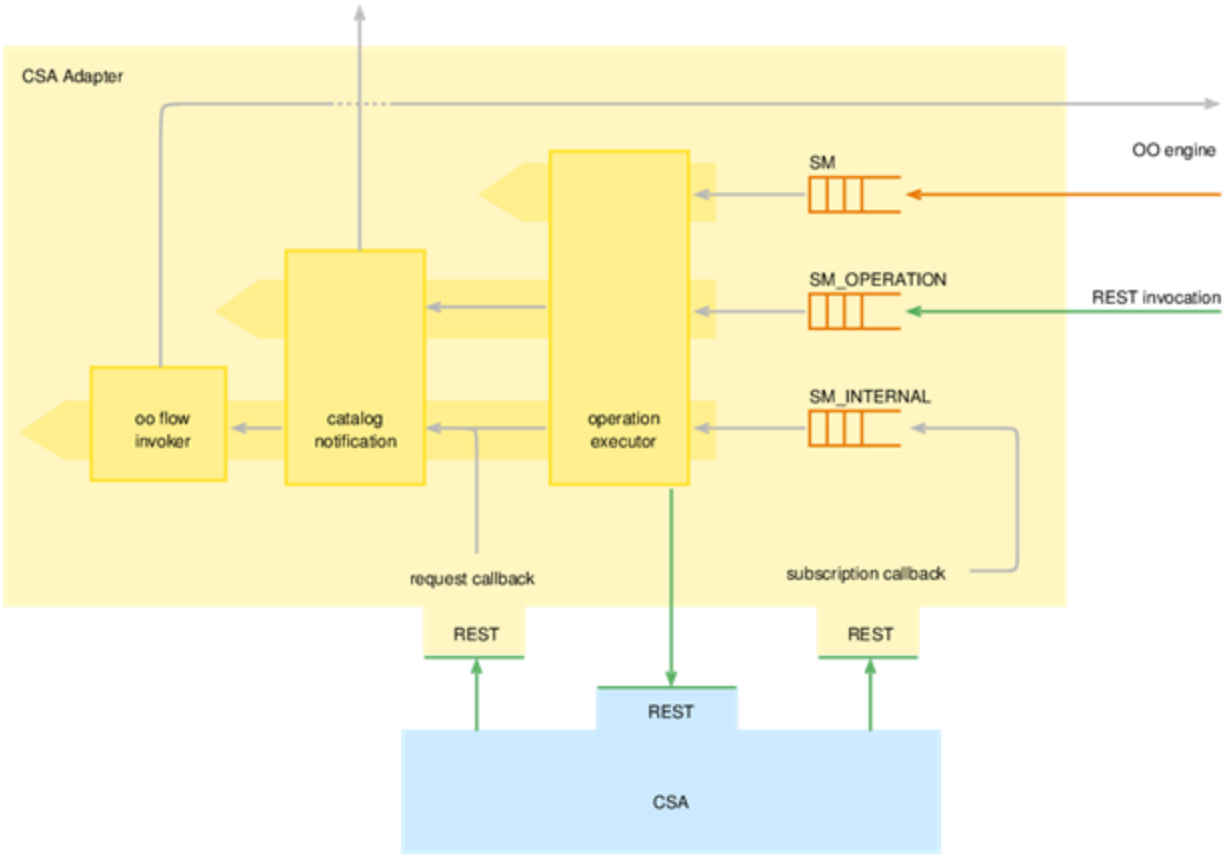
- SM:
  Messages from the OO engine result in operation execution only. Operation execution almost always involves an interaction (SOAP) with an HP SM backend system. Because the result of such an interaction is often asynchronous, the result/get notified back is checked via SmChangeListener.

- SM_OPERATION:
  Messages directly pass here via the HP SX non-blocking RESTful interface. Non-blocking, meaning that the RESTful client is just responded to with an operation-id (uuid.) The message is then processed asynchronously (because it is put into an AMQP queue.) When the message is read from the queue it results in operation execution again. Whether the execution fails or succeeds is reported back via catalog notification using the previously returned operation-id.

- SM_INTERNAL:
  This is executed if the SmChangeListener detects a change in an entity which was previously created/updated via operation executor. The change initiates another operation execution, catalog notification - how the execution went, and optionally the invocation of an OO flow (which could again result in an SM pipeline execution.)

SX CSA adapter

The SX CSA adapter enables HP Propel to interact with HP CSA. This diagram shows its architecture:



The inputs and three ‘pipelines’ are similar to the SM adapter. Yet, different from the SM adapter, the CSA adapter:

- Interacts with HP CSA via RESTful interface (not SOAP.)
- Notifies back via the adapter's RESTful callback endpoints (not JDBC connection.) These endpoints are global to the entire HP CSA, therefore it is impossible to use several unrelated HP SX instances on top of one HP CSA backend system. This is different to HP SM, where one can have several unrelated instances of HP SX on top one HP SM instance.
- When a request change is reported back to HP SX (request callback endpoint), no operation is performed. The catalog is notified that there was a request change and unconditionally invokes an OO flow.

SX Email adapter

HP SX makes it possible to fulfill requests concerning offerings in the HP Propel Portal that are created independent of third party products (such as HP SM, HP CSA etc.) These requests are handled by the SX Email adapter, which is delivered OOTB with HP SX.

This adapter makes it possible for HP SX to work with catalog items that are created 'natively', such that they can be fulfilled by a named email recipient clicking a yes/no link in an email.

Manually configure HP SX-required files

NOTE: If you followed the HP Propel Service Exchange installation procedure and have a functional system up and running, these configurations will already be in place.

Use the following instructions to check, troubleshoot or customize configurations:

- "Configuring for OO server" below
- "Configuring for RabbitMQ" on the next page
- "Configuring for the HP Propel Portal " on the next page
- "Configuring for IdM" on the next page
- "Configuring for PostgreSQL" on page 60

Configuring for OO server

To set an internal connection to a specific OO server add/edit the OO entry in the JSON file:

`[%SX_HOME%] /WEB-INF/classes/config/infrastructure.json`
Required fields: endpoint, loginName and password

Example:

```
{
    "OO": {
        "endpoint": "http://oo.example.com:8080/oo/rest",
        "loginName": "oouser",
        "password": "oopassword"
    }
)
```

NOTE: Change the endpoint, loginName and password to your unique values.

To enable the OO server to send email messages, change values in the JSON file:
`[%SX_HOME%] /WEB-INF/classes/config/oo/properties.json`
Example:

```
{
   "smtpServer": "smtp3.example.com",   "smtpPort": "25",   "mailFrom":
"noreply@example.com",   "emailBcc": "joe.doe@example.com"

}
```

Configuring for RabbitMQ

To set an internal connection to a specific RabbitMQ add/edit the JMS_BROKER entry in the JSON file:
[%SX_HOME%] /WEB-INF/classes/config/infrastructure.json
Required field: endpoint

Example:

```
{
    "JMS_BROKER": {
          "endpoint": "oo.example.com"
    }

}
```

NOTE: Change the endpoints to those for your organization.

Configuring for the HP Propel Portal

To enable communication with the HP Propel Portal, entry SERVICE_CATALOG has to be added/edited in the JSON file:

[%SX_HOME%] /WEB-INF/classes/config/infrastructure.json

Example:

```
{
    "SERVICE_CATALOG": {
        "requestCallbackEndpoint": "https://<CATALOG_
HOSTNAME>:8444/consumption/api/service/state/request",
        "subscriptionCallbackEndpoint": "https://<CATALOG_
HOSTNAME>:8444/consumption/api/service/state/subscription",
        "internalCallbackEndpoint": "https://<SX_HOSTNAME>:8444/sx/api/catalog"
    }
}
```

NOTE: Change the endpoints to those for your organization.

It is possible to use the string ${hpIPAddress} instead of a specific IP address of SX, but it is still required to add the server port manually.

Configuring for IdM

To use Identity Manager, the entry AUTHENTICATION has to be added/edited in the JSON file:
[%SX_HOME%] /WEB-INF/classes/config/infrastructure.json
Example:

```
{
    "AUTHENTICATION": {
        "secretKey": "<YourSecretKey>"
    }
```

Required field: secretKey

Endpoint and other information has to be set in the properties file:
`[%SX_HOME%] /WEB-INF/sx.properties`
Concerned lines contain the prefix **security**.

Configuring for PostgreSQL

To use your PostgreSQL installation, change properties with the prefix 'db' in the properties file:
`[%SX_HOME%] /WEB-INF/sx.properties`
Preset values:

- username=sx
- password=sx
- dbname=sx

Troubleshooting

General recommended steps

1. Run Self-test from the HP SX admin UI to check your connections are working and your configuration is correct, see "Self-test HP SX configuration" on page 7.
2. When a problem happens, go first to the relevant log files (for locations see "Log files" on the next page), and look for any sign of an error.
2. If no error is found, look into the OO Flows input parameters (see "OO Flows" below.) For example, look for wrongly set notification email addresses etc.
3. Use the SX Debug UI (see "SX Debug UI" on page 63.) If the functionality you tried to perform through the HP Propel Portal is available in the SX Debug UI, try to run it there. If it is successful, it is an important point to note in any defect report logged to HP Support. If it still does not work, check the UI error messages and see the log files for any changes in the error printouts, when compared with the HP Propel Portal execution.

Where to find help

- OO Flows
- Log files
- SX Debug UI
- HP SM item types supported by HP SX

OO Flows

If a problem occurs and you suspect the OO Flows did not execute properly:

1. Navigate to OO.

The URL of the OO used for HP SX will look similar to this:

```
http://oo_server_hostname:8080/oo/#/runtimeWorkspace/runs
```

2. Check the following:

a. Check that there is an entry in the **Run Management** section that corresponds to your request. View it.

b. Check that the Flow was executed properly. It is fine that it goes through failure transitions, but the Flow should not end in an error state.

c. If the flow ends in an error state, follow these steps to look for details in the Flow Input parameters:

i. When viewing the Flow, click on its header (where the Flow name is displayed together with a down-expand arrow.)

ii. You will see all the input parameters for the flow. Look for any suspicious or incorrect values, and make a note of them in case you need to report the issue later.

Log files

SX

Log files are located in the log directory of the JBoss server, where HP SX is deployed. Typically: `jboss-as/standalone/log`.
Below is a listing of HP SX log files:

Log file	Description
sx.log	All the HP SX debug and error messages. Search here first for error messages.
server.log **(for JBoss server)**	Look here during HP SX deployment when something happens during boot time. After booting up, all the HP SX related information is logged to the other HP SX log files.
notification.log	Contains all callbacks SENT TO Catalog (HP Propel Portal) Look here for possible explanations when having difficulty connecting to Catalog.
sx-messages.log	Contains all messages RECEIVED FROM Catalog (HP Propel Portal) Look here for possible explanations when having difficulty receiving from Catalog.
csa-messages.log **sm-messages.log**	Contains all callbacks sent to HP CSA and HP SM servers. Search here when experiencing problems connecting to these two server types.
case-exchange.log	Contains all Case Exchange module initialization messages, including database initialization, runtime message, and processing information regarding the Case Exchange feature. Could be helpful when Case Exchange seems not to be working for you.

OO
The OO Flows should be viewed in the OO UI, see "Where to find help " on page 61.

HP Propel MPP

Log file	Description
aggregation.log	All the aggregation related errors and information messages are logged here. Look into this file when you have a problem aggregating data from external servers.
ccue.log **consumption.log** **hpcloud-idm-service.log** **mpp.log**	See the HP Propel MPP Documentation for detailed descriptions of these log file contents.
server.log	Look here only when something happens during the HP Propel MPP deployment. After deployment all application related information is logged to other log files.

SX Debug UI

NOTE: See "Setting User roles and Organizations" on page 30 for information about the role settings that allow access to the SX Debug UI.

This Debug UI is useful when assessing possible causes of problems. By default the UI can be found at the following address:

`http: <sx_machine_hostname>:8081/sx/`

It contains the following functionality:

Create Order Wizard

Use to run a standard Item order in either HP SM or HP CSA instances (that are configured in SX.) Because ordering an Item via the HP SX Debug UI fetches the Offerings list directly from the remote HP SM and HP CSA instances, it is particularly useful, for example, when you have problems aggregating the offerings from HP SM and HP CSA instances. Select the desired instance first. Then select a Requestor (user) for whom you want to order the item, and then the Item required. Finally you can specify all the Parameters related to the Item being ordered.

Clear HP SM orders

Use to clear all HP SM orders for a given user (Requestor), and given HP SM Instance.

Tickets

Use Tickets to find the cause of problems with HP Propel MPP Support Tickets and if they are HP SX related. You can browse (sort, search), view and create Tickets here.

Notifications

Find both existing and completed HP SX Request and Subscription statuses here. You can also view detailed Subscription parameters when applicable.

Content Management

Here you can upload, download or delete content packs. See "Using the Content Management UI" on page 32 for details.

Administration

This is a debugging developer utility.

Self-test

A script that checks for correct configurations and latest upload file versions, see "Self-test HP SX configuration" on page 7.

HP SM item types supported by HP SX

Two Service Manager item types are supported by the current version of HP SX: Changes and Quotes. See the following for details:

- How to discern supported items in HP SM
- Quotes order processing in HP SM
- Changes order processing in HP SM

How to discern supported items in HP SM

Look at the item in HP SM, and view the details in the **Service Catalog > Administration > Manage Catalog** section under the **Connector Details** tab. To function correctly with HP SX an item should have the following attributes:

Changes:

- **Interface Type**: Open a Change
- **Create Subscription**: Checked

Quotes:

- **Interface Type**: Open New Request
- **Create Subscription**: *NOT* checked

Quotes order processing in HP SM

The expected process advancement of a Quote Offering order in HP SM is:

1. The Offering Item order is started and an Interaction is created in HP SM.
2. An interaction starts in **Open - Idle** status.

3. It then moves to the **Open - Linked** state, which indicates a Quote was created for the Item and it is now linked with the Interaction.
4. The Quote starts in the **Initial** state and its Approval Status is set to **Manager Approval**.
5. After an Approval, the Quote's status changes to **Ordering**.
6. Now all Line Items defined for this Quote (if any) need to be solved.
7. When all Line Items are solved, the Quote moves to the **Customer Follow-up** state.
8. When the Item is received by the Requester, he acknowledges receipt and the Quote moves to the **Closed** state.

- Find the **Interactions** in HP SM under **Service Desk > Interaction Queue > Search**.
- Find **Quote** details in HP SM under **Request Management > Quotes > Search Quotes**.

Changes order processing in HP SM

The Changes ordering functionality is similar to that for Quotes (see above), minus a few steps. See the process description in the following table:

	Change	**Interaction**	**Subscription**
Order by requester		Status: **Open - Idle** Approval Status: **Approved**	
After 30 - 60 seconds	Phase: **Subscription Approval** Status: **Initial** Approval Status: **Pending**	Status: **Open - Linked** Approval Status: **Approved**	Status: **Requested**
Approve by manager	Phase: **Subscription Acceptance** Status: **Initial** Approval Status: **Approved**		
Approve by requester	Phase: **Subscription Acceptance** Status: **Closed** Approval Status: **Approved**	Status: **Closed** Approval Status: **Approved**	Status: **Active**

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Configuration Guide (Propel Service Exchange 1.01)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to csa_propel_ie@hp.com.

We appreciate your feedback!


Java™
COMPATIBLE