HP Service Request Catalog

Software Version: 9.40
For the supported Windows® and Linux® operating systems

Customization Guide

Document Release Date: December 2014
Software Release Date: December 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

© 2010 - 2014 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.
Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.
Oracle and Java are registered trademarks of Oracle and/or its affiliates.
UNIX® is a registered trademark of The Open Group.
Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

For a complete list of open source and third party acknowledgements, visit the HP Software Support Online web site and search for the product manual called HP Service Manager Open Source and Third Party License Agreements.

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 indicates the release date of this version of the software.
- Document Date, which indicates the date the document is updated.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: https://softwaresupport.hp.com

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 website at: https://softwaresupport.hp.com

This website 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 website 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 website. 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 website is http://h20230.www2.hp.com/sc/solutions/index.jsp
Multi-company mode ................................................................. 33
Example: Multi-company mode ............................................... 34
Use the Service Manager software load balancer ...................... 35

Miscellaneous tasks ................................................................. 36
Enable Knowledge Management ............................................. 36
Configure subscriptions ......................................................... 37
Enable the approval delegation .............................................. 37
Improve SRC performance .................................................... 40
Additional performance tips .................................................. 47
Look-up fields ........................................................................ 48
Add additional contact information ....................................... 49
Define a Date/Time user selection .......................................... 49
Update Service Manager lists ................................................ 50
Supporting other locales ....................................................... 50
Authorizing bi-directional text .............................................. 57
Understanding an unexpected text order ............................. 57
Input bidirectional text in the Service Manager client ............ 59
Configure additional details for Service Requests .................. 60
Enable journal updates .......................................................... 61
Require users to read catalog item details .............................. 61
Enable the line item status ..................................................... 62
Configure IPv6 ...................................................................... 62
Enable FIPS for Service Request Catalog ............................... 62
Use the FIPS standard in Service Request Catalog ................. 63
Common Access Card introduction ....................................... 63
Enable SRC for Common Access Cards ................................. 64
Auto-suggest/Type ahead ...................................................... 67
Enable the Remember Login User Name feature .................. 67
Disable the Delivery Targets drop-down list .......................... 67

Appendix A: Launcher examples ............................................ 68
Example: Add a Bomgar live chat .......................................... 68
Create a Cloud Service Automation launcher ....................... 69
Example: Get ...................................................................... 71
Appendix B: Service Catalog in Service Manager ........................................ 72
  Dynamic user options in Catalog Items ............................................. 72
    Create a dynamic user option ..................................................... 72
  Notes on expressions in dynamic fields ........................................ 73
    Example: Create two dependent Dynamic User Options .................. 77
  Multiple selections for Service Catalog item user options ................ 80
    Add multiple user selections ..................................................... 80
  Service Request Catalog HTML tag support .................................... 82
    Refining the Service catalog .................................................... 85
      Remove empty categories ....................................................... 85
      Add images for catalog items .................................................. 85
      Separate catalog items for different companies in multi-company mode 85
  Working with the Support catalog ................................................. 87
    Search for a support catalog category or item ............................. 87
    Create a support catalog category ............................................. 87
    Create a support catalog item .................................................. 89
    Delete a support item .............................................................. 91
    Controlling the permission to create, update, or close interactions ... 91
    Disable the permission to create, update, or close interactions ....... 92

Appendix C: XML tag reference ......................................................... 94

Appendix D: Styling the interface ...................................................... 100
  Client customization reference .................................................... 103
    Customize the SRC header ....................................................... 104
    Customize the SRC footer ....................................................... 105
    Customize the Dashboard ......................................................... 106
    Customize the Services page .................................................... 107
    Customize the Support page ..................................................... 108
    Customize the Account View page .............................................. 109
    Customize the Details View page .............................................. 110
    Customize the Approval history ................................................. 111
    Customize the Request Inbox Welcome page ................................ 112
    Customize the Browse Catalog button ...................................... 112
    Customize Your Subscriptions ................................................... 113
    Customize the Approvals page .................................................. 115
    Customize the Service Approval Details page ............................... 116
    Customize the Change Approval Details page ............................... 117
Overview

There are many aspects of Service Request Catalog which are customizable to your organization. The contents of this Customization Guide are to help you reference the information you are looking for quickly. It is not intended as a step-by-step guide. Instead, each section provides loosely coupled concepts to assist you in achieving an immediate goal, such as enabling knowledge management in Service Request Catalog, configuring approval rights for a user, or customizing the user interface.

Configuration files

There are configuration files that enable you to take advantage of the customization features. From the installation directory structure on the server where you installed Service Request Catalog, the following files are important when you design your customization. You will define a company folder to contain some of these resource files.

<table>
<thead>
<tr>
<th>Configurable files</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File path and name</strong></td>
</tr>
<tr>
<td><img src="...%5Csrc%5Csecure%5Cconfiguration%5Cmanifest.xml" alt="Path" /></td>
</tr>
<tr>
<td>Identifies customization resources and settings.</td>
</tr>
<tr>
<td><img src="...%5Csrc%5Cresources%5Cclient%5Cen%5CCustom.properties" alt="Path" /></td>
</tr>
<tr>
<td><img src="...%5Csrc%5CWEB-INF%5Cclasses%5CApplicationContext.properties" alt="Path" /></td>
</tr>
<tr>
<td><img src="...%5Csrc%5CWEB-INF%5Cclasses%5CcacConfiguration.properties" alt="Path" /></td>
</tr>
<tr>
<td><img src="...%5Csrc%5Csecure%5Cconfiguration%5Cdefault%5Cpanel.swf" alt="Path" /></td>
</tr>
<tr>
<td><img src="...%5Csrc%5Csecure%5Cconfiguration%5Cdefault%5Cstyles.swf" alt="Path" /></td>
</tr>
</tbody>
</table>

**Note:** Styles.swf must be generated from the styles.css file. For instructions on how to do this, see "Customize the text format" on page 146.

Working with the manifest file

This file contains information about the presentation of the Dashboard and other landing pages for your company. This file also points to other resources that contribute to the look and feel of your user
interface, such as your company name, color palette, and other presentation styles.

The manifest.xml file has basic sections for capturing this information.

```xml
<configuration name="">
  <companyDisplayNameKey>
    <resources>
      <styles>
        <launchersGroup>
        <widgetsGroup>
      </configuration>

See the "Appendix C: XML tag reference" on page 94 for more information about configuring the individual tags in each section.

Use the validator tool for manifest.xml

Service Request Catalog comes with a prepackaged validation tool that you can use to help ensure that your manifest.xml file is valid and usable. The validator tool can be found in the Service Request Catalog release .zip package. The vtool consists of the following two script files for Windows and Linux operating systems:

- validator.bat
- validator.sh

For example, you would enter the following command on a Windows system to use the validation tool:

```
validate.bat -c "<path>"
```

**Note:** The `<path>` placeholder tag represents the file system path where you have the manifest.xml file.

After running this command, you will receive the following message if your manifest.xml file is valid:

"manifest.xml is valid!"

If your manifest file is not valid, you will receive an error message. The following table demonstrates some of the error messages you may receive:

<table>
<thead>
<tr>
<th>Error: Cannot read manifest file: <code>&lt;path&gt;</code></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.xml.sax.SAXParseException: cvc-datatype-valid.1.2.1: 'abc' is not a valid value for 'integer'.</td>
<td>A string value is present in some elements that requires an in integer value.</td>
</tr>
</tbody>
</table>
Using the Migration Tool

The Service Request Catalog Migration Tool is a dedicated tool to preserve your Service Request Catalog customizations as you migrate to HP Service Manager.

What does the Migration Tool do?

Some files in Service Request Catalog are open for customization. Without the Migration Tool, you must migrate any customizations manually to complete the migration process. Doing this manually exposes the process to potential mistakes. The Migration Tool copies the customized files to the new Service Request Catalog release directly and merges certain customized files into the Service Request Catalog 9.40 release.

Migration Tool usage

$TOOL_HOME refers to the folder to which you extracted this tool.

$SOURCE_SRC refers to the folder in which the older version of Service Request Catalog is deployed and which contains the customizations you wish to migrate.

$TARGET_SRC refers to the folder in which you are deploying the new version of Service Request Catalog.

Linux usage:
To use the migration tool in a Linux operating system, follow these steps:

1. Grant execution permission for migrateSRC.sh. To do this, navigate to $TOOL_HOME and then run the following command:
   
   ```bash
   chmod u+x bin/migrateSRC.sh
   ```

2. Launch the Migration Tool by running the following command:
   
   ```bash
   bin/migrateSRC.sh $SOURCE_SRC $TARGET_SRC
   ```

3. Follow the onscreen instructions to migrate the specific migrations you wish to move to the new version of Service Request Catalog.

**Windows usage:**

To use the migration tool in a Windows operating system, follow these steps:

1. Double-click the migrateSRC.bat file in the $TOOL_HOME\bin\ directory.

2. Follow the onscreen instructions to migrate the specific migrations you wish to move to the new version of Service Request Catalog.

**Note:** Currently, you can use this tool to migrate from Service Request Catalog 1.40, or Service Request Catalog 9.32, or Service Request Catalog 9.33, or Service Request Catalog 9.34 to Service Request Catalog 9.40.
Service Request Catalog customization

You can make many changes to brand Service Request Catalog with your own corporate identity. Service Request Catalog allows you to customize the interface through a variety of mechanisms. You can show or hide any of these five primary widgets, or use the launcher to add links to third party Web sites, applications, videos, or anything else you prefer. Read the following sections to learn more about each task:

Interface

Embed SRC in an HTML frame ................................................................. 11
Customize the landing pages ............................................................... 12
Use the launcher to create third party links ............................................. 14
Customize error messages .................................................................... 17
Hide the service or support tabs ............................................................. 18
Configure the Your Requests and Your Service Request .......................... 18
Customize the application name ............................................................. 19
Add a new language ................................................................................ 20
Configuration steps ................................................................................ 20
Configure session timeout ...................................................................... 22

For information on how to customize the colors, background, and other graphical related elements of Service Request Catalog, see "Appendix D: Styling the interface" on page 100.

Embed SRC in an HTML frame

If you wish to embed Service Request Catalog within your corporate intranet, you can embed Service Request Catalog within standard HTML Web pages by using the FRAME html tag.

To do this, follow these steps:

1. Open this file with a text editor.

   C:\...\WEBINF\classes\applicationContext.properties
2. Set the value of the following parameter to true:

```javascript
src.enableFrame=true //enables embedding of SRC within standard HTML Web pages by using the FRAME html tag.
```

By default, the default value of the parameter is false, which means that the embedding Service Request Catalog in an HTML frame is disabled.

3. Create a Web page that contains a FRAME element, and then specify the source property of the FRAME tag as the url of your Service Request Catalog installation.

```html
<HTML>
<HEAD>
<TITLE>Service Request Catalog</TITLE>
</HEAD>
<FRAMESET rows="25,*" cols="*">
  <FRAME name="Top" src="corporateHeader.html" FRAMEborder=0 noresize scrolling=no FRAMESpacing=0 border=0>
  <FRAME name="Content"
  src="http://<domainname>:<port>/<srcDeploymentName>/secure/main.jsp"
  FRAMEborder=0 noresize scrolling=no FRAMESpacing=0 border=0>
</FRAMESET><noframes></noframes>
</HTML>
```

In this example, the `<domainname>` placeholder represents the domain on which you installed Service Request Catalog, the `<port>` placeholder represents the port number that was specified during installation of Service Request Catalog, and the `<srcDeploymentName>` represents the name you used to deploy Service Request Catalog. Therefore, a complete example for a deployed installation of Service Request Catalog may resemble the following:

http://myCompany:1701/src/secure/main.jsp

**Note:** The `corporateHeader.html` file that is referenced in the example is an example file that your organization can create to crown the frame that contains Service Request Catalog.

---

**Customize the landing pages**

You can specify how many information objects appear in the Dashboard, Services, and Support pages, and their order of appearance. The order is defined in rows that appear left-to-right, and top-to-bottom. For example, you might want to hide the object that lists popular bundles or popular items. There are seven information objects (widgets) that can appear on any landing page. The following list describes the name and configuration ID of these widgets:
To customize the order of appearance of these widgets, follow these steps:

1. **In the manifest.xml file, locate the <widgetsGroup></widgetsGroup> section.** Within this section, you can configure the number of widgets for each landing page and the display order (left to right and top to bottom).

   - `<widgetsGroup>
     + <widgets displayOn="DASHBOARD">
     + <widgets displayOn="SERVICE">
     + <widgets displayOn="SUPPORT">
   </widgetsGroup>

   There are sub-parameters for each available object.

2. **Locate the <widgets></widgets> parameter.** There are sub-parameters for each available object. For example, your manifest.xml may resemble the following:

   ```xml
   <widgets displayOn="pagename">
     <widget displayOrder="1"
       type="APPROVAL_STATUS" visible="true"/>
     <widget displayOrder="2"
       type="REQUEST_STATUS" visible="true"/>
     <widget displayOrder="3"
       type="POPULAR_ITEMS" visible="true"/>
     <widget displayOrder="4"
       type="ASSISTANCE" visible="true"/>
   </widgets>
   
   3. You cannot change the name of the object, but you can change the order that they appear in by changing the number assigned to the object. For example, to make the REQUEST_STATUS object...
appear first, change the value of its displayOrder attribute from 2 to 1.

- Make sure to change the order value assigned to APPROVAL_STATUS.
- Make sure each object has a unique displayOrder value.

4. To hide a widget, change its visible attribute from true to false.

5. Save your changes, but do not close the file until you are done.

Use the launcher to create third party links

The launcher is an empty space directly below the Dashboard, Services, Support, and Subscription menu bar. You can use the launcher to link to third party applications, such as video training links, an Internet search Web site, an internal HR Web site, or a chat support Web application. You can customize the presentation to show just an icon, or you can display a small amount of text with each icon.

To create a third party application launcher that appears on the Dashboard, Support, or Services landing pages, you will need to modify the following two files:

- manifest.xml
- Custom.properties

To create a third party link from the Dashboard, Services, or Support pages, follow these steps.

**Important note:** In the manifest.xml file, there are many xml elements that end with the term "key." Elements that end with the term "key" correspond to a property specified in the Custom.properties file. For example, the value of the <titleKey> element in the manifest.xml file is specified as "launcher_title" which corresponds to the "launcher_title" property in the Custom.properties file.

1. Open this file with a text editor.
   
   C:\...\src-9.40\resources\client\en\Custom.properties

2. Locate the template example in the file.

3. Follow the directions to copy the template section and paste it below the template. Customize only the pasted section. It is a good idea to add a comment line above the new section to identify its
4. Follow the guidelines in the value table to make sure this section contains values that you want to appear in the Service Request Catalog Dashboard.

- The identifier can be user-defined; however, when you edit the manifest.xml file, the identifiers have to match.
- The .value for each identifier is exactly what will display in the user interface.

In this example, we build a launcher for the XYZ third party business application.

```xml
<launcher launcherId="XYZ_Launcher" displayOrder="5" showLogoBox="true" showLogoDropShadow="true">
    <logo>XYZ_image.png</logo>
    <titleKey>launcher_title</titleKey>
    <descriptionKey>launcher_desc</descriptionKey>
    <launcherType>APP</launcherType>
    <fields>
        <field fieldId="text_label">
            <labelKey>text_label</labelKey>
            <type>TEXT</type>
        </field>
        <field fieldId="text_area">
            <labelKey>text_area</labelKey>
            <type>TEXTAREA</type>
        </field>
    </fields>
</launcher>
```

5. Save the file, but keep it open for reference as you configure the next file.

6. Open this file with a text editor.

C:\...\src-9.40\secure\configuration\manifest.xml

Use the following lines as a reference for the remaining steps:

```xml
<launcher launcherId="XYZ_Launcher" displayOrder="5" showLogoBox="true" showLogoDropShadow="true">
    <logo>XYZ_image.png</logo>
    <titleKey>launcher_title</titleKey>
    <descriptionKey>launcher_desc</descriptionKey>
    <launcherType>APP</launcherType>
    <fields>
        <field fieldId="text_label">
            <labelKey>text_label</labelKey>
            <type>TEXT</type>
        </field>
        <field fieldId="text_area">
            <labelKey>text_area</labelKey>
            <type>TEXTAREA</type>
        </field>
    </fields>
</launcher>
```
7. Locate the `<launchers>` section. The `<launchers>` element may contain one or more `<launcher>` sub-sections that describe the attributes of a third party application(s) that you want to display on the Dashboard. Replace the blue italicized text between each tag with a valid value. Follow the guidelines in the value table to make sure this section is syntactically correct and contains values that can be processed successfully.

8. Save the file, but keep it open for reference as you configure the next file.

9. If the application is running, log out and log in again to confirm that the third party application link appears in the dashboard.

   - For a detailed example of how to configure the launcher to start a support chat option, see "Example: Add a Bomgar live chat" on page 68
   - If you chose to link to use the Launcher to play a video, note that Service Request Catalog uses an open source video player class from Adobe. Therefore acceptable video formats are .flv, .f4v,
and mp4-based container formats. For more information regarding the video player, reference the following link:


Customize error messages

You can customize error messages that appear when you launch third party applications. These messages have default content but you may want to change them. To customize error messages, follow these steps:

**Tip:** It is always a good idea to make a backup of any system files before you make changes to them. If an error occurs later, you can always return the file to its original state.

1. If necessary, stop the server.

2. Open this file with a text editor:

   C:\...\src-9.40\resources\client\en\Custom.properties

3. Locate this section:

   # Customizable error message properties

4. Each message consists of a property ID that you should not change, and a property value that you can change.

   **Property ID Property value**

   - default_error_title = Error
   - default_success_title = Success
   - default_error_message = Message {0}.
   - default_success_form_message = Your information is posted.

   Edit one or all property values as necessary.

5. Save and then close the file.
Hide the service or support tabs

You can show or hide the Service, Support, and Generic Support widgets, including any references to those services in other areas of Service Request Catalog. To hide any of these tabs, set the value of the following parameters to false as appropriate. By default, these parameters are shown, which means that the default values are true:

src.sm.enableService=false //hides Service tab and hides the Services section from the Your Requests widget and Request Inbox.
src.sm.enableSupport=false //hides Support tab and hides the Support section from the Your Requests widget and Request Inbox
src.sm.canRequestGenericSupport=false //hides the generic support widget.

**Note:** You must still hide the Services and Support widgets by following the steps outlined in "Customize the landing pages" on page 12

Configure the Your Requests and Your Service Request

You can choose to show or hide the Pending, Closed, Denied, or Approved menu items for either the Your Requests widget or the Your Service Request pane. These settings are controlled by the following parameters in the applicationsContext.properties file. The default values for these parameters are true, which means the menu item will be shown. You can hide an item by setting the appropriate parameter to false.

**Your Service Requests**

src.ui.service.enablePendingInbox=true
src.ui.service.enableApprovedInbox=true
src.ui.service.enableClosedInbox=true
src.ui.service.enableDeniedInbox=true

**Your Requests**

src.ui.service.enablePendingOnWidget=true
src.ui.service.enableApprovedOnWidget=true
src.ui.service.enableClosedOnWidget=false
src.ui.service.enableDeniedOnWidget=false

**Note:** The Your Requests widget can show a maximum of three of the menu items. This is due to the height limitation of the widget.
Customize the application name

You can replace the application name, Service Portal, with the name of your company or a customized application name. For example, you would like the application name to reflect your company branding and your company name is XYZ. To do this, follow these steps:

1. Open this file with a text editor.
   
   `src\resources\client\en\Custom.properties`

2. Locate the // Company Display Name section.

3. Change the name of the default company name to your file name. For example, change:

   `default_company_name_label = Service Central`

   to

   `default_company_name_label = XYZ Company`

   You can change the ID on the left side to some other ID. For example, you could change this:

   `default_company_name_label = XYZ Company`

   to

   `my_company_name = XYZ Company`

4. Save and then close the file.

Edit manifest.xml

The manifest.xml file contains a section that identifies the key of the company name to be displayed.

1. Open this file with a text editor:

   `...\src\secure\configuration\manifest.xml`

2. Locate this section:

   `<companyDisplayNameKey>default_company_name_label</companyDisplayNameKey>`

3. If you changed the ID value in Custom.properties, make the corresponding change in this file.
Example: Changing default_company_name_label to my_company_name in the Custom.properties file. requires the new ID to appear in the manifest.xml file.

```xml
<companyDisplayNameKey>my_company_name</companyDisplayNameKey>
```

4. Save manifest.xml. The changes will be captured and propagated to all distributed servers the next time the server synchronization scripts run. To see them immediately, log out and log back into your local instance of Service Request Catalog.

Add a new language

Service Request Catalog is delivered in 16 language packages, which can meet most localization requirements. However, if you wish to add additional languages, you can do so on your own by using the following procedure. To do this, you must modify the following two areas:

- The resource files located in the following directory:
  ```
  [release war]\resources\client and [release war]\resources\server.
  ```
  All resources, messages, and user interface labels are stored in these properties files.

- Service Manager Server: The resources for configurable items, such as catalog items, user options, and checkout panels are stored here.

Configuration steps

In this example, we convert the current language of Service Request Catalog to Italian.

Add a new language into mlu.properties

1. Find resource under [release war]\resources\mlu.properties.

2. Add the language code for key "supportedLanguages":

   ```
   supportedLanguages=US,DE,IT,FR,ES,JP,BRAZILIAN,ZH,NL,KO,RU,CS,HU,PL,AR,HE
   ```

3. Add the following lines for the new language:

   ```
   IT.languageId=ITALIAN
   IT.localeLanguage=it
   ```
If you do not care about using the country code for Italian, just put IT as the language code, and all resource files must be appended as follow:

```
abc it.properties
```

If you do wish to put the country code (for example Switzerland (CH)) as the localeCountry for this language, then all resource properties files must be appended as follow:

```
abc it_CH.properties
```

Create a folder for Italian

If you do not care about using the country code for Italian, create a new folder under \release\war\resources\client\ named it as it.

If you do wish to put the country code, as the localeCountry for this language, create a new folder under \release\war\resources\client\ named it as it_CH.

Copy all client resource files

1. Copy all resource files from the \release\war\resources\client\en folder to the folder that you created in the previous step.

2. Append “_it” or ”_it_CH” to each client resource file, according to your decision in the previous step.

Copy all server resource files

1. Copy all server resource files from the \release\war\resources\server\en to the folder the you created previously.

2. Append “_it” or ”_it_CH” to each server resource file, according to your decision.

Translate each properties file

1. Translate the property files by hand.

Please heed those files that are marked by the “DO NOT TRANSLATE THIS FILE” statement; similarly, there are several lines in the files that are marked “DO NOT TRANSLATE.” If you are
not sure what lines should be translated or not, you can compare the existing files between a non-English version and the English version.

**Localize the resources in HP Service Manager**

Service Manager provides extensive language support. For details on how to localize Service Manager, refer to *SM_Open_L10N_UG.pdf*.

**Restart Tomcat**

1. Restart Tomcat, and then navigate to Service Request Catalog.

2. Set the Browser Language Preference in Service Request Catalog.

**Reference Materials**

- **SRC Service Manager customization reference** .......................................................... 141
- **Server customization reference** .................................................................................. 133
- **Client customization reference** .................................................................................. 103

**Configure session timeout**

In SRC, user sessions are closed when they have been idle for a specified period of time. By default, if users are inactive for 30 minutes, they are logged out from SRC automatically. You can configure the `session-timeout` parameter in the web.xml in Service Manager to control the session timeout interval:

```xml
<session-config>
  <session-timeout>30</session-timeout>
</session-config>
```

Two minutes before the session is closed, a pop-up message appears warning users that they will be logged out. Because the system calculates the session timeout interval based on mouse events, if users have a mouse operation during this time, the session timeout interval will restart with a new time recycle.

If users close the SRC browser or tabs without logging out, the session will be cleaned up in 15 to 30 seconds.
Service Manager configuration for Service Request Catalog

Some Service Request Catalog customization tasks require you to make changes to the HP Service Manager server to see the result in Service Request Catalog. Make sure you have a Service Manager client that connects to the same server that you use for Service Request Catalog.

Tailoring best practices

Making changes to a Service Request Catalog tailoring configuration can create problems. The best practice recommendation is to invest in a complete pre-production test cycle to be sure you have all the fields you need and that the names of these fields are meaningful to your user community. It is important to know that when these user fields are available and populated with information by users, if you make post-production changes to a tailoring configuration, there is an impact to existing requests.

<table>
<thead>
<tr>
<th>If I do this</th>
<th>This is the result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add new fields to an existing configuration.</td>
<td>Existing requests will not display those fields when you check their status.</td>
</tr>
</tbody>
</table>
| Remove any (or all) custom fields from an existing configuration and add new ones. | Existing requests will not display the data associated with the deleted fields. The new fields will not appear in those requests. The collected data associated with the deleted fields exists in the Service Manager database, but cannot display in the Service Request Catalog user interface.  
  **Best practice:** Do not remove a custom field once it is used in a production environment. |
| Add a new field that is mandatory.                 | Existing requests will not display this field as long as there is no attempt to edit and resubmit a request.  
  If the user resubmits the request, the new mandatory field appears and the user must supply any required information or form validation fails.  
  **Best practice:** Set a default value for new mandatory fields. |

Tailoring check-out panels

If you are using Service Request Catalog to submit a catalog or support request, you may want to customize Service Request Catalog to your specific business requirements. You can use a simple Service
Manager wizard to add new custom checkout panels, define custom fields and labels that appear on the Service Request Catalog checkout panels when you submit a request. You can also completely modify all out-of-box checkout panels in the Service Request Catalog user interface.

Checkout panels

In Service Request Catalog, users can submit different types of requests through various navigation paths. Each of these paths brings users to a specific request panel, which is called a checkout panel. The following are the default checkout panels in an out-of-box implementation of Service Request Catalog:

<table>
<thead>
<tr>
<th>Panel name in SRC user interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Catalog</td>
<td>This checkout panel is used to request for an item that is in the Service Catalog. To access this panel, click the Services tab on the Service Request Catalog home page, click Browse Catalog, and then browse to a service catalog item/bundle.</td>
</tr>
<tr>
<td>Support Catalog</td>
<td>This checkout panel is used to request for an item that is in the Service Catalog. To access this panel, click the Support tab on the Service Request Catalog home page, click Browse Catalog, and then browse to a Support Catalog item/bundle.</td>
</tr>
<tr>
<td>General Support</td>
<td>This checkout panel is used to request for an item or service that is not listed in the regular Service Catalog or Support Catalog. To access this panel, click the Support tab on the Service Request Catalog home page, click Create, and then create a custom Support Request.</td>
</tr>
</tbody>
</table>

Note:

You can add custom fields for the three checkout panels. However, for each company, you can create only one configuration for each checkout panel.

If you upgraded from an earlier version of Service Request Catalog by using the Migration Tool, you should note that only a limited migration is performed upon the checkout panel. Therefore, you should verify your checkout panel configuration after the migration, and adjustment as necessary. If you have also corresponding configurations for this company, these configurations are used. If there are no corresponding configurations for this company, the default configuration are used.
Configurations

A checkout panel is defined by a configuration in the Service Request Catalog Tailoring section of Service Manager. A configuration is associated with one or more sections, and each section can contain one or more custom fields that will be added to a Service Request Catalog checkout panel. You must specify a label for each section and field, which will display on the relevant checkout panel in the Service Request Catalog user interface.

Sections

A new section in the checkout panel can have a title and a combination of these objects:

- Fields that exist in a Service Manager table
- Labels that exist in a Service Manager form or a user-defined label
- Radio buttons
- Check boxes
- Lines of text for a user description text box
- Service Manager lookup fields
- Combo boxes
- Date fields
- Number fields
- Time fields*

*Time is usually specified by taking the system from the Service Manager server and adjusting for the client's location by using the GMT adjustment in the user's browser.

Custom fields

Custom fields are useful when you need to gather more information for a certain type of catalog request. Custom fields appear as a new accordion section in the checkout panel after the standard accordion sections for the request type. The data collected from the user with the custom fields is
stored in the appropriate Service Manager table that contains each field. New fields and labels appear in Service Request Catalog as a new section in the checkout panel. You can use this feature to collect additional information from the user that is relevant to the item or service fulfillment business logic.

Notes on creating custom fields

To create a custom field, you must specify the following:

- The checkout panel to which the field belongs
- The section of the checkout panel to which the field belongs
- A label for the field
- User restrictions on the field
- The field type
- Default values for the field, if any
- Mandatory requirements for the field, if any

You can specify the data type for a custom field and Service Manager will validate the user input against that data type and return an error message if there is a mis-match. In this case, the user can try again to provide data that is consistent with the assigned data type.

You must have the SysAdmin capability word in your user profile to access the Service Request Catalog tailoring wizard.

If a custom field was optional, and the user does provide a response or a value, the custom field does not appear in the status view.

Lookup tables and lookup fields

Some fields have a lookup table associated to them because of the business logic in Service Manager. For example, if you want to add a custom field named City to a checkout panel, users should not be allowed to type a city name in this field. Instead, users should select a value retrieved from the list of city field values in the location table. To achieve this, specify the location table as the lookup table, and the city field in this table as the lookup field for the custom field.

Note: To use a lookup table and a field, you must specify the Display Type of the custom field as Text.

You must also specify restrictions on the custom fields to limit the user's ability to change the content of the custom field.
Custom field restrictions

<table>
<thead>
<tr>
<th>Restriction</th>
<th>Restriction description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Only</td>
<td>Setting a custom field as read-only means that the user will not be able to modify this field. If no default value is specified, this field will not appear on the checkout panel.</td>
</tr>
<tr>
<td>Create Only</td>
<td>Specifying a custom field as create only means that the user can only add the field and its associated information when creating the request. This field will be disabled if the user is viewing their requests.</td>
</tr>
<tr>
<td>Create and Update</td>
<td>Create and Update means that user can add the field to their service request, and that this field can also be updated from the Service Request Catalog.</td>
</tr>
</tbody>
</table>

When the user checks the status of a request that contains a custom field, the field is read-only information until the user resubmits that request. At that point, the user can edit the custom field value and any other fields to be updated.

To make custom fields visible in the Service Request Catalog checkout panel, your Service Request Catalog applicationContext.properties file must point to the same Service Manager server that you used to create the custom fields.

Custom fields and the cache

It is possible that an Administrator adds new fields to a Service Catalog item at the same time that an user is attempting to order that item. If the Administrator implements a new custom field before the user submits a request, an inconsistency may be introduced to the system. To prevent this, user data is cached on the server side. Therefore, if an administrator modifies a panel or item while a user is actively using Service Request Catalog, the user’s data will be preserved and repopulated.

Note: We recommend that the Administrator avoids modifying custom fields while users are actively engaged in using the system.

Add a configuration

A configuration is the description of one or more new sections with custom fields that appears in a Service Request Catalog checkout panel. Use the examples to create a sample configuration of your own.

To complete any of the Service Request Catalog tailoring tasks, start with these basic steps.

1. Start a Service Manager Windows client session. Make sure the Service Manager Windows client connects to a Service Manager 9.40 server. The client can be an earlier version, but the server must be a 9.40 server.
2. Expand the left Navigation pane.

3. Click **Tailoring > SRC Tailoring**. Service Manager displays an SRC Configuration wizard. The first page of the wizard lists existing configurations. There can be a maximum of three configurations that add new sections with custom fields to the Support catalog, Services catalog, or Support checkout panel.

To add a configuration, follow these steps:

4. From the SRC Configuration Wizard home page, click **Add a Configuration**.

5. Create a **Name** for the new configuration.

   **Example**: Type **Support Custom Fields**.

6. Click the drop-down list to specify which checkout panel will display the new custom fields. Choose **Service Catalog**, **Support Catalog**, or **General Support**.

   **Example**: Select **Support Catalog**.

   **Note**: A General Support request is a request for an item or service that is not listed in the regular Service Catalog or Support Catalog.

7. If your HP Service Manager instance is in multi-company mode, choose a specific **Company** from the drop-down list.

8. Click **Next**.

9. You can choose an existing label for a new section, or you can create a new one. Click **Select a label** or **Add a label**.

   **Example**: Click **Add a label** and type **CONTACT NUMBERS** for the name of the new label.

10. You can choose an existing label for a new field, or you can create a new one. Click **Select a label** or **Add a label**.

   **Example**: Click **Add a label** and type **FAX** for the name of the new label.

11. Click **Next**.

12. Click the **Field Name** drop-down to choose the field that you want to populate with new
information. The values of the drop-down list can be changed in the “incidents” table from the Database Dictionary (DBdict).

**Example:** Choose Fax from the drop-down list.

**Note:** If you select a lookup table in step 16, then you must use the lookup field name here.

13. In the **Modifiability** field, you can set the field permission to enable the user to do one of the following:
   - **Create-only** changes the field only when submitting a new request. It cannot be updated if the request is resubmitted.
   - **Read-only** displays data that is never changed during the initial submission or during resubmission.
   - **Create and Update** changes the field when submitting a new request and updating a request if the request is resubmitted.

   **Example:** Select create-only from the drop-down list.

14. In the **Display Type** list, select the type of field that you want to appear in the Service Request Catalog checkout panel. Choose a display that is consistent with the type of data that you want to gather. For example, if you want the user to provide information, choose **Text**. If you want the user to choose from a pre-defined list, choose **Pick List**.

   **Example:** Choose the Text option.

15. Click **Next**.

16. In the **Lookup Table** field, select the table that contains the field to be populated with (or display) new information.

   **Example:** Leave this blank if you want to create a plain text field.

   **Note:** If you use a lookup table here, then you must use the lookup field name in step 12.

17. Select **Mandatory** if the user must provide this information. If the information is optional, skip this step.

   **Example:** Click the Mandatory checkbox.
18. Select **Default Value** if you want to specify a default value for the user.

    **Example:** Type None as the default value (in case the user has no fax number).

19. Click **Next**.

20. The Wizard displays the custom field that you created. You can click **Add New Custom Field** to add another.

    **Example:** Click **Add New Custom Field**.

21. Repeat step 10 through step 19 to add another custom field.

    **Example:** Create one more custom field for a Phone Number using the contacts table and Alternate Phones field.

22. When you have more than one custom field, you can:

   - Change the order of appearance by selecting a field and then clicking **Move Field Up** or **Move Field Down**.
   
   - Select a field and then click **Edit Custom Field** to make changes.
   
   - Click **Remove Custom Field** to delete that field.

23. When you are satisfied with the result, click **Finish**.

### Edit an existing configuration

You can edit an existing configuration for checkout panel, including the default out-of-box checkout panels. You can use the following steps to modify the default checkout panels to do any of the following:

- Enable or disable attachments.

- Change the name of the configuration as it appears in HP Service Manager.
- Change the name of the checkout panel as it appears in the Service Request Catalog user interface.
- Change the company for which the checkout panel applies.

To complete any of the Service Request Catalog tailoring tasks, start with these basic steps.

1. Start a Service Manager Windows client session. Make sure the Service Manager Windows client connects to a Service Manager 9.40 server. The client can be an earlier version, but the server must be a 9.40 server.

2. Expand the left Navigation pane.

3. Click **Tailoring > SRC Tailoring**. Service Manager displays an SRC Configuration wizard. The first page of the wizard lists existing configurations. There can be a maximum of three configurations that add new sections with custom fields to the Support catalog, Services catalog, or Support checkout panel.

Then, to edit the configuration, follow these steps:

1. Click **Edit Configuration**.

2. Edit the field you wish to change.

3. Click **Next**.

### Localize a configuration

To localize a configuration, follow these steps:

1. Log on to the Service Manager client as a system administrator.

2. Type `db` in the command box, type `svcSrcConfig` in the Table field, and then click **Search**.

3. Double-click `svcSrcConfig` in the search list, and then click **Search**.

4. Select the checkout panel configuration you want to localize.

5. In the details panel, find the **Label Id** under Options fields that you want to localize.

6. Type `message` in the command box, type `srcconfig` in the class field, 23 in the Message Number field, and then click **Search**.
7. Modify the text and language code that you want to localize. For example for the fr language code, enter the appropriate text in French, and then click Add.

Add a section to an existing configuration

To complete any of the Service Request Catalog tailoring tasks, start with these basic steps.

1. Start a Service Manager Windows client session. Make sure the Service Manager Windows client connects to a Service Manager 9.40 server. The client can be an earlier version, but the server must be a 9.40 server.

2. Expand the left Navigation pane.

3. Click Tailoring > SRC Tailoring. Service Manager displays an SRC Configuration wizard. The first page of the wizard lists existing configurations. There can be a maximum of three configurations that add new sections with custom fields to the Support catalog, Services catalog, or Support checkout panel.

To add a new section to a custom configuration, follow these steps

1. From the SRC Configuration Wizard home page, double-click an existing configuration.

   Example: If you added a section to the Support Catalog checkout panel, double-click the Support Catalog configuration.

2. To create a new section in that configuration, click Add a Section.

3. You can Select a Label from labels that you already created or you can click Add a Label to create a new section heading.

   Example: Type More Information as the new section heading.

4. Click Next.

5. Repeat step 10 through step 19 in "Add a configuration" on page 27 to add one or more custom fields to the new section.

6. When the new section is complete, click Finish.
Modify an existing section

You can make some editorial changes to an existing section.

- To change the name that identifies that section in the Checkout panel, select a label and then click *Edit Section Label*.
- To delete the section from the Checkout panel, select a label and then click *Remove Section*.
- To change the order of a section’s appearance, select a label and then click *Move Section Up* or *Move Section Down*.

Delete a configuration

To complete any of the Service Request Catalog tailoring tasks, start with these basic steps.

1. Start a Service Manager Windows client session. Make sure the Service Manager Windows client connects to a Service Manager 9.40 server. The client can be an earlier version, but the server must be a 9.40 server.
2. Expand the left Navigation pane.
3. Click *Tailoring > SRC Tailoring*. Service Manager displays an SRC Configuration wizard. The first page of the wizard lists existing configurations. There can be a maximum of three configurations that add new sections with custom fields to the Support catalog, Services catalog, or Support checkout panel.

To delete a custom configuration, follow these steps:

1. From the SRC Configuration Wizard home page, right-click an existing configuration.
2. Click *Remove Configuration*.

Multi-company mode

If you already operated in multi-company mode, your configuration of custom fields changes slightly. In this case, you can define a custom field configuration for each type of request (Service Catalog, Support Catalog, and Support request) for each company.

To verify whether multi-company mode is configured, follow these steps:
1. Log on to Service Manager as a system administrator.

2. Click System Administration > Base System Configuration > Miscellaneous > System Information Record.

3. On the General tab, make sure that Run in Multi-Company Mode is selected.

If you already configured multi-company mode, you also created the individual companies and associated those companies with users. You can verify the settings for any specific company by logging into Service Request Catalog as a user assigned to that company.

Example: Multi-company mode

The following steps provide an example of how you might configure multi-company mode for Service Request Catalog:

1. Create two new users in HP Service Manager. To do this, navigate to System Administration > Ongoing Maintenance > User Quick Add Utility.

2. Enter the required information to create a new user named kuser, click Next, and then follow the onscreen instructions to save the operator record. Repeat this step and create another user named duser.

3. Create two new companies. To do this, navigate to System Administration > Base System Maintenance > Companies.

4. Click Add and then enter the required information to create a new company named kcompany. Repeat this step and create another company named dcompany.

5. Assign the new users to the new companies. To do this, type operator in the command line, press the Execute button.

6. Search for kuser. In the Operator Record for kuser, select kcompany from the Default Company comfill control. Repeat this step for duser, but set the Default Company comfill control to dcompany.

7. In the manifest.xml file for your Service Request Catalog deployment, modify the configuration section as shown in the following sample:

```xml
<configurations>
  <configuration name="kcompany">
    ...
```
Customization Guide
Service Manager configuration for Service Request Catalog

8. To customize Service Request Catalog for each company, add the theme resources (for example, panel.swf and style.swf) to each company's folder.

Use the Service Manager software load balancer

Service Request Catalog supports the HP Service Manager software load balancer automatically, provided that you configure Service Request Catalog to point to the Service Manager software load balancer port. For example, if the http port for the Service Manager software load balancer is 13080, your Service Request Catalog applicationContext.properties file should resemble the following:

```properties
serviceManager.protocol=http
serviceManager.hostname=hostname.domainName
serviceManager.port=13080
```

If you are using SSL, make sure that Service Request Catalog points to the HTTPS port of the software load balancer:

```properties
serviceManager.protocol=https
serviceManager.hostname=hostname.SdomainName
serviceManager.port=13443
```
Miscellaneous tasks

The tasks in this section are optional tasks, which you may need to complete according to your organization requirements. You should read and familiarize yourself with these tasks in case you wish to return to perform additional configuration of Service Request Catalog at a later time.

Enable Knowledge Management

Before a user creates a support request, you may want to encourage them to search an external knowledge database to find a documented solution or recommendation. This feature is optional, but can be useful to help your user community resolve problems quickly without opening formal support requests.

Before you try to enable Knowledge Management from Service Request Catalog, verify the following:

- You have a Knowledge Management license. Without this license, SRC cannot display a Knowledge Management Search page.

- Knowledge Management is configured according to the recommendations in the Knowledge Management documentation.

When you enable the Knowledge Management link, the interface displays a Search Knowledgebase window where the user can type a search phrase.

To enable the Knowledge Management link, follow these steps:

1. Open this file with a text editor:

   C:\...\WEBINF\classes\applicationContext.properties

2. Locate and then set the src.km.enabled property to true.

3. Save and close the file.

4. Restart the Tomcat application server to make the changes take effect.

After enabling Knowledge Management, you must re-index the Service Request Catalog Knowledge Base index. To do this, follow these steps:
1. In the HP Service Manager client, navigate to Knowledge Management -> Administration -> Environment.

2. Check the SRC? box.

3. Verify that the correct server is specified in the Search Server Name: drop-down box. If not, select the correct server from this drop-down box.

4. Click the Full Re-index button.

**Configure subscriptions**

The configuration of subscriptions in Service Request Catalog is performed in the standard HP Service Manager Service Catalog. For documentation on how to set up Service Catalog items including subscriptions, consult your HP Service Manager documentation.

**Note:** It is not required that a user of Service Request Catalog must make or have a subscription to request an item.

**Enable the approval delegation**

To enable other operators to delegate approval authority, a system administrator must make the following changes to an operator’s application profile:

<table>
<thead>
<tr>
<th>To delegate approval authority for this application</th>
<th>Requires enabling this application profile option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Management</td>
<td>Can Delegate Approvals</td>
</tr>
<tr>
<td>Request Management</td>
<td>Delegate Approvals</td>
</tr>
<tr>
<td>Service Catalog (controlled by Service Desk profiles)</td>
<td>Delegate Approvals</td>
</tr>
</tbody>
</table>

The Delegate Approvals or Can Delegate Approvals profile option only controls whether an operator can view the Approval Delegation wizard. Separate application profile options control whether an operator can view or approve objects in an application.

**Enable delegation for service**
To enable Service Request Catalog users to delegate approval authority, a system administrator must update each operator’s service profile record with the Delegate Approvals and View options. There is no additional capability word required for a Service Request Catalog user to delegate their approval rights. To access the service profile settings from the operator record, follow these steps:

1. Click **System Administration > Ongoing Maintenance > Operators**.

2. On the **General > Application Profiles** tab, make sure you specify self service in the **Service Profile** field.

3. Click **Save**.

4. On the **Security** tab, locate the **Template Information** section.

5. If **Template_SelfService** appears in the **Template** field, delete it.

6. Click **Save**.

7. Return to the **General** tab.

8. Place the cursor in the **Service Profile** field.

9. Click **Find** on the toolbar. Service Manager displays the Service Desk Security Profile form.

10. On the **Security** tab, select **Delegate Approvals**. This enables the user to assign their approval rights to someone else.
11. If necessary, select View.

12. Click Save.

Enable Approval Delegation for Change

The following example illustrates how to enable approval delegation for the Change Management application:

1. Log in to Service Manager with a system administrator account.


3. In the Profile Name text box, type the name of a profile you want to grant approvals. For example, enter change approver.
4. Click **Search**.

5. Service Manager displays the current Change Management Security Profile.

6. In the Approvals/Groups tab, select **Can Delegate Approvals**.

7. Click **Save**.

**Note:** Best practice is to enable the Delegate Approvals or Can Delegate Approvals option only for operators who can both view and approve objects in the application.

### Enable Approval Delegation for Request (Quote)

The following example illustrates how to enable approval delegation for the Request (Quote) Management application:

1. Log in to Service Manager with a system administrator account.

2. Click **System Administration > Ongoing Maintenance > Profiles > Request Management Profiles**.

3. Service Manager displays your custom Request Management profile form.

4. In the **Profile Name** text box, type the name of a profile you want to grant approvals. For example, type `request approver`.

5. Click **Search**.


7. In the Alert/Approval tab, select **Can Delegate Approvals**, and then make sure that the **Approvals** check box is checked.

8. Click **Save**.

**Note:** Best practice is to enable the Delegate Approvals or Can Delegate Approvals option only for operators who can both view and approve objects in the application.

### Improve SRC performance

**Note:** These steps only improve performance only if you do not have an ApprovalA1 table in your
installation of HP Service Manager. In Service Manager 9.40, this table has already existed.

The following changes to the Service Manager database can improve the performance of Web Service requests from Service Request Catalog. These steps should be completed by an experienced Service Manager administrator.

These changes remedy problems that occur when the Service Manager server cannot fully translate Service Manager queries to an SQL server when fields that appear in the query are mapped to large object (LOB) type fields. The changes prevent inefficient scans by the Service Manager server when it runs queries against tables that reference LOB type fields.

**Note:** If you have already mapped any repeating group in a table that is referenced by an Array table, you have an Array table that uses the A1 alias. In this case, use a different alias, such as A2, and append that alias value to the base table name to form the table name. For example, you may already have an Array table for the Approval file. If you already have an A1 table, use alias A2 and name the table APPROVALA2 instead.

**Create the ApprovalA1 table**

To remap the current.pending.groups array from a CLOB/TEXT field to an Array table, create APPROVALA1 table. You should use the Dbdict utility, neither the System Definition utility nor the SQLMapping utility. To edit the dbdict for the Approval file, follow these steps:

1. In the **Fields** pane, scroll to and then double-click on the current.pending.groups array definition line.
2. Click the **Edit** field.
3. Set **SQLTable** to **a1**.
4. Click **Next**. You should be positioned on the dbdict entry for the current.pending.groups character field.
5. Set **SQLTable** to **a1**.
6. Change the **SQL Type** from **CLOB** to **VARCHAR(120)** and click **OK**.
7. Select the **SQL Tables** tab and add a new line with alias, name, and type of **a1 APPROVALA1 oracle10**.
8. Click **OK**. A pop-up dialog displays the DDL to create the new table.
Note: Do not modify the DDL.

9. Copy this DDL to the clipboard for future reference.

10. Click **User Alters**.

    Note: Do not click SM Alters.

11. Click **OK** on the main dbdict window to update the dbdict.

    Note: After you click **User Alters**, Service Manager displays a dialog warning that you must now alter the database by using the displayed DDL. However, Service Manager also tries to modify the database directly by using that DDL. If it succeeds, nothing further is required.

    The operation will succeed, provided the sqllogin account information in the Service Manager sm.ini file is for a database user with the rights to issue CREATE TABLE and CREATE INDEX commands. If the sqllogin account information in the Service Manager sm.ini file is for a database user who does not have these rights, the operation will fail, and an appropriately authorized user must manually run the DDL that you copied to the clipboard against the Service Manager database schema.

12. Verify that Service Manager succeeded in creating the new table and index by examining the Service Manager database in Oracle to confirm that the new APPROVALA1 table was created.

13. Investigate any SQL error messages that appear.

14. Use Oracle SQL Developer or another database management tool to verify that the APPROVALA1 table was created.

**Remap the parent.tree field**

The following task is to remap the parent.tree field in the capability file to an Array table. You should use the Dbdict utility, neither the System Definition utility nor the SQLMapping utility. To edit the capability file to an Array table, follow these steps:

1. Select the parent.tree type array.

2. Click the Edit field.

3. Set SQLTable to a1.
4. Click **Next**. You should be positioned on the **parent.tree** type character.

5. Change the **SQL** type to **VARCHAR(50)**.

6. Change **SQLTable** to **a1**.

7. Click **OK**.

8. Go to **SQLTables**.

9. Add **a1 CAPABILITYA1 oracle10**.

10. Click **OK**. A pop-up dialog displays the DDL to create the new table.

    **Note:** Do not modify the DDL

11. Copy this DDL to the clipboard for future reference.

12. Click **User Alters**.

    **Note:** Do not click SM Alters.

13. Click **OK** on the main dbdict window to update the dbdict.

    **Note:** After you click **User Alters**, Service Manager displays a dialog warning that you must now alter the database by using the displayed DDL. However, Service Manager also tries to modify the database directly by using that DDL. If it succeeds, nothing further is required.

    The operation will succeed, provided the sqllogin account information in the Service Manager sm.ini file is for a database user with the rights to issue `CREATE TABLE` and `CREATE INDEX` operations. If the sqllogin account information in the Service Manager sm.ini file is not for a database user who does not have these rights, the operation will fail, and an appropriately authorized user must manually run the DDL that you copied to the clipboard against the Service Manager database schema.

14. Verify that Service Manager succeeded in creating the new table and index by examining the Service Manager database in Oracle to see if the new **CAPABILITYA1** table was created.

15. Investigate any SQL error messages that appear.
16. Use Oracle SQLDeveloper or another database management tool to verify that the CAPABILITYA1 table was created.

**Remap the access.list Field**

The following task is to remap the access.list field in the svcCatalog file to an Array table. Use the Dbdict utility, not the System Definition utility or the SQLMapping utility. To edit the svcCatalog file to an Array table, follow these steps:

1. Select the access.list type array.
2. Click the edit field.
3. Set SQLTable to a2.
4. Click Next. You should be positioned on the access.list type character.
5. Change the SQL type to VARCHAR(50).
6. Change SQLTable to a2.
7. Click OK.
8. Go to SQLTables.
10. Click OK. A pop-up dialog displays the DDL to create the new table.

**Note:** Do not modify the DDL.

11. Copy this DDL to the clipboard for future reference.
12. Click User Alters.

**Note:** Do not click SM Alters.

13. Click OK on the main dbdict window to update the dbdict.

**Note:** After you click User Alters, Service Manager displays a dialog warning that you must now alter the database by using the displayed DDL. However, Service Manager also tries to modify the database directly by using that DDL. If it succeeds, nothing further is required.
The operation will succeed, provided the sqllogin account information in the Service Manager sm.ini file is for a database user with the rights to issue CREATE TABLE and CREATE INDEX operations. If the sqllogin account information in the Service Manager sm.ini file is not for a database user who does not have these rights, the operation will fail, and an appropriately authorized user must manually run the DDL that you copied to the clipboard against the Service Manager database schema.

**Remap the operators Field**

The next task is to remap the operators field in the kmgroup file to an Array table. Use the Dbdict utility, not the System Definition utility or the SQLMapping utility. To edit the kmgroup file to an Array table, follow these steps:

1. Select the operators type array.
2. Click the edit field.
3. Set SQLTable to a1.
4. Click Next. You should be positioned on the operators type character.
5. Change the SQL type to VARCHAR(60).
6. Change SQLTable to a1.
7. Click OK.
8. Go to SQLTables.
9. Add a1 KMGROUPA1 oracle10.
10. Click OK. A pop-up dialog displays the DDL to create the new table.

**Note:** Do not modify the DDL.

11. Copy this DDL to the clipboard for future reference.
12. Click User Alters.

**Note:** Do not click SM Alters.

13. Click OK on the main dbdict window to update the dbdict.
**Note:** After you click User Alters, Service Manager displays a dialog warning that you must now alter the database by using the displayed DDL. However, Service Manager also tries to modify the database directly by using that DDL. If it succeeds, nothing further is required.

The operation will succeed, provided the sqllogin account information in the Service Manager sm.ini file is for a database user with the rights to issue CREATE TABLE and CREATE INDEX operations. If the sqllogin account information in the Service Manager sm.ini file is not for a database user who does not have these rights, the operation will fail, and an appropriately authorized user must manually run the DDL that you copied to the clipboard against the Service Manager database schema.

14. Verify that Service Manager succeeded in creating the new table and index by examining the Service Manager database in Oracle to see if the new KMGROUPA1 table was created.

15. Investigte any SQL error messages that appear.

16. Use Oracle SQL Developer or another database management tool to verify that the KMGROUPA1 table was created.

**Remap the members Field**

Remap the members field in the cm3groups file to an Array table. To edit the cm3groups file to an Array table, follow these steps:

1. Select the members type array.

2. Click the editfield.

3. Set SQLTable to a1.

4. Click Next. You should be positioned on the members type character.

5. Change the SQL type to VARCHAR(60).

6. Change SQLTable to a1

7. Click OK.

8. Go to SQLTables.

9. Add a1 CM3GROUPSA1 oracle10.
10. Click **OK**. A pop-up dialog displays the DDL to create the new table.

   **Note:** Do not modify the DDL.

11. Copy this DDL to the clipboard for future reference.

12. Click **User Alters**.

   **Note:** Do not click SM Alters.

13. Click **OK** on the main dbdict window to update the dbdict.

   **Note:** After you click **User Alters**, Service Manager displays a dialog warning that you must now alter the database by using the displayed DDL. However, Service Manager also tries to modify the database directly by using that DDL. If it succeeds, nothing further is required.

The operation will succeed, provided the `sqllogin` account information in the Service Manager `sm.ini` file is for a database user with the rights to issue `CREATE TABLE` and `CREATE INDEX` operations. If the `sqllogin` account information in the Service Manager `sm.ini` file is not for a database user who does not have these rights, the operation will fail, and an appropriately authorized user must manually run the DDL that you copied to the clipboard against the Service Manager database schema.

14. Verify that Service Manager succeeded in creating the new table and index by examining the Service Manager database in Oracle to see if the new `CM3GROUPSA1` table was created.

15. Investigate any SQL error messages that appear.

16. Use Oracle SQL Developer or another database management tool to verify that the `CM3GROUPSA1` table was created.

### Additional performance tips

If the user receives a “max session exceed” error when running SRC with large number of concurrent users, increase the Service Manager max threads per node. To do this, follow these steps:
1. Edit the sm.ini in the root Service Manager directory: $SM_install_dir$\server\RUN

2. Locate the Threadsperprocess parameter, and then increase the number. If the parameter is not present in the sm.ini file, add the parameter by using the following format:

   Threadsperprocess:{number}

3. Restart the Service Manager server.

For Oracle databases, you can improve the SRC login and approval performance by adding an expression index. To do this, follow these steps:

1. Log in to Oracle Database under the sysop role.

2. Locate the APPROVALM1 table.

3. Create the following expression index on this table:

   FILE_NAME || '-' || COMPONENT || '-' || UNIQUE_KEY || '-' || NAME

Look-up fields

Service Request Catalog displays certain information, such as contact information, based on the corresponding QBE lists in Service Manager.

All Service Request Catalog users must have complete contact information specified in their HP Service Manager operator and contact records for the contact look-up to search successfully for a match. The contact record and the operator record must contain the first name, last name, and company information for the user. Service Manager generates a formatted contact ID that may also be used in the contact look-up.

Example: The Service Manager user Jennifer Falcon has this information in her company and operator records:

- Last name = Falcon
- First name = Jennifer
- Company = HP

Her Service Manager generated contact ID is FALCON, JENNIFER.

If you are using the Contact look-up in Service Request Catalog, you can find her by typing one of the following:
- Her first name only
- Her last name only
- Both first and last name
- Her contact ID (in upper case)

**Add additional contact information**

You can add additional contact information by modifying the following property in the applicationContext.properties file:

```
src.sm.additionalContactInfo=N,F,L,E
```

The N, F, L, and E represent the following:

- N - : FullName
- F - : FirstName
- L - : LastName
- E - : E-mail address

If you do not wish to use any additional info, you can comment out this line by using the preceding pound (#) character.

**Define a Date/Time user selection**

This version of Service Request Catalog allows you to use the Date/Time Validation mechanism of Service Manager. You can use the `tod()` to define an appropriate expression for a Date/Time user selection.

For example, suppose you want to specify a minimum 5-day delivery date for the delivery of a catalog item. To do this, you can configure a "Deliver By:" user selection in the Catalog Item to use the following expression (in the "greater than" field):

```
tod()+ '5 00:00:00'
```

When a user tries to order this catalog item, the user must choose a date that is greater than 5 days from the current system time. Otherwise, a warning message appears, and the user must pick another date.
Update Service Manager lists

HP Service Manager provides list information to Service Request Catalog that appears in different sections as the user completes a request. The items in the lists are configured in Service Manager. Therefore, if you want to change the contents of certain lists that appear in Service Request Catalog, you must use the Service Manager Windows or Web client to change the appropriate Service Manager record.

The Service Manager Help Server has all the information and steps to make changes to global lists.

1. Obtain the URL for the Service Manager Help Server.
2. Type or paste it into a browser window.
3. Type global lists in the Search area of the Help Server interface to locate all information about these lists.

For Service Request Catalog, you can make changes to these lists:

- Callback Type (to specify the users preferred contact method)
- Interactions Activity Types (such as “Operator update”)
- Resolution Code (to categorize the reason for closure)
- Urgency (such as “Critical” or “High”)

Service Manager localizes these lists to make sure they render correctly in different locales. If you customize any list value, you must create manual translations of your new values and store them in the appropriate locale folder.

Supporting other locales

You may have a global community of users who want to view the service catalog in their preferred language. Service Request Catalog provides complete out-of-box localization for multiple languages. You must complete tasks to configure support for your selected languages.

Service Request Catalog also supports the currency code and symbol for each of these languages. Complete the following tasks to enable more than one locale. The only constraint is that each locale
must be supported by HP Service Manager. Complete the first three tasks with a Service Manager client that connects to your Service Manager server.

**Review Out-of-box Supported Languages**

<table>
<thead>
<tr>
<th>Language</th>
<th>Language Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arabic</td>
<td>ar_SA</td>
</tr>
<tr>
<td>Brazilian Portuguese</td>
<td>pt_BR</td>
</tr>
<tr>
<td>Czech</td>
<td>cs_CZ</td>
</tr>
<tr>
<td>Dutch</td>
<td>nl_NL</td>
</tr>
<tr>
<td>English</td>
<td>en_US</td>
</tr>
<tr>
<td>French</td>
<td>fr_FR</td>
</tr>
<tr>
<td>German</td>
<td>de_DE</td>
</tr>
<tr>
<td>Hungarian</td>
<td>hu_HU</td>
</tr>
<tr>
<td>Italian</td>
<td>it_IT</td>
</tr>
<tr>
<td>Japanese</td>
<td>ja_JP</td>
</tr>
<tr>
<td>Korean</td>
<td>ko_KR</td>
</tr>
<tr>
<td>Polish</td>
<td>pl_PL</td>
</tr>
<tr>
<td>Russian</td>
<td>ru_RU</td>
</tr>
<tr>
<td>Simplified Chinese</td>
<td>zh_CN</td>
</tr>
<tr>
<td>Spanish</td>
<td>es_ES</td>
</tr>
</tbody>
</table>

**Task 1: Configure the language in Service Manager**

The first task is to make the selected language available when the user logs in to Service Request Catalog. Some languages may already be enabled on the Service Manager side. Others may not be enabled. To verify whether your target language is enabled, or to enable your target language, follow these steps:

1. From the Service Manager client, expand the left navigation and click **Tailoring > Database Manager**.

2. From the **Table** field, type or select **language**.

3. Click the **Search** icon.
Service Manager returns a list of Service Manager language files.

4. Double-click the desired language file.

5. From the Language Identification dialog, click the Search icon.

   A list of available languages appears. If the language you want has an active flag set to true, the language is enabled, and your task is complete. If the flag is false, proceed to the next step.

6. Select the language that you want to enable.

7. Check the empty Active for logins check box.

8. Click Save to update the language record.

9. Click OK.

10. Repeat step 1 through step 9 to add more languages.

11. Click Back to exit this task.

12. Restart the Service Manager client to verify your changes.

Task 2: Configure the user profile

The second task is to associate a default language with the individual user in their Service Manager user profile. For users in a multi-lingual environment, one might prefer to search the catalog in French, while another user would be more comfortable with Spanish.

Note: If the user does not exist in the Service Manager system, you can follow these steps to navigate to the Operator Record dialog, but you must create the user before you complete the language association task.

1. Verify the target language is configured in Service Manager by reviewing Task 1 in the Support Other Locales section.

2. In the System Navigator pane, click System Administration > Ongoing Maintenance > Operators to edit the user’s operator record.

3. Click Search.

4. Select the user from the Login Name or Full Name column.
5. Click the **Login Profile** tab.

6. Click the **Language** drop-down list to select the default language for that user.

7. Click **Save**.

8. Click **OK**.

9. Repeat step 2 through step 8 if you have more than one operator record to configure.

10. Click **Back** to exit this task.

**Task 3: Localize the item record**

1. To display any item in the user’s preferred language, you must add the specific language attribute to the item record. If you skip this step, the item appears in the default language.

2. In the **System Navigator** pane, click **Service Catalog > Manage Catalog** to locate the record to be localized.

3. Click **Search**. Service Manager displays a list of the item records in the catalog.

4. Scroll to find the target record in the list.

5. Click the record in the list to display all the record information in the **Category Definition** form.

6. When Service Manager displays the record, right-click anywhere in the form and select **Localize this record** from the context menu.

7. Service Manager displays the **Create Localized Data** wizard.

8. Choose the language from the drop-down list. **Example**: select German.

9. Replace the English **Name**, **Short Description**, and **Long Description** with equivalent values in the language you selected in step 6.

10. Click **Finish**. Service Manager displays a confirmation message above the **Category Definition** form.

11. Repeat step 3 through step 8 to create localized versions of each catalog record.

12. When you are finished, click **OK**.

13. Click **Back** to exit the task.
Task 4: Configure the currency

All currencies related to supported locales are available in the Service Request Catalog application. Currency values can appear with a symbol, such as the dollar symbol ($), or a code, such as USD (United States Dollars). The default display for the English locale is the currency symbol. The default display for non-English locales is the currency code.

Change a currency code or symbol

You can change the way currency displays, or you can add a new code or symbol.

Example: You would like to display using a different currency code or symbol, or even add a new currency.

Change the default currency symbol for Service Request Catalog

The Service Manager server normally provides localized item descriptions and converts the item cost to your locale currency symbol and value without intervention. You can override the currency display by overriding the default value.

1. On the server where you deploy the Service Request Catalog, navigate to the Control Panel > Administrative Tools > Services dialog.

2. Navigate to this folder:
   
   Windows: C:\...\src-9.40\resources\client\ 
   Unix: /opt/.../src-9.40/resources/client/
   
   All localization sub-folders contain translated text for the application.

3. Open the folder for your locale.

   Example: Hungarian translations are in this folder:
   
   Windows: C:\...\src-9.40\resources\client\hu\ 
   Unix: /opt/.../src-9.40/resources/client/hu/

4. Open the CurrencySymbol_hu.properties file.

5. Locate the currency code and symbol that you want to display and remove the comment character from that entry.

6. Save and close the file.

Add a new currency to Service Manager
If you want to display items in the new locale by using the correct currency value and symbol, browse the Service Manager documentation to locate “Displaying currency in the Service Catalog.” Follow the recommendations in this and related topics to make sure that:

- Display currency field is set correctly in the affected operator records.
- Currency table contains the correct currency definition record.
- Currency conversion rate, established by the daily currency exchange rate, appears in the currency conversion table.

**Add a New Code and Symbol to Service Request Catalog**

In this example, we want to add the Korean code and symbol.

Follow these steps on the server where you deploy the Service Request Catalog .war file.

1. On the server where you deploy the Service Request Catalog, navigate to the **Control Panel > Administrative Tools > Services** dialog.

2. Navigate to the following folder:
   - Windows: C:\...\src-9.40\resources\client\ko/CurrencySymbol_ko.properties
   - Unix: /opt/.../src-9.40/resources/client/ko/CurrencySymbol_ko.properties
   All localization sub-folders contain translated text for the application.

3. The /ko folder contains all property files for the Korean locale. Open this file with a text editor:
   - Windows: C:\...\src-9.40\resources\client\ko\CurrencySymbol_ko.properties
   - Unix:/opt/.../src-9.40/resources/client/ko/CurrencySymbol_ko.properties

4. Add the correct currency code and currency symbol to this file.

   **Example:** To add the Korean currency, add this value:
   ```
   KRW=₩
   ```

5. Save and close the file.

**Task 5: Display the currency for a user**

To activate the target currency, follow these steps:
1. In the Service Manager Windows client, navigate to **System Administration -> Base system Configuration -> Currencies**.

2. Enter the target currency name (for example, CNY) and then click **Search**.

3. Check the Active? checkbox and click then click **OK**.

To add conversion rates, follow these steps:

1. Navigate to **System Administration -> Base system Configuration -> Conversion Rates**.

2. Search for the target currency Conversion Rates. Add one if it does not exist. Generally, the **Root Currency Code** should be **USD**.

To add the currency to the user profile, follow these steps:

1. Log in to Service Manager and then enter operator in command line.

2. Input a valid Service Request Catalog username and then click **Search** to open the user profile.

3. Click the **Login Profiles** tab and change the **Display Currency** value to the target one.

4. Click **Ok**.

5. Log in to Service Request Catalog by using the username from step 2 and set the language to one which is consistent with the currency. For example, to test Chinese Yuan, set the language to **Chinese**.

6. Open the catalog item that has a specified cost in Service Manager and verify that the correct currency symbols, converted price, and currency format are correct.

**Task 6: Configure the browser for a locale**

To view Service Request Catalog in a preferred language, users should configure their browser settings. The following general guidelines are for Microsoft Internet Explorer and Mozilla Firefox. If you use an unsupported browser, consult the browser documentation.

**Microsoft Internet Explorer**

1. Click **Tools > Internet Options**.

2. Click the **General** tab.

3. Click **Languages**.
4. Click Add.

5. Choose a supported language and then click OK.

6. Select that language and move it to the top of the preference list.

7. Click OK.

Mozilla Firefox

1. Click Tools > Options.

2. Click Content.

3. In the Languages section, click Choose.

4. Click Select a language to add.

5. Choose a supported language and then click Add.

6. Select that language and move it to the top of the preference list.

7. Click OK.

Authorizing bi-directional text

Starting from HP Service Manager 9.32, Service Request Catalog supports right-to-left direction for both Arabic and Hebrew language environments. However, without certain modifications, you may find that some text is displayed left-to-right (LTR) or right-to-left (RTL). The following sections provide a brief introduction on bidirectional text, and show you how to input bidirectional texts in the Service Manager client.

Understanding an unexpected text order

HP Service Manager stores texts in Unicode encoding. Service Request Catalog displays texts by using the Unicode standard, which defines the order in which text should be displayed by specifying a particular base direction.

The base direction influences the order of the display of text of different directions, and the display of directionally-neutral text (i.e., characters or sequences of characters that do not have inherent directionality, as defined in the Unicode Character Standard).
In Service Request Catalog, if the language is Arabic or Hebrew, the base direction is RTL; otherwise, LTR.

For example, suppose a user writes the following text in English as the name of a catalog item. In an LTR language in Service Request Catalog, the text is displayed exactly as shown:

Printing (North America)

However, in a Hebrew or Arabic language environment, where the base direction is RTL, the text displays as follow:

(Printing (North America

This behavior occurs because a parenthesis is directionally neutral in the Unicode standard and does not have an inherent direction. The Unicode text engine first checks whether a parenthesis can inherit a direction the from surrounding text. If a parenthesis cannot inherit a direction from the surrounding text, it defaults to the base direction. In the example text, the first parenthesis is embedded in a single run of LTR text. Therefore, it adopts the direction of its surrounding text. The second parenthesis, which is at the end of the text, is not surrounded by text of a certain direction. Therefore, the second parenthesis defaults to the RTL base direction.

Thus, the text can be divided into two parts with different directions, wherein the first part is LTR:

Printing (North America

And, the second part is RTL:

(  

Because “)” displays as “(“ in the RTL direction, and the two divided text parts are ordered in the RTL base direction, the text is displayed as shown:

(Printing (North America

To resolve this issue, the Unicode standard contains two special characters that are used in bi-direction-enabled text.

<table>
<thead>
<tr>
<th>Unicode character</th>
<th>Code</th>
<th>HTML</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRM</td>
<td>U+200E</td>
<td>‎</td>
<td>LEFT-TO-RIGHT MARK</td>
<td>Left-to-right zero-width character</td>
</tr>
<tr>
<td></td>
<td></td>
<td>‎</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;lrm;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RLM</td>
<td>U+200F</td>
<td>‏</td>
<td>RIGHT-TO-LEFT MARK</td>
<td>Right-to-left zero-width character</td>
</tr>
<tr>
<td></td>
<td></td>
<td>‏</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;rlm;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
These zero-width invisible characters act as surrounding text. These characters enable directionally-neutral characters (such as parentheses) to inherit the correct direction from the surrounding text instead of defaulting to the base direction.

In the prior example, you can add a left-to-right mark after the second parenthesis to fix the problem as shown in the following example:

Printing (North America) LRM

Because both parentheses are surrounded by LTR text, the whole text is treated as a single run of LTR text. Therefore, the text is displayed as we want in Hebrew or Arabic environment:

Printing (North America)

There are other control characters related to bi-directional text in the Unicode standard which are rarely used. For full instruction on the Unicode Bidirectional Algorithm and the related control characters, see the following links:

http://www.w3.org/International/articles/inline-bidi-markup/uba-basics
http://www.unicode.org/reports/tr9/

Input bidirectional text in the Service Manager client

The text input boxes in the Service Manager clients use Unicode encoding. However, unlike other third-party text editing tools where you can easily insert a left-to-right mark or other Unicode control characters by using a keyboard shortcut or context menu, the Service Manager clients are somewhat limited.

You can use any of the following three methods to input bidirectional text in the Service Manager clients.

**Copy from text editing tools (Recommended)**

If you are familiar with authoring bidirectional text, you probably have a text editing tool that you often use. You can copy previously-authored bidirectional text from your text editing tool and paste directly into the text input box in the Service Manager client.

**Note:** Make sure that you do not have the invisible Unicode characters, especially at the start or the end of the text.
Insert a single unicode control character

If you already have some text in Service Manager, and you just want to insert several LRM or RLM marks to make that text bidirectional, you can create these marks easily by using WordPad in Windows. To do this, use one of the following methods:

**Method 1**

1. Open a new document in WordPad.
2. Type `200E` (200E is the hexadecimal code of LRM) and then press the Alt+X keyboard command. The “200E” should disappear and the LRM is inserted.
3. Press **CTRL+A** to select all, and then press **CTRL+C** so you can paste it back into Service Manager.

**Method 2**

1. Open a new document in Wordpad.
2. Press and hold the **ALT** key, and then type `8206` (the decimal code of LRM) by using the numeric keypad. When you release the ALT key, the LRM is inserted.
3. Press **CTRL+A** to select all, and then press **CTRL+C** so you can paste it back into Service Manager.

**Rich text input box in Service Manager client**

There are a few rich text input boxes in Service Manager client, such as the detailed description of Catalog items. Rich text input boxes have two editing modes: “normal” and “source”.

- In the “normal” mode, you can input bidirectional text as you would in other text input boxes.
- In the “source” mode, you must use the HTML escaped form of special Unicode characters. For example, to input a RLM (right-to-left mark), you should input one of the following:

  ```
  &\#8207;
  &\#x200F;
  &rlm;
  ```

**Configure additional details for Service Requests**

You can configure Service and Support request to display additional details from HP Service Manager.
Enable journal updates

You can enable journal updates to appear in Service Request Catalog for each request to provide additional information to the user on the history of their request. Journaling is a feature of Service Manager that allows additional details or updates to a record. If Journaling is enabled on your Service Manager server, and you want to enable those journal updates for Service Request Catalog, follow these steps:

1. Open the `applicationContext.properties` file, and specify the following properties as **true**:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>src.sm.service.journalUpdatesEnabled=true</code></td>
<td>A value of true enables the Journal Updates for service. Change the value to false to hide Journal Updates information</td>
</tr>
<tr>
<td><code>src.sm.support.journalUpdatesEnabled=true</code></td>
<td>A value of true enables the Journal Updates for support. Change the value to false to hide Journal Updates information.</td>
</tr>
</tbody>
</table>

Edit the custom definition label for journal updates

By default, the label for journal updates is "Request Updates." However, you can modify this label. To do this, edit the `Custom.properties` file, and specify a new label for the `request.update` property.

Require users to read catalog item details

In Service Request Catalog the **Request Now** button allows a user who knows what they are ordering to quickly place a catalog item in the shopping cart and proceed to check out. However, in some cases, you may wish to force the user to read the details of the catalog item. To do this, you can set the `enableDirectRequest` parameter for each of the Service and Support sections to false.

```
src.ui.support.enableDirectRequest = false
src.ui.service.enableDirectRequest = false
```

**Note:** A value of false means that users must read the catalog item description. A value of true allows users to use the **Request Now** button to place and item in their shopping cart.
Enable the line item status

You can use the following parameter to enable the display of the status for individual line items of a request. If this option is enabled, users can see the status of individual components of a request when they view the status of their request.

src.ui.service.enableLineItemStatus=true

Configure IPv6

To configure IPv6 for Service Request Catalog, follow these steps:

1. Enable IPv6 support in Tomcat. To do this, see the Tomcat section of the "Configure the Web clients for IPv6" page in the HP Service Manager Help Center.

2. Enable IPv6 in Service Manager. To do this, see the "Configure the Service Manager server for IPv6" page in the Service Manager Help Center.

3. In folder where Service Request Catalog is deployed, set the sm.hostname parameter in the applicationContext.properties file to a full qualified domain name or to a valid IPv6 address:

   sm.hostname=[2620:0:a17:e006:4a8:91c1:c9cb:93dd]

4. You can access the Service Request Catalog application by entering the valid IP address in your browser, as shown in the following example:

   http://[2620:0:a17:e006:4a8:91c1:c9cb:93dd]:8080/src9.40

Enable FIPS for Service Request Catalog

FIPS (Federal Information Processing Standards) are a set of standards that describe document processing, encryption algorithms and other information technology standards for use within U.S. non-military government agencies and by U.S. government contractors and vendors who work with such agencies.

As of version 9.32, HP Service Manager supports the FIPS 140-2 approved Advanced Encryption Standard (AES) 256-bit data encryption algorithm. To use FIPS together with Service Request Catalog, you must first enable FIPS mode in Service Manager. For information on how to do this, see the following sections of the Service Manager Help Center.
Use the FIPS standard in Service Request Catalog

Service Request Catalog provides one tool to generate encrypted password. To support the FIPS standard encryption, this tool has been upgraded from Service Request Catalog 9.32.

For the details of this tool, see the “Create an Encrypted Password for the Service Manager Administrator” in the SRC 9.40 Interactive Installation Guide.

By default, the runme.bat script generates an encryption key that meets the FIPS standard. More specifically, the runme.bat script uses an Advanced Encryption Standard (AES) 256-bit data encryption algorithm. However, to remain compatible with previous versions of Service Request Catalog, an optional parameter (PBE) is now accepted by the runme.bat script, which uses the MD5 and DES encryption of earlier versions of Service Request Catalog. Therefore, the following commands are valid:

- Runme.bat //uses the AES 256-bit algorithm
- Runme.bat PBE //uses the earlier MD5 and DES algorithm

Common Access Card introduction

The Common Access Card (CAC) is the United States Department of Defense (DoD) smart card for multi-factor authentication. CACs are issued as standard identification for active-duty military personnel, reserve personnel, civilian employees, non-DoD government employees, state employees of the National Guard and eligible contractor personnel. In addition to its use as an ID card, a CAC is required for access to government buildings and computer networks.

If you have not configured CACs for Service Manager, you must do so before you can enable CAC for Service Request Catalog. To do this, refer to the following sections of the Service Manager Help Center:

- Common Access Card (CAC) sign-on
- Requirements for CAC sign-on

You may also want to review the following example in the Service Manager Help Center:
Example: Enabling CAC sign-on

If you have already configured CACs for Service Manager, you can skip ahead to the following section:

"Enable SRC for Common Access Cards" below

Enable SRC for Common Access Cards

To enable Common Access Cards for Service Request Catalog, follow these steps:

1. Enable CAC in HP Service Manager. To do this, use the cacsign on parameter from a Service Manager command line, or specify it in the sm.ini file.

   Command line: sm -httpPort:13080 -cacsignon:1
   Initialization file: cacsignon:1

   For more information on this parameter, see the following section of the Service Manager Help Center:

   - "Parameter: cacsignon"

2. Complete the following tasks from the "Example: enabling CAC sign-on" section of the Service Manager Help Server.

   Task 1. Enable required SSL encryption and trusted clients.
   Task 2. Enable CAC sign-on in the SM server.
   Task 5. Connect your web application server to the web server.
   Task 6. Configure your web application server to use SSL.
   Task 7. Configure the web server to use SSL.
   Task 8. Create operator records for CAC users
   Task 9. Configure web browsers to enable web client users to use CAC.
   Task 10. Test your CAC sign-on setup.

Note:

- When you perform "Task 1. Enable required SSL encryption and trusted clients", you are referred to "Example: Enabling required SSL encryption and trusted clients" section of the Service Manager Help Center. You need to copy the keystore files generated at step 9 of this procedure to the WEB-INF folder in your Service Request Catalog deployment or, if needed, to a folder that you plan to specify in the applicationContext.properties file.
- Tasks 3 and 4, which are missing from the preceding list, refer to the configuration of the Service Manager Web tier and Windows client, and are not a consideration for Service Request Catalog.

- If you choose to use CAC, make sure that you have configured Service Request Catalog to use SSL and either TSO or LW-SSO (as appropriate for your environment) as described in the Service Request Catalog Interactive Installation Guide.

The following tables show a summary of how your property files should be configured after you have enabled CAC:

**ApplicationContext.properties**

<table>
<thead>
<tr>
<th>Configure these properties</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>sm.protocol=sm.hostname=sm.port=13443 src.trustStore=src.trustStorePassword=src.keyStore=src.keyStorePassword=</td>
<td>Enables two-way SSL between SRC and Service Manager.</td>
</tr>
<tr>
<td>Note: The keys are generated when you complete &quot;Task 1. Enable required SSL encryption and trusted clients.&quot; See the previous note for details.</td>
<td></td>
</tr>
<tr>
<td>sm.protocol=https sm.hostname=[FQDN] sm.port=13443 src.trustStore=C:\client\cacerts src.trustStorePassword=changeit src.keyStore=C:\client\src.keystore src.keyStorePassword=clientkeystore</td>
<td></td>
</tr>
<tr>
<td>src.security.mode=cac src.security.cac=true</td>
<td>Enables CAC.</td>
</tr>
<tr>
<td>src.security.secureLogin=true src.security.sslPort=8443</td>
<td>Enables SSL Connection between Service Request Catalog and the end user's client browser.</td>
</tr>
</tbody>
</table>

**cacConfiguration.properties**

<table>
<thead>
<tr>
<th>Configure these properties</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>certificateFieldExtractDN=Subject.CN</td>
<td>This parameter is used to get the field in the certificate that stores the login name. If the DN is from one of the fields in the Subject, set this parameter using the following format: Subject.CN. Subject.CN means extracting the DN from the Common Name field in Subject.</td>
</tr>
<tr>
<td>validationStrategy=1</td>
<td>Client Certificate Verification Sequence is as follows:</td>
</tr>
<tr>
<td></td>
<td>1. Check the revocation status by using local CRL.</td>
</tr>
<tr>
<td></td>
<td>2. Check revocation status by using an online CRL.</td>
</tr>
<tr>
<td></td>
<td>3. Check revocation status by using OCSP.</td>
</tr>
<tr>
<td></td>
<td>4. Check if the status is expired.</td>
</tr>
<tr>
<td></td>
<td>5. Check if the certificate type is smart card.</td>
</tr>
<tr>
<td>RootCertPath=</td>
<td>Specifies file path to the issuer CA certificate. This is a mandatory property.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>RootCertPath=C:/client/CAC/hp_ca_for_cac.cer</td>
</tr>
<tr>
<td>CRLRefreshScheduler=</td>
<td>Specifies a schedule in minutes for CRL refresh, if CRL check is enabled.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>CRLRefreshScheduler=1440</td>
</tr>
<tr>
<td>CRLDownloadURL=</td>
<td>The URL that identifies the location of the CRL. If it is not specified, the CRL Distribution Point from the certificate being validated will be used. This behavior will slow performance dramatically.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>CRLDownloadURL=<a href="http://onsitecrl.example.com/LatestCRL.crl">http://onsitecrl.example.com/LatestCRL.crl</a></td>
</tr>
<tr>
<td>CRLStoreLocation=</td>
<td>Specifies the CRL file path.</td>
</tr>
</tbody>
</table>
**Auto-suggest/Type ahead**

Service Request Catalog supports an auto-suggest/type-ahead feature to suggest the top ten matching Catalog items when users enter information in the Search Box. You can enable or disable this feature by modifying the `src.ui.enableAutoComplete` property to true or false in the `applicationsContext` file, as shown in the following example:

```
src.ui.enableAutoComplete=true
```

*Note: A value of false disables the type ahead feature.*

**Enable the Remember Login User Name feature**

You can enable the Remember Login User Name feature in Service Request Catalog. The `src.enableRememberLoginUserName` parameter is introduced to enable or disable the Remember Login User Name feature in the user login panel. This parameter is set to false by default, which means SRC does not remember the user name when the user logs on to SRC.

```
src.enableRememberLoginUserName=false
```

**Disable the Delivery Targets drop-down list**

You can hide the Delivery Targets drop-down list in Service Request Catalog. The `src.hideDeliverySLA` parameter is introduced in the `applicationContext.properties` file to hide or show the Delivery Targets drop-down list globally in SRC. By default, the parameter is set to false, which means the Delivery Targets drop-down list is displayed.

```
src.hideDeliverySLA=false
```
Appendix A: Launcher examples

The following pages demonstrate various ways you can use the Launcher. All configuration for the Launcher is performed by editing the manifest.xml file. Some examples are included in the manifest.xml file, but are commented out by default.

Example: Add a Bomgar live chat

Service Request Catalog supports integration only with a Bomgar™ chat solution. As such, you must independently purchase a Bomgar™ chat solution for your organization. For more information about the Bomgar API and API documentation, visit www.bomgar.com.

A live chat interface can enhance your catalog support strategy. If a user has a question about an item or a request, live chat support provides immediate assistance. You can embed a live chat link at the top of the Dashboard, Services, or Support landing pages. The link launches a chat window in a new tab or window, depending on the user’s browser settings. To create a live chat link from the Dashboard, Services, or Support pages, follow these steps.

**Tip:** It is always a good idea to make a backup of any system files before you make changes to them. If an error occurs later, you can always return the file to its original state.

1. Open this file with a text editor.
   
   C:\...\src-9.40\resources\client\en\Custom.properties

2. Locate the chat example in the file.

3. If necessary, change chat_launcher_title, and the chat_launcher_description display text.

   Note that:

   - The identifier can be user-defined; however, when you edit the manifest.xml file, the identifiers have to match.
   - The value for each identifier is exactly what will display in the user interface.

4. Save and the file but keep it open for reference as you configure the next file.

5. Open this file with a text editor.
Use the following lines as a reference for the remaining steps.

```xml
<launcher
  launcherId="Chat_Launcher"
  showLogoBox="true"
  showLogoDropShadow="true">
  <logo>chat_image.png</logo>
  <titleKey>chat_launcher_title</titleKey>
  <descriptionKey>chat_launcher_description</descriptionKey>
  <launcherType>APP</launcherType>
  <action>
    <labelKey></labelKey>
    <link><![CDATA[http://bomgar_server_name/api/start_session.ns]]></link>
    <method>GET</method>
  </action>
</launcher>
```

6. Locate the `<launchers>` sub-sections that describe the attributes of third party applications that you want to display on the Dashboard.

7. Copy the Live Chat Launcher template section and insert it as a new `<launcher>` section.

8. Omit the `<fields>` section. It is not relevant for a live chat link.

9. The `<labelKey>` section is not essential. You do not need to insert a value into this tag. The user will click the chat icon to launch the chat window.

10. Verify the `<link>` element. This link must pass the correct parameter values to connect to the chat server. For more information about the correct parameters in the link, consult the Bomgar documentation.

11. If the application is running, log out and log in again to confirm that the third party application link appears in the dashboard.

---

**Create a Cloud Service Automation launcher**

To create a Cloud Service Automation launcher from the Dashboard, Services, or Support pages, follow these steps.

**Tip:** It is always a good idea to make a backup of any system files before you make changes to them. If an error occurs later, you can always return the file to its original state.
1. Open this file with a text editor.
   
   C:\...\src-9.40\resources\client\en\Custom.properties

2. Locate the chat example in the file.

3. Change chat_launcher_title and the chat_launcher_description display text to csa_launcher_title and to csa_launcher_description respectively. When you do this, note the following:
   - The identifier can be user-defined; however, when you edit the manifest.xml file, the identifiers must match.
   - The value for each identifier is exactly what will display in the user interface.

4. Save and the file but keep it open for reference as you configure the next file.

5. Open this file with a text editor.

   C:\...\src-9.40\secure\configuration\manifest.xml

   Use the following lines as a reference for the remaining steps.

   ```xml
   <launcher launcherId="CSA_Launcher" showLogoBox="true"
   showLogoDropShadow="true">
     <logo>CSA.png</logo>
     <titleKey>CSA_Launcher_title</titleKey>
     <descriptionKey>CSA_Launcher_description</descriptionKey>
     <launcherType>APP</launcherType>
     <action>
       <labelKey></labelKey>
       <link>
         <!-- Specify the IP Address of the CSA Launcher here -->
         <![CDATA[http://<IPAddressCSA>:<port>/csa/login]]>
       </link>
       <method>GET</method>
     </action>
   </launcher>
   ```

6. Locate the <launchers></launchers> section, which contains one or more <launcher></launcher> sub-sections that describe the attributes of third party applications that you want to display on the Dashboard.

7. Copy the Live CSA Launcher template section and insert it as a new <launcher></launcher> section.
8. Omit the `<fields></fields>` section.

9. The `<labelKey>` section is not essential. You do not need to insert a value into this tag. The user will click the CSA icon to navigate to your organization's CSA Web site.

10. Verify the `<link>` element. This link must specify the correct address to CSA server.

11. If the application is running, log out and log in again to confirm that the third party application link appears in the dashboard.

Example: Get

The following example shows a launcher configuration that specifies a launcher item to trigger a Google search.

```
<launcher displayOrder="2"
     launcherId="getLauncher"
     showLogoBox="true"
     showLogoDropShadow="true"
>
  <logo>test2_app.png</logo>
  <titleKey>get_launcher_title</titleKey>
  <descriptionKey>get_launcher_description</descriptionKey>
  <launcherType>APP</launcherType>
  <fields>
    <field fieldId="q">
      <labelKey>get_search</labelKey>
      <type>TEXT</type>
      <defaultValueKey>get_search_value</defaultValueKey>
    </field>
  </fields>
  <action>
    <labelKey>get_action_label</labelKey>
    <link><![CDATA[http://www.google.com/search]]></link>
    <method>GET</method>
  </action>
</launcher>
```

**Note:** POST HTTP Requests are not supported due to issues with security that occur when executed across multiple Web application servers. However, if the target of the POST request is on the same Web application server instance as Service Request Catalog, these security issues are not present.
Appendix B: Service Catalog in Service Manager

Dynamic user options in Catalog Items

You can configure Service Request Catalog to use dynamic user options when a user is making a request. Dynamic user options provide a mechanism by which an Service Request Catalog administrator can gather additional information about a catalog item from a user based on some criteria the user previously entered. For example, users might need to have a request fulfilled within a certain time frame. In this case, the Service Request Catalog administrator can implement a check box (for example, one that uses the mandatory condition) that asks users whether they have a such a requirement. If the user does select the check box, another text field may appear (one which uses the visible condition), where the user can enter in such a requirement, specify the date, and so on.

Create a dynamic user option

- The later steps (after Step 6) in this guideline may vary somewhat, depending on which rules and conditions you have selected. For example, if you select a Pick List, the screen on which you specify the Mandatory and Visible conditions appears after the screen on which you specify the values of the Pick List. However, the functionality remains the same.

- The fields condition and query fields leverage Service Manager's Dynamic View Dependency (DVD) mechanism. Through this mechanism, you can enter an expression to evaluate some value, and determine whether the Mandatory or Visible condition is met. For example, suppose you have a check box (CheckBox1). Depending on whether that check box is selected, you want a text box to appear, so that a user can add additional information. To do this, you would set the Visible Condition to an expression that evaluates the selected status of the check box. For more information on how to write these conditions and their behaviors, see “Notes on expressions in dynamic fields” on the next page and see the Dynamic View Dependencies entry in the Service Manager Help Center.

To specify a mandatory or visible condition on a dynamic field, follow these steps:
1. In the Service Manager client, navigate Service Catalog > Administration > Manage Catalog.

2. Enter the name of the catalog item you wish to modify and then click Search.

3. Select the catalog item from the list.

   You must select an item of the “item” type. You cannot select a bundle or category.

4. In the Catalog Item Definition, click User Selections.

5. Click Add User Selection.

6. Specify a Name, a Label, and the Display Type and then click Next.

7. Specify the Type (if present), Mandatory Condition, and Visible Condition as required by your business needs.

8. Click Next. If you have specified the Display Type as Text and the Type as String, you are brought to the Validation Rule screen. Here, you may specify no validation or a validation based on a query run on some table ("Record in Table").

Notes on expressions in dynamic fields

Before you write expressions for the Dynamic fields in a catalog item, it is important that you note the following limitations and behaviors.

Supported Operators

The Condition and Query rules follow the HP Service Manager standard, except that only the following operators are supported:

- & (the AND operator)
- | (the OR operator)
- != (NOT EQUAL TO operator)
Mandatory Conditions

You can only specify a Mandatory condition for the Text, Multi-Text, and Pick List Display Type. The Mandatory condition can be specified as any of the following four types:

- Hard coded: true, false
- Variable condition: name in $lo.operator="falcon"
  
  Only global variables are supported. The global variable list is generated only when a user logs in to Service Request Catalog.

- Dependent on other widget in the user selection: [widget_name] ="falcon"

  In this example, widget_name is name of another widget in the user selection. If the dependent widget does not exist, an error message appears.

- Dependent on the fields of a catalog item outside of the user selection: $requested.for="FALCON, JENNIFER"

  Only the $requested.for,$requested.for.dept,$delivery,and $service.sla variables are supported.

Visible Conditions

You can specify a Visible condition for all Display Types.

### Mandatory and Query Condition Sample

<table>
<thead>
<tr>
<th>Type:</th>
<th>Possible values:</th>
<th>Notes:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard code condition:</td>
<td>true false</td>
<td></td>
</tr>
<tr>
<td>Variable condition:</td>
<td>company in $lo.operator=&quot;advantage&quot;</td>
<td>The company on the right is an element of the object. Only Global variables are supported</td>
</tr>
</tbody>
</table>
Mandatory and Query Condition Sample, continued

<table>
<thead>
<tr>
<th>Dependent on other widget:</th>
<th>[widget name]=&quot;specificValue&quot;</th>
<th>[widget name] is a placeholder for the name of another widget in the user selection.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dependent on the fields of catalog item outside of the user selection:</td>
<td>$requested.for=&quot;FALCON, JENNIFER&quot;</td>
<td>contact.name is a column of the selected table. Only the following values are supported in Service Request Catalog: $requested.for $Requested.type $requested.for.dept $delivery $service.sla</td>
</tr>
</tbody>
</table>

Query fields

The Query field only appears when you have specified the following attributes for your dynamic user option field:

- Display Type: Text
- Type: String
- Validation Rule: Record in Table

If your Dynamic field meets these criteria, you must then specify the following:

- Table Name: some_table
- Field Name: some_field
- Query Condition:

  - Group by Fields: This field is not supported by Service Request Catalog.

**Note:** If you click the Edit button, the advanced query window appears.

Query sample

<table>
<thead>
<tr>
<th>Type:</th>
<th>Possible values:</th>
<th>Notes:</th>
</tr>
</thead>
</table>
### Query sample, continued

<table>
<thead>
<tr>
<th>Hard code condition</th>
<th>Variable condition</th>
<th>Dependent on other widget</th>
<th>Dependent on the fields of catalog item outside of the user selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>display.currency.code=$G.my.currency</td>
<td>name = [widget name]</td>
<td>contact.name=$requested.for</td>
</tr>
<tr>
<td>false</td>
<td>company = company in $lo.operator</td>
<td></td>
<td>contact.name is a column of the selected table. Only the following values are supported in Service Request Catalog: $requested.for $Requested.type $requested.for.dept $delivery $service.sla</td>
</tr>
<tr>
<td>contact.name=&quot;FALCON, JENNIFER&quot;</td>
<td>The company on the left is a column of the selected table. The company on the right is an element of the object.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Other Behaviors

For any set custom Dynamic user option fields, you can very easily create a complicated set of rules. Therefore, it is important that you recognize the following behaviors:

- By default, the Mandatory condition is false. If the condition format is not correct, or any JavaScript function call does not exist, then the user option is not mandatory.

- By default, the Visible condition is true. If the condition format is not correct, or any JavaScript function call does not exist, then the user option is visible.

- If a Mandatory condition for a user option is true, but the visible condition is false, a request for that item can still be submitted successfully.
- If the condition format is not correct, or any JavaScript function call does not exist, an empty list is returned.

- If a user option has dependencies on two other user selected fields and the first field has no value, whether second field returns all or empty is decided by the condition expression of the second field. 

- A field that is dependent on another field is cleared automatically if the second field is cleared.

- The Mandatory designation only exists in the Service Request Catalog user interface. In Service Manager, an error message will display when a user does not set a value for mandatory field.

- The Visible behavior only exists in the Service Request Catalog user interface.

- In the case of a circular query condition (where A depends on B, and B depends on A), an empty set is returned.

- In the case of a circular Mandatory condition (where A depends on B, and B depends on A), an empty set is returned.

- In the case of a circular Visible condition (where A depends on B, and B depends on A), an empty set is returned.

- You cannot delete a user selection from Service Manager if another field or user selection depends on it.

- The Query Condition does not support the currency exchange rate. This is due to a limitation in Service Manager, which supports dollar values as numbers, instead of currency. To work around this issue, you can display the currency as a number together with a list that specifies the actual currency (for example, USD or RMB), and the approving manager’s name based on your business requirements (if the request is over 500 USD, for example).

Example: Create two dependent Dynamic User Options

To help you further understand Dynamic User Options, this section shows you an example about how to create two dependent Dynamic User Options (that is, the value of a User Option depends on what you select in another User Option) for a catalog item by using dependency expressions. For more information on how to write expressions, see Notes on expressions in dynamic fields and the Dynamic View Dependencies entry in the Service Manager Help Center.
Suppose a catalog item has two dependent Dynamic User Options: Select your computer and Select the software to uninstall. When you request this item, the lookup table of the Select your computer User Option lists all the devices that are belonged to the user name you request for. After you select a device, the lookup table of the Select the software to uninstall User Option lists all software installed on the selected computer in the Select your computer User Option.

To do this, follow these steps:

**Create the Select your computer User Option**

1. In the Service Manager client, navigate to Service Catalog > Administration > Manage Catalog.

2. Enter the name of the catalog item you wish to modify and then click Search.

3. Select the catalog item from the list.

   You must select an item of the "item" type. You cannot select a bundle or category.

4. In the Catalog Item Definition tab, click User Selections.

5. Click Add User Selection.

6. Populate the fields in the form as follows:

   - Name: computer
   - Label: Select your computer
   - Display Type: Text

7. Click Next.

8. Select String in the Type drop-down list, and then specify the Mandatory Condition and Visible Condition fields as required by your business needs.

9. Click Next.

10. Select Record in Table in the Validation Rule option button.

11. Populate the fields in the form as follows:

   - Table Name: device
   - Field Name: title
Query: type="computer" and contact.name=requested.for in $L.file

12. Click Next.

Now you have create a “Select your computer” Dynamic User option for your desired catalog item. When you request this item, the lookup table retrieves records from the device table and only lists all the computers that are belonged to the user name you have selected in the This request is for field for this item.

Create the Select the software to uninstall User Option

13. Repeat step 1 to 5.

  Note: Make sure to select the same catalog item you have selected for the Select your computer Dynamic User option.

14. Populate the fields in the form as follows:

   - Name: software
   - Label: Select the software to uninstall
   - Display Type: Text

15. Repeat step 7 to 10.

16. Populate the fields in the form as follows:

   - Table Name: pcsoftware
   - Field Name: Software Key
   - Query: logical.name=[computer]

  Note: In this expression, the logical.name list in the pcsoftware table is the same as the computer name list in the device table. Therefore, make sure that the logical.name column has the same data type as the computer column when you define the pcsoftware table.

17. Click Next.

Now you have create a "Select the software to uninstall" Dynamic User option for the same catalog item. When you request this item, if you select a computer in the Select your computer field, the lookup
table of the **Select the software to uninstall** field retrieves records from the pcsoftware table and only lists software installed on which computer you have selected in the **Select your computer** field.

### Multiple selections for Service Catalog item user options

SRC supports array data type for Service Catalog item user options.

As an administration, once you have added multiple selections for Service Catalog item user options in Service Manager, SRC users can select multiple values (For example, multiple CIs from the *device* table) in an Item Option field when ordering a catalog item.

For information about how to define and apply a user selection template, see the *HP Service Manager 9.40 Help Center*.

### Add multiple user selections

To add multiple user selections for a Service Catalog item, follow these steps:

1. Log in to the Service Manager web client as a system administrator.

2. Navigate to the **Service Catalog > Administration > Manage Catalog**.

3. Type or select search criteria for the item you want to modify and click **Search**.

   Click the item in the search list to select it, if necessary. The catalog item definition record opens.

   **Note:** You must select an item of the "item" type. You cannot select a bundle or category.

4. On the **Definition tab** in the User Selections page, click **Add User Selection**.

   The Dynamic Filed Description form opens.

5. Populate the fields in the form as follows:

   - In the **Name** field, type a field name for the user selection. For example, City.

     **Note:** The field name must be alphanumeric, contain no spaces, and start with a character.
In the **Label** field, type a display name for the user selection. For example, City.

- In the **Display Type** drop-down list, select **Text**.

6. Click **Next**.

7. Complete the following fields:

- In the **Type** drop-down list, select **String**.

- Specify the **Mandatory Condition** and **Visible Condition** fields based on your business needs. For information on how to write these conditions, see "Notes on expressions in dynamic fields" on page 73

**Note:** For a Service Catalog item, a user option configured with multiple user selections cannot be used in the condition expression of other user options. Otherwise, the condition might not work. Which means if you specify a dynamic user option (A) depending on another user option configured with multiple selections (B), when you select multiple values in the B user option, the A user option cannot work as expected.

8. Click **Next**.

9. In the **Validation Rule** option box, select **Record in Table**.

10. Populate the fields in the form as follows:

- In the **Table Name** field, type or select the name of a lookup table.

- In the **Field Name** field, type or select a field of the lookup table as a lookup field.

- In the **Multiple Selections** field, type an integer equal to or greater than 2 to allow users to select multiple values that match the selected table and fields.

- Complete the other fields as needed.

11. Click **Next**.

Now you have successfully created multiple user selections for one service catalog item.

To edit the User Selection, double-click it. The User Selection modification wizard opens. You can click **Next** to modify the User Selection, and click **Finish** to save it.
Service Request Catalog HTML tag support

HP Service Manager supports a variety of HTML tags as part of the Service Catalog. However, Service Request Catalog supports only a subset of the HTML tags. Therefore, we recommend that you refine the Service Catalog in Service Manager to ensure that catalog items are rendered as expected in Service Request Catalog. The following table shows the tags that are supported in Service Request Catalog.

Note: HTML tags that are not listed in the following table are not supported.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
</table>
| Anchor | The `<a>` tag creates a hypertext link, and supports the following attributes:  
  - target: This attribute is ignored by Service Request Catalog. By default, all URLs are opened in a new browser window to prevent Service Request Catalog from being unloaded.  
  - href: Specifies a URL. The URL can be either absolute or relative to the location of `main.jsp` of Service Request Catalog. An example of an absolute reference to a URL is `http://www.hp.com`; an example of a relative reference is `index.html`.  
  Note: You must prefix a URL with `http(s)://` to designate it as an absolute URL. Otherwise, Service Request Catalog treats it as a relative URL. |
<p>| Bold   | The <code>&lt;b&gt;</code> tag renders text as bold. |
| Break  | The <code>&lt;br&gt;</code> tag creates a line break in the text. |
| Emphasis | The <code>&lt;em&gt;</code> tag displays the tagged text in italics. An italic typeface must be available for the font used. |</p>
<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Font tag</td>
<td>The <code>&lt;font&gt;</code> tag specifies a font or list of fonts to display the text. The font tag supports the following attributes:</td>
</tr>
<tr>
<td></td>
<td>• color: Only hexadecimal color (#FFFFFF) values are supported.</td>
</tr>
<tr>
<td></td>
<td>• face: Specifies the name of the font to use. You can specify a list of comma-delimited font names, in which case Service Request Catalog selects the first available font. If the specified font is not installed on the local computer, Service Request Catalog selects a substitute font.</td>
</tr>
<tr>
<td></td>
<td>• size: Specifies the size of the font. The valid value is an integer between 1 and 7; or a relative increase from base font size 3 (for example, “+1” means 4 and “-2” means 1). All relative sizes also belong to the scale of 1 to 7.</td>
</tr>
<tr>
<td></td>
<td>• letterspacing: Specifies the tracking (manual kerning) in pixels to be applied to the right of each character.</td>
</tr>
<tr>
<td></td>
<td>• kerning: Specifies whether kerning is enabled or disabled. A non-zero value enables kerning, while zero disables it.</td>
</tr>
<tr>
<td>Image</td>
<td>The <code>&lt;img&gt;</code> tag lets you embed external image files (JPEG, GIF, PNG) and SWF movies inside text. The <code>img</code> tag supports the following attributes:</td>
</tr>
<tr>
<td></td>
<td>• src: Specifies the URL to an image or SWF file. This attribute is required; all other attributes are optional. The images or SWF movies do not show until they are downloaded completely.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Embedded SWF movies cannot access the network unless there’s no sandbox security violation.</td>
</tr>
<tr>
<td></td>
<td>• width: The width of the image being inserted, in pixels.</td>
</tr>
<tr>
<td></td>
<td>• height: The height of the image being inserted, in pixels.</td>
</tr>
<tr>
<td>Italic</td>
<td>The <code>&lt;i&gt;</code> tag displays the tagged text in italics. An italic typeface must be available for the font used.</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| List           | The `<ol>` tag (ordered list) and `<ul>` tag (unordered list) are rendered in an identical manner except that ordered list items are numbered and unordered list items are not. Both types of lists are made up of sequences of list items defined by the `<li>` tag (list item). Service Request Catalog does not support any attributes of the list tags. Lists can be nested as shown in the following examples:  
```html
<ul>
    <li> ... level one, number one... </li>
</ul>
```
| Strike         | The `<s>` and `<strike>` tags render strike-through style text.                                                                                   |
| Strong         | The `<strong>` tag renders text as bold.                                                                                                            |
| Paragraph      | The `<p>` tag creates a new paragraph. The `<p>` tag supports the following attributes:                                                          |
|                |  - align: Specifies alignment of text within the paragraph; valid values are left, right, justify, and center.                                   |
| Text format tag| The `<textformat>` tag is not a standard HTML tag but a customized tag in Service Request Catalog. It lets you use a subset of paragraph formatting attributes including the following:  
|                |  - indent: Specifies the indentation from the left margin to the first character in the paragraph. Both positive and negative numbers are acceptable. |
|                |  - blockindent: Specifies the indentation applied to all lines of the paragraph.                                                                |
|                |  - leftmargin: Specifies the left margin of the paragraph, in points.                                                                            |
|                |  - rightmargin: Specifies the right margin of the paragraph, in points.                                                                        |
|                |  - leading: Specifies the leading (line height) measured in pixels between a line’s ascent and the previous line’s descent.                 |
| Underline tag  | The `<u>` tag underlines the tagged text.                                                                                                        |
Refining the Service catalog

Remove empty categories

Empty Service Manager categories (that contain no items) cause performance degradation when Service Request Catalog attempts to populate existing categories with child items. Service Catalog administrators should verify that the Service Manager Service Catalog contains no empty categories.

Add images for catalog items

You obtain optimum results if all catalog images are the same size and in a similar format. Catalog items display as a “thumbnail” image. Follow these basic rules for attaching a thumbnail image to a catalog item:

- The recommended size for an image is 196x140 pixels. Use a good image utility to crop or resize your images to a consistent size.

- The default background color for a smaller image is white. For consistency, consider adding an appropriate background to maintain the same image dimensions for all items.

- The recommended file type is .jpg or .png. Using other formats may produce unpredictable results.

Separate catalog items for different companies in multi-company mode

These steps should be completed by an experienced Service Manager administrator. An administrator can configure Service Manager to support Multi-Company mode to filter the information that Service Request Catalog users see when making requests. In Multi-Company mode, Service Request Catalog users see only request items for their own company.

An administrator can also enable the Mandanten feature with Multi-Company mode to store company catalog data in its own secure database.

Use the Service Manager Windows Client to complete these steps.
1. To enable Multi-Company mode in the System Information Company record, click 
   System Administration > Base System Configuration > Miscellaneous > System Information 
   Record > General.

2. Verify that you have a complete company record with Multi-Company enabled. Click 
   Tailoring > Database Manager > Table (company) > Search > company > Search > Show Company 
   in Multi-Company Lists.

3. Add a new “company” field of type character type to the svcCatalog tables and to the svcDisplay 
   table.

4. Enable Mandanten security by specifying a filtering condition for the svcCatalog and svcDisplay 
   tables. Use the Mandanten Field Restriction form to specify the company value (that you specified 
   in step 3) in the Mandant Field Name field. Create two records: one for File Name svcCatalog and 
   one for File Name svcDisplay. Click 
   System Administration > Ongoing Maintenance > Mandanten > Mandanten Field Restrictions.

5. Create a Mandanten security group. Make sure the Security ID is a an upper case value. For 
   example, Service Request Catalog is an upper case value. Click 
   System Administration > Ongoing Maintenance> Mandanten > Mandanten Security Groups.

6. Type the “company” value in the first row of the Include Value List. Leave the Exclude Value List 
   empty.

7. Click Add to save the security group record.

8. If you enable Mandanten security, add the name of the Mandanten security group to the Security 
   Groups tab on the operator record of each Service Request Catalog user.

9. Assign the company to each category, sub-category, package, and item. This property does not 
   cascade.

   **Note:** If you are an advanced user, you can use these RAD Debugger commands to accomplish 
   this task:
   
   - d company in $L.file
   
   - x company in $L.file="company name"

10. Click Save.
11. If you have more than one company defined in Service Manager, make sure each company has categories, subcategories, and product types assigned associated with service catalog.

For the detailed steps to complete each Service Manager task, see the Service Manager Help server.

Working with the Support catalog

HP Service Manager self-service users can request support items only through the Service Request Catalog. Users can only request generic support tickets from the Enterprise Self Service (ESS) portal (via the “Submit a Request” selection). The Service Manager administrator must configure the support categories and category items that the user selects in the Service Request Catalog interface.

For an introduction to this process, you can begin by starting a Service Manager Windows client session and expand the left navigation pane. Follow the Example suggestions to create a sample category and item. Your actual Service Catalog and item configuration are dependent on your particular organization’s requirements.

For more information, see the Service Catalog section of the Service Manager Help system.

Search for a support catalog category or item

To search for a support catalog category or item, follow these steps:

1. Click **Service Catalog > Administration > Manage Catalog**.

2. Type any available information into the appropriate fields.

3. Click the **Support Item** checkbox to narrow the search scope.

4. Click **Search**. Service Manager displays a list of all support categories and items in the catalog.

Create a support catalog category

To create Support Catalog categories that contain only sub-categories and items that are related to support, follow these steps:
1. Click Service Catalog > Administration > Manage Catalog.

2. Click Add New Category. Service Manager displays a wizard that prompts you for the information about a new category.

3. Type a name in the Category Name field.
   
   **Example:** Type PCHelp.

4. Type a name in the Display Name field that appears in the interface as a category that you can search for items.
   
   **Example:** Type PC Help.

5. Type a description in the Description field for the new category.
   
   **Example:** Type This is a new category.

6. Click Next.

7. Make the new category a Top Level category, or you can click the drop-down list to make it a Subcategory of a selected Top Level category.
   
   **Example:** Choose Top Level.

8. Specify whether the new category should contain Subcategories, or Items and/or Bundles. Support categories can contain only items, not bundles.
   
   **Example:** Choose Items and/or Bundles to allow support items to be added to the category.

9. Click Next.

10. When Service Manager displays a message that it added a new Service Catalog category, click OK. Service Manager displays the Category Definition interface where you can refine the category by choosing an Owner or even attaching an image.

11. To create a Support category, select the Non-cart category checkbox.

12. Select the Support category checkbox.

13. Click the Access tab.

14. Click the first empty row in the Available to table.
15. Select the Service Catalog group capability word that you want to associate with this new category.

   **Example:** svcCatEmployeeRequester.

16. Add as many capability word groups to the list as necessary.

17. Click **Save**.

### Create a support catalog item

You can create individual support items for the Support Catalog, but you cannot create bundles or packages of support items. To do this, follow these steps:

1. Click **Service Catalog > Administration > Manage Catalog**.

2. Click **Add New Support Catalog Item**. Service Manager displays a wizard that prompts you for the information about a new item.

3. Type a name in the **Item Name** field.

   **Example:** Type Repair Software.

4. Select **Restrict request to a single item** in the checkbox. Support catalog items should be ordered only once.

5. Type a name in the **Display Name** field that appears in the interface as a selectable item.

   **Example:** Type Repair Software.

6. Optional. Select the checkbox for an **Information-only** item if the item is to be read-only in the catalog and not to be requested.

   **Example:** A temporarily unavailable item might appear in the catalog but not be selectable. Information-only items can also be instructions that appear in the catalog with an image, description, or even attachments, but they do not have an associated price and they cannot be requested.

7. Type a brief description in the **Description** field for the new item.

   **Example:** Type Repair email software.

8. Type a description in the **Detailed Description** field.
Example: Type Reinstall or upgrade.

9. Type the cost in the Cost field.

Example: Type 100.

Tip: For support items, you may need to gather labor and material costs to accurately reflect the true cost of the support item.

10. Select the Currency unit from the drop-down list.

Example: Use the default currency that appears.

11. Click Next.

12. Click the lookup icon to display available categories.

13. Choose a Category from the list to specify what type of record Service Manager should create when a user selects this item. Choose:
   - Complaint
   - Incident
   - Request for change
   - Request for information

Example: Choose Request for change.

14. Select a category Subarea:
   - New service
   - Upgrade/new release

Example: Choose Upgrade/new release.

Service Manager displays the New Support Catalog Item wizard.

15. If necessary, change the Category, Area, or Subarea values. Click Next.

16. Choose a connector from the Connector drop-down list. The connector is the type of record that Service Manager creates. If you do not select a connector, Service Manager creates a new
interaction with the Category, Area, and Subarea values that you specified for this Support Catalog item.

**Example:** Leave this blank.

17. Choose a category from the **Category** drop-down list.

   **Example:** Choose Applications.

18. Click **Next**.

19. When Service Manager displays a message that it added a new Service Catalog item, click **OK**.

20. Service Manager displays the Catalog Item Definition interface where you can add more information about the item, add an attachment or an image.

21. If you make changes, click **Save**

### Delete a support item

1. Click **Service Catalog > Administration > Manage Catalog**.

2. Type any available information into the appropriate fields.

3. Click the **Support Item** checkbox to narrow the search scope.

4. Click **Search**. Service Manager displays a list of all categories and items in the catalog.

5. Browse the list to find the category or item that you want to delete.

   **Example:** Repair software

6. Click **Delete**.

### Controlling the permission to create, update, or close interactions

Users whose user role or profile has the right permission can create, update, or close a support request, or create or update a service request.
Note: Closing a service request is not supported in SRC.

Disable the permission to create, update, or close interactions

When you disable a role’s or profile’s permission to create, update, and close interactions, users with the role or profile are not authorized to create, update, or close a service request or a support request in Service Request Catalog.

To disable the corresponding permission, follow these steps:

1. Log on to Service Manager as an Administrator, and then do one of the following:
   - For a Service Manager Codeless system:
     i. Navigate to System Administration > Roles.
     ii. Select a name form the Name drop-down list on the Security Role tab, for example, sysadmin.
     iii. Click Search.
     iv. Select Service Desk from the list.
   - For a Service Desk Profile:
     i. Navigate to System Administration > Ongoing Maintenance > Profiles > Service Desk Profile.
     ii. Click Search, and then select a profile name from the name list.
     Alternatively, you can type a name in the Profile Name field.

2. Do one of the following:
   - To disable the permission to create an interaction, click to clear the New checkbox.
   This disables the Request buttons of service items and support items, and the Create button of generic support when users with the role or profile log on to SRC.
   - To disable the permission to update an interaction, select Never in the Update drop-down list.
This disables the **Resubmit** buttons of service items and support items, and the **Update** button of generic support when users with the role or profile log on to SRC.

- To disable the permission to close an interaction:
  - For a Service Manager Codeless system, select **Never** in the **Delete/Closed** drop-down list.
  - For a Service Manager Classic system, click to clear the **Close** checkbox.

This disables the **Close** buttons of support request when users with the role or profile log on to SRC.

**Note:** You can specify the **Allowed Statuses** for users. In this case, the new, update, and close capabilities are available only when interactions are in a specified status.
# Appendix C: XML tag reference

<table>
<thead>
<tr>
<th>For this XML tag</th>
<th>Insert this value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;configurations&gt;</code> / <code>&lt;configuration&gt;</code></td>
<td>Define one or more <code>&lt;configuration&gt;</code> sections</td>
</tr>
<tr>
<td><code>&lt;configuration name=&quot;default&quot;&gt;</code> / <code>&lt;configuration&gt;</code></td>
<td>Define all the necessary values to describe company customization values. The company name that you configure should match the default company name that is in the end user operator record.</td>
</tr>
<tr>
<td><code>&lt;companyDisplayNameKey&gt;</code> / <code>&lt;companyDisplayNameKey&gt;</code></td>
<td>company_name is defined in <code>...\resources\client\en\Custom.properties</code>&lt;br&gt;&lt;br&gt;<em>Custom.properties Example:</em> company_name = XYZ&lt;br&gt;&lt;br&gt;<em>Example:</em>&lt;br&gt;&lt;companyDisplayNameKey&gt;company_name&lt;/companyDisplayNameKey&gt;</td>
</tr>
<tr>
<td><code>&lt;resources&gt;</code> / <code>&lt;resource&gt;</code></td>
<td>Define one or more <code>&lt;resource&gt;</code> sections within this parameter</td>
</tr>
<tr>
<td><code>&lt;resource&gt;</code> / <code>&lt;resource&gt;</code></td>
<td>panel.swf should be in this folder:&lt;br&gt;<code>...\secure\configuration\default\panel.swf</code>&lt;br&gt;&lt;br&gt;<em>Example:</em> <code>&lt;resource&gt;panel.swf&lt;/resource&gt;</code></td>
</tr>
<tr>
<td><code>&lt;styles&gt;</code> / <code>&lt;style&gt;</code></td>
<td>Define one or more <code>&lt;style&gt;</code> sections.</td>
</tr>
<tr>
<td><code>&lt;style&gt;</code> / <code>&lt;style&gt;</code></td>
<td>styles.css should be in this folder:&lt;br&gt;<code>...\secure\configuration\default\styles.css</code>&lt;br&gt;&lt;br&gt;<em>Example:</em> <code>&lt;style&gt;styles.css&lt;/style&gt;</code></td>
</tr>
<tr>
<td><code>&lt;launchersGroup&gt;</code> / <code>&lt;launchersGroup&gt;</code></td>
<td>Define the launchers that will appear on each landing page.</td>
</tr>
<tr>
<td>For this XML tag</td>
<td>Insert this value</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td><code>&lt;launchers displayOn=&quot;Landingpage&quot; showIconOnly=&quot;false&quot; showButtons=&quot;true&quot;&gt; &lt;/launchers&gt;</code></td>
<td>Identify the third party launchers that will appear on the Dashboard and other landing pages. displayOn=DASHBOARD</td>
</tr>
<tr>
<td><code>&lt;launcher displayOrder=&quot;n&quot; launcherId=&quot;id&quot; showLogoBox=&quot;true&quot; showLogoDropShadow=&quot;true&quot;&gt;</code></td>
<td>You can show the logo as an icon and apply a drop shadow to the icon. displayOrder=&quot;n&quot; - order of appearance (left-to-right and top-to-bottom) launcherid=&quot;id&quot;- a unique user-defined value for each launcher. showLogoBox=&quot;true&quot;</td>
</tr>
<tr>
<td><code>&lt;logo&gt;</code>&lt;/logo&gt;</td>
<td>Your launcher logo file should be in this folder: <code>...\secure\configuration\default\applications</code> The optimum size is 83x83px and the resolution should be 72 dpi. <strong>Example:</strong> <code>...\secure\configuration\default\applications\abc.png</code> The file format must be .png, .gif, or .jpg <strong>Example:</strong> <code>&lt;logo&gt;abc.png&lt;/logo&gt;</code></td>
</tr>
</tbody>
</table>
| `<titleKey></titleKey>` | titleKey value from the Custom.properties file. **Example:** `<titleKey>launcherName</titleKey>` The inserted ID value cannot contain blanks. Use underscore characters as separators if necessary. The display value can contain blank characters. **Custom.properties Example:**

<table>
<thead>
<tr>
<th>Id Value</th>
<th>Display Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>launcherName = XYZ Company</td>
<td></td>
</tr>
</tbody>
</table>
### For this XML tag

<table>
<thead>
<tr>
<th>Insert this value</th>
</tr>
</thead>
<tbody>
<tr>
<td>descriptionKey value from the Custom.properties file.</td>
</tr>
<tr>
<td><strong>Example:</strong> <code>&lt;descriptionKey&gt;launcherDesc&lt;/descriptionKey&gt;</code></td>
</tr>
<tr>
<td>The inserted ID value cannot contain blanks. Use underscore characters as separators if necessary. The display value can contain blank characters.</td>
</tr>
<tr>
<td><strong>Custom.properties Example:</strong></td>
</tr>
<tr>
<td><strong>Id Value</strong></td>
</tr>
<tr>
<td>launcherDesc</td>
</tr>
</tbody>
</table>

### `<launcherType>`

<table>
<thead>
<tr>
<th>Specify the type of third party link:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• APP to link to an application, like Wikipedia.</td>
</tr>
<tr>
<td>• MEDIA to open a video window.</td>
</tr>
<tr>
<td>The link type value must be upper case.</td>
</tr>
<tr>
<td><strong>Example:</strong> <code>&lt;launcherType&gt;MEDIA&lt;/launcherType&gt;</code></td>
</tr>
</tbody>
</table>

### `<fields>`

| Optional: Define one or more individual fields to appear in a popup window. |

### `<field fieldID="ID">`

| Define a field that will appear in a pop-up window. |
| Field ID is a unique user-defined value for each field. |
| **Example:** `<field fieldId="GetName">` |

### `<labelKey>`

<p>| labelKey value from the Custom.properties file. |
| Typically, this is an informational label or text string that might precede a text box or just communicate information. |
| <strong>Example:</strong> <code>&lt;labelKey{textLabel&lt;/labelKey&gt;</code> |
| <strong>Custom.properties Example:</strong> |
| <strong>Id Value</strong> | <strong>Display Value</strong> |
| textLabel | Select One: |
| or |
| textLabel | You are about to open a new window. |</p>
<table>
<thead>
<tr>
<th>For this XML tag</th>
<th>Insert this value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;type&gt;&lt;/type&gt;</code></td>
<td>Type of field. Choose one:</td>
</tr>
<tr>
<td></td>
<td>- TEXT to display any text</td>
</tr>
<tr>
<td></td>
<td>- TEXTAREA to gather user information</td>
</tr>
<tr>
<td></td>
<td>- DROPDOWN for list information</td>
</tr>
<tr>
<td></td>
<td>- HIDDEN for passing information, like a token</td>
</tr>
<tr>
<td><strong>Note:</strong> Make sure you type these values in upper case.</td>
<td></td>
</tr>
<tr>
<td><strong>Examples:</strong></td>
<td><code>&lt;type&gt;TEXT&lt;/type&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;type&gt;TEXTAREA&lt;/type&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;type&gt;DROPDOWN&lt;/type&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;type&gt;HIDDEN&lt;/type&gt;</code></td>
</tr>
<tr>
<td>`&lt;defaultValue&gt;</td>
<td>Specify a constant value to display, such as a list item or other value that you want to pre-populate with static content.</td>
</tr>
<tr>
<td>&lt;/defaultValue&gt;`</td>
<td><strong>Examples:</strong> <code>&lt;defaultValue&gt;Welcome!&lt;/defaultValue&gt;</code></td>
</tr>
<tr>
<td><code>&lt;options&gt;</code></td>
<td>Insert individual <code>&lt;option&gt;&lt;/option&gt;</code> values to define values for a dropdown list.</td>
</tr>
<tr>
<td><code>&lt;option&gt;&lt;/option&gt;</code></td>
<td>For each <code>&lt;option&gt;&lt;/option&gt;</code> tag set, insert <code>&lt;labelKey&gt;&lt;/labelKey&gt;</code> and <code>&lt;value&gt;&lt;/value&gt;</code> tags and related values.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>&lt;option&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;labelKey&gt;&lt;/labelKey&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;value&gt;&lt;/value&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/option&gt;</code></td>
</tr>
</tbody>
</table>
### For this XML tag

<table>
<thead>
<tr>
<th>For this XML tag</th>
<th>Insert this value</th>
</tr>
</thead>
</table>
| `<labelKey></labelKey>` | labelKey value from the Custom.properties file.  
In an option list, specify the key name of a single list value that you want to display in the user interface.  
**Example:** `<labelKey>dropdown_1</labelKey>`  
**Custom.properties Example:**  
<table>
<thead>
<tr>
<th>Id Value</th>
<th>Display Value</th>
</tr>
</thead>
</table>
dropdown1 = Black  
dropdown2 = Red |
| `<value></value>` | This value can be the same as you defined in the Custom.properties file, or might be a different value to be passed to the third party application server.  
**Example:**  
| `<value>color</value>` |  |
| `<action></action>` | The action describes how data will be passed to the third party application. |  
| `<link></link>` | Insert the URL for the link in a CDATA tag.  
**Example:** `<![CDATA[www.google.com]]>`  
**Example:** `<link> <![CDATA[http://www.XYZ.com]]> </link>` |  

---

**Customization Guide**  
**Appendix C: XML tag reference**
**For this XML tag** | **Insert this value**
---|---
`<method></method>` | Specify the appropriate action:
- GET enables form data to be encoded by the browser into a URL.
- POST enables form data to be encapsulated in a message.

**Note:** If the Post request is to access a different domain (server URL), you must confirm that there is a cross-domain policy file configured at the root directory of the remote domain. The cross domain policy file grants access to the sender passing transactional data to that server.

**Example:** Successful connection to a third-party application in a remote domain, like **www.XYZ.com** requires a crossdomain.xml file to exist at this location:

`www.XYZ.com/crossdomain.xml`

The crossdomain.xml file would contain a section like this:

```
<cross-domain-policy>
  <allow-access-from domain="*.ABC.com" secure="false"/>
</cross-domain-policy>
```

**Examples:**

```
<method>GET</method>
<method>POST</method>
```
Appendix D: Styling the interface

You can style Service Request Catalog to show your Company logo, change the application name, or set text formatting and the background as needed.

There are two methods you can use to style the interface. The first method is to modify the .swf files. This method is the original method for Service Request Catalog, and supports only desktop clients. However, this implementation was unsuitable for use on tablets, because of its reliance on Adobe Flash. In addition, this method supports all font changes. If you intend to use both the desktop and tablet clients of Service Request Catalog experience on tablets, you can modify the .zip files. However, this method does not support all font changes. The following table summarizes this information:
### Required elements in `manifest.xml`

<table>
<thead>
<tr>
<th>**Required elements in ** <code>manifest.xml</code></th>
<th><strong>Desktop support</strong></th>
<th><strong>Tablet support</strong></th>
<th><strong>Comments</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;swfStyles&gt;</code>style.swf&lt;/swfStyles&gt;</td>
<td>Yes</td>
<td>No</td>
<td>These two elements were present from Service Request Catalog 1.40 and are still available for compatibility. No changes are required to them due to the introduction of the .zip method.</td>
</tr>
</tbody>
</table>
| `<swfResources>`panel.swf</swfResources> |                     |                   | - The `<swfStyles>` and `<swfResources>` elements have priority over the `<cssStyles>` and `<imageResources>` elements.  
- The `<swfStyles>` style(modified).swf file is present from Service Request Catalog 9.34 and enables you to customize the current fonts to Service Request Catalog 1.40. For more information, see “Customize the font family” on page 147. |
| `<cssStyles>`font.css</cssStyles>    | Yes                 | Yes               | These two elements were newly added in Service Request Catalog 9.33 to support tablets, although your regular desktop browser will apply them as well. |
| `<imageResources>`image.zip</imageResources> |                     |                   | **Note:** The desktop browser will apply the `<cssStyles>` and `<imageResources>` elements only if the `<swfStyles>` and `<swfResources>` elements are not defined in the `manifest.xml` file. |

**Note:** You should specify the following images as follows, regardless of which customization method you use:

1. `backgroundImag`: height < 600px, width < 1600px.
2. `companyLogo`: height < 40px, width < 120px.
3. `footerLogo: height < 40px, width < 120px;`

4. `headerSeperator: height < 10px;`

5. `footerSeperator: height < 10px;`

All other graphics should be kept at their original size.
Client customization reference

The following user interface labels can be customized in SRC client customization file, which you can find in the following default location:

/resources/client/xx/custom.properties

**Note:**

- In this example path, the `xx` placeholder represents the language code of your SRC installation.
- For any custom property file, we recommend that you use Unicode encoding for any non-English characters. English characters in a non-English resource files will render in plain text as expected.

**Example:**

Suppose you want to change the default "Dashboard" label in the following screen shot to say "Welcome":

![Dashboard](image)

To implement this change, follow these steps:
1. In the file, locate the following UI text label key:
   
   `header.navigation.tab.dashboard = Dashboard`

2. Change the "Dashboard" value to "Welcome" and then press `CTRL+C` to copy this line.

3. Paste the line into the custom properties file:
   
   `/resources/client/xx/custom.properties`

4. Restart the Service Request Catalog service.

You Service Request Catalog should appear as follows:

![Dashboard with Welcome label](image)

**Note:** If the label has not changed, clear your browser cache and then restart Service Request Catalog service again.

**Customize the SRC header**
Id | Key | Value
---|---|---
1. | Configured in the manifest.xml, default key : default_company_name_label | Service Portal
2. | header.navigation.tab.dashboard | Dashboard
3. | header.navigation.tab.subscription | Subscriptions
4. | header.navigation.tab.services | Services
5. | header.navigation.tab.support | Support
6. | header.savedItem.button.label | Cart 0

Customize the SRC footer

![Customized SRC footer](image)

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
</table>
| 1. | default_company_name_label | Service Portal
    (Configured in the manifest.xml, default key) |
| 2. | footer.slogan.label | IT Service and Support |
| 3. | footer.navigation.link.dashboard | Dashboard |
| 4. | footer.navigation.link.account | My Account |
| 5. | footer.navigation.link.services | Services |
| 6. | footer.navigation.link.support | Support |
Customize the Dashboard

**Dashboard**

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>dashboard.dashboard_landing.title</td>
<td>Dashboard</td>
</tr>
<tr>
<td>2.</td>
<td>dashboard.dashboard_landing.message</td>
<td>Everything you want ...</td>
</tr>
<tr>
<td>3.</td>
<td>dashboard.approval_status</td>
<td>Your Approvals</td>
</tr>
<tr>
<td>4.</td>
<td>dashboard.approval_widget_instruction</td>
<td>Review these requests</td>
</tr>
<tr>
<td>5.</td>
<td>dashboard.view_approval_details</td>
<td>Approval Pending</td>
</tr>
<tr>
<td>6.</td>
<td>dashboard.approval_status</td>
<td>Your Approvals</td>
</tr>
<tr>
<td>7.</td>
<td>dashboard.request_widget_title</td>
<td>Your Requests</td>
</tr>
<tr>
<td>8.</td>
<td>dashboard.sm_services_request_widget_instruction</td>
<td>Review your request ...</td>
</tr>
<tr>
<td>9.</td>
<td>dashboard.sm_services_request_status</td>
<td>Service</td>
</tr>
<tr>
<td>10.</td>
<td>account_view.status_pending</td>
<td>Pending</td>
</tr>
<tr>
<td>11.</td>
<td>account_view.status_approved</td>
<td>Approved</td>
</tr>
<tr>
<td>12.</td>
<td>dashboard.support_request_status</td>
<td>Support</td>
</tr>
<tr>
<td>13.</td>
<td>account_view.status_open</td>
<td>Open</td>
</tr>
<tr>
<td>14.</td>
<td>account_view.status_closed</td>
<td>Closed</td>
</tr>
</tbody>
</table>
### Customize the Services page

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>dashboard.service_landing.title</td>
<td>Services</td>
</tr>
<tr>
<td>2.</td>
<td>dashboard.service_landing.message</td>
<td>Service Catalog and ...</td>
</tr>
<tr>
<td>3.</td>
<td>dashboard.popular_items</td>
<td>Popular Requests</td>
</tr>
<tr>
<td>4.</td>
<td>dashboard.popular_items_instruction</td>
<td>There are the most requested ...</td>
</tr>
<tr>
<td>5.</td>
<td>dashboard.request_on_behalf_title</td>
<td>Service Assistant</td>
</tr>
<tr>
<td>6.</td>
<td>dashboard.request_on_behalf_instruction</td>
<td>Request Services for another ...</td>
</tr>
</tbody>
</table>
Customize the Support page

1. dashboard.support_landing.title: Support
2. dashboard.support_landing.message: Support Catalog and...
3. dashboard.support_title: Support
4. dashboard.support_instruction: Request an item that ...
5. buttonTextCreate: Create
6. dashboard.assistance: Assistance
7. dashboard.assistance_instruction: Get more task and ...
8. dashboard.getting_started: What’s new?
9. dashboard.faqs: Request a service?
10. dashboard.info: Request support?
11. dashboard.assistance_portal: Help Portal
Customize the Account View page

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>account_view.order_id</td>
<td>Request ID</td>
</tr>
<tr>
<td>2.</td>
<td>account_view.title</td>
<td>Request Name</td>
</tr>
<tr>
<td>3.</td>
<td>account_view.price</td>
<td>Cost (Price)</td>
</tr>
<tr>
<td>4.</td>
<td>account_view.view</td>
<td>View</td>
</tr>
<tr>
<td>5.</td>
<td>account_view.your_service_request</td>
<td>Your Service Requests</td>
</tr>
<tr>
<td>6.</td>
<td>account_view.your_support_request</td>
<td>Your Support Requests</td>
</tr>
<tr>
<td>7.</td>
<td>account_view.status_approved</td>
<td>Approved</td>
</tr>
<tr>
<td>8.</td>
<td>account_view.status_pending</td>
<td>Pending</td>
</tr>
<tr>
<td>9.</td>
<td>account_view.status_denied</td>
<td>Denied</td>
</tr>
<tr>
<td>10.</td>
<td>account_view.status_closed</td>
<td>Closed</td>
</tr>
<tr>
<td>11.</td>
<td>account_view.status_open</td>
<td>Open</td>
</tr>
<tr>
<td>12.</td>
<td>account_view.status_closed</td>
<td>Closed</td>
</tr>
<tr>
<td>13.</td>
<td>account_view.request_inbox</td>
<td>Request Inbox</td>
</tr>
<tr>
<td>14.</td>
<td><em>This box takes its value from numbers, 7,8,9,10, and 11, as appropriate.</em></td>
<td>N/A</td>
</tr>
</tbody>
</table>
Customize the Details View page

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>account_view.request_placed_on</td>
<td>Request place on</td>
</tr>
<tr>
<td>2.</td>
<td>account_view.approvers</td>
<td>Approvers</td>
</tr>
<tr>
<td>3.</td>
<td>account_view.request_by</td>
<td>Requested By</td>
</tr>
<tr>
<td>4.</td>
<td>account_view.contact_info</td>
<td>Contact Information</td>
</tr>
</tbody>
</table>
Customize the Approval history

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>account_view.header_approval_status</td>
<td>Additional Details</td>
</tr>
<tr>
<td>2.</td>
<td>account_view.button_hide_approval_history</td>
<td>VIEW DETAILS</td>
</tr>
<tr>
<td>3.</td>
<td>interaction_history.section_name.history</td>
<td>History</td>
</tr>
<tr>
<td>4.</td>
<td>interaction_history.section_name.activity_logs</td>
<td>Activity Logs</td>
</tr>
<tr>
<td>5.</td>
<td>interaction_history.section_name.approval_logs</td>
<td>Approval Logs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>account_view.header_approval_status</td>
<td>Additional Details</td>
</tr>
<tr>
<td>2.</td>
<td>account_view.button_view_approval_history</td>
<td>COMMENT &amp; HISTORY</td>
</tr>
<tr>
<td>3.</td>
<td>account_view.header_line_items</td>
<td>Line Items</td>
</tr>
</tbody>
</table>
Customize the Request Inbox Welcome page

1. ✉️ Check the Status of Your Requests

2. Hi, Jennifer Falcon! This is your Account summary. It contains information about the status of your requests.

3. Browse your Inbox to see your requests sorted by their current status.

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>account_view.welcome.header</td>
<td>Check the Status of ...</td>
</tr>
<tr>
<td>2.</td>
<td>account_view.welcome</td>
<td>Hi, Jennifer ...</td>
</tr>
<tr>
<td>3.</td>
<td>account_view.instruction</td>
<td>Browser your inbox ...</td>
</tr>
</tbody>
</table>

Customize the Browse Catalog button

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>header.search.openCatalog.button.label</td>
<td>Browse Catalog</td>
</tr>
<tr>
<td>2.</td>
<td>header.search.closedCatalog.button.label</td>
<td>Hide Catalog</td>
</tr>
</tbody>
</table>
Customize Your Subscriptions

Your Subscriptions

Track and modify your subscriptions here.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Personal - All</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>subscription.header.title</td>
<td>Your Subscriptions</td>
</tr>
<tr>
<td>2</td>
<td>subscription.header.description</td>
<td>Track and modify ...</td>
</tr>
<tr>
<td>3</td>
<td>subscription.itemlist.header.name</td>
<td>Name</td>
</tr>
<tr>
<td>4</td>
<td>subscription.itemlist.header.subscriber</td>
<td>Subscriber</td>
</tr>
<tr>
<td>5</td>
<td>subscription.itemlist.header.date</td>
<td>Date</td>
</tr>
<tr>
<td>6</td>
<td>subscription.itemlist.header.status</td>
<td>Status</td>
</tr>
</tbody>
</table>
## MyDevices

### Subscription Details
1. **Id**: 1  
   **Key**: subscription.itemDetail.header.title  
   **Value**: Subscription Details
2. **Id**: 2  
   **Key**: subscription.itemDetail.subscriber  
   **Value**: Subscriber
3. **Id**: 3  
   **Key**: subscription.itemDetail.subscriptionType  
   **Value**: Subscription Type
4. **Id**: 4  
   **Key**: subscription.itemDetail.associatedRequest  
   **Value**: Associated Request
5. **Id**: 5  
   **Key**: subscription.itemDetail.serviceSLA  
   **Value**: Service Level Package

### Subscription History
6. **Id**: 6  
   **Key**: subscription.itemDetail.history.title  
   **Value**: Subscription History
7. **Id**: 7  
   **Key**: subscription.itemDetail.history.for  
   **Value**: for
8. **Id**: 8  
   **Key**: subscription.itemDetail.history.status  
   **Value**: Status
9. **Id**: 9  
   **Key**: subscription.itemDetail.history.timestamp  
   **Value**: Timestamp
10. **Id**: 10  
    **Key**: subscription.itemDetail.history.Notes  
    **Value**: Notes
11. **Id**: 11  
    **Key**: subscription.itemDetail.history.noresult  
    **Value**: No Change history for ...
12. **Id**: 12  
    **Key**: subscription.itemDetail.contolPanel.statusLabel  
    **Value**: Subscription Status
13. **Id**: 13  
    **Key**: subscription.itemDetail.contolPanel.unsubscriberButton  
    **Value**: unsubscribe
14. **Id**: 14  
    **Key**: subscription.itemDetail.header.return  
    **Value**: Return to Your Subscriptions
Customize the Approvals page

1. Your Approvals
2. Items that need your approval are listed here

<table>
<thead>
<tr>
<th>ID</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>header.title</td>
<td>Your Approvals</td>
</tr>
<tr>
<td>2</td>
<td>header.description</td>
<td>Item that need ...</td>
</tr>
<tr>
<td>3</td>
<td>list.column.id</td>
<td>ID</td>
</tr>
<tr>
<td>4</td>
<td>list.column.title</td>
<td>Title</td>
</tr>
<tr>
<td>5</td>
<td>list.column.type</td>
<td>Type</td>
</tr>
<tr>
<td>6</td>
<td>list.column.requestedFor</td>
<td>Requested for</td>
</tr>
<tr>
<td>7</td>
<td>list.column.date</td>
<td>Date</td>
</tr>
<tr>
<td>8</td>
<td>list.column.cost</td>
<td>Cost</td>
</tr>
</tbody>
</table>

**Note:** The Approvals Page also requires some customization on the SRC server side. For more information, see "Customize Your Approvals" on page 138
Customize the Service Approval Details page

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>service.detail</td>
<td>Service Details</td>
</tr>
<tr>
<td>2</td>
<td>account_view.request_placed_on</td>
<td>Request placed on</td>
</tr>
<tr>
<td>3</td>
<td>account_view.approvers</td>
<td>Approvers</td>
</tr>
<tr>
<td>4</td>
<td>request.information</td>
<td>Request Information</td>
</tr>
<tr>
<td>5</td>
<td>account_view.request_by</td>
<td>Requested By</td>
</tr>
<tr>
<td>6</td>
<td>account_view.contact_info</td>
<td>Contact Information</td>
</tr>
<tr>
<td>7</td>
<td>request.information.callbackType</td>
<td>Contact by</td>
</tr>
<tr>
<td>8</td>
<td>request.information.urgency</td>
<td>Priority</td>
</tr>
<tr>
<td>9</td>
<td>request.information.status</td>
<td>Fulfillment</td>
</tr>
<tr>
<td>10</td>
<td>detail.back</td>
<td>Return to your Approvals</td>
</tr>
<tr>
<td>11</td>
<td>account_view.approve_button</td>
<td>APPROVE</td>
</tr>
<tr>
<td>12</td>
<td>account_view.deny_button</td>
<td>DENY</td>
</tr>
</tbody>
</table>
Customize the Change Approval Details page

### Change Details

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>approval.change.detail.title</td>
<td>Change Details</td>
</tr>
<tr>
<td>2</td>
<td>approval.change.detail.label.status</td>
<td>Status</td>
</tr>
<tr>
<td>3</td>
<td>approval.change.detail.label.approvalStatus</td>
<td>Approval Status</td>
</tr>
<tr>
<td>4</td>
<td>approval.change.detail.label.currentPhase</td>
<td>Current Phase</td>
</tr>
<tr>
<td>5</td>
<td>approval.change.detail.label.requestedEndDate</td>
<td>Requested End Date</td>
</tr>
<tr>
<td>6</td>
<td>approval.change.detail.label.changeCoordinator</td>
<td>Change Coordinator</td>
</tr>
<tr>
<td>7</td>
<td>approval.change.detail.label.initiatedBy</td>
<td>Initiated By</td>
</tr>
<tr>
<td>8</td>
<td>approval.change.detail.label.assignedGroup</td>
<td>Assigned Group</td>
</tr>
<tr>
<td>9</td>
<td>approval.change.detail.label.affectedConfigurationItem</td>
<td>Affected Configuration Item</td>
</tr>
<tr>
<td>10</td>
<td>approval.change.detail.label.category</td>
<td>Catalog</td>
</tr>
</tbody>
</table>
Customize the Request Approval Details page

The legacy Request Approval module

Id | Key | Value
---|-----|-----
1. | approval.quote.detail.title | Request Details
2. | approval.quote.detail.requestedFor | Request For
3. | approval.quote.detail.requestedBy | Request By
4. | approval.quote.detail.requestedOn | Request On
The new Process Designer Request Approval module

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.</td>
<td>approval.quote.detail.status</td>
</tr>
<tr>
<td>6.</td>
<td>approval.quote.detail.approvalStatus</td>
</tr>
<tr>
<td>7.</td>
<td>list.column.approvals</td>
</tr>
<tr>
<td>8.</td>
<td>approval.quote.detail.currentPhase</td>
</tr>
<tr>
<td>9.</td>
<td>approval.quote.item.title</td>
</tr>
<tr>
<td>10.</td>
<td>approval.quote.item.id</td>
</tr>
<tr>
<td>11.</td>
<td>approval.quote.item.quantity</td>
</tr>
<tr>
<td>12.</td>
<td>approval.quote.item.description</td>
</tr>
<tr>
<td>13.</td>
<td>approval.quote.item.status</td>
</tr>
<tr>
<td>14.</td>
<td>approval.quote.item.price</td>
</tr>
<tr>
<td>15.</td>
<td>approval.quote.return</td>
</tr>
<tr>
<td>16.</td>
<td>account_view.approve_button</td>
</tr>
<tr>
<td>17.</td>
<td>account_view.deny_button</td>
</tr>
</tbody>
</table>
### Need pad bundle for new employee

**Request Details**

- **Requested For**: FALCON, JENNIFER
- **Requested By**: FALCON, JENNIFER
- **Status**: Open
- **Approval Status**: Pending
- **Current Phase**: Logging
- **Category**: Generic Request
- **Subcategory**: Employee On-boarding
- **Delivery Date**: October 23, 2014 8:53:38 PM
- **Assignment Group**: Application
- **Priority**: Emergency Request
- **Impact**: Enterprise
- **Urgency**: Critical

**Line Items**

- **POA Bundle**

#### Description

Provides employees with a complete mobility solution supported by iPaq/PDA technology.

<table>
<thead>
<tr>
<th>Option</th>
<th>Requested For</th>
<th>Submitted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile Phone Service</td>
<td>Jennifer Falcon</td>
<td>Submitted</td>
</tr>
<tr>
<td>International Roaming</td>
<td>Galaxy Note 3</td>
<td></td>
</tr>
<tr>
<td>iPaq</td>
<td>Extra Battery</td>
<td>$75.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>approval.quote.detail.title</td>
<td>Request Details</td>
</tr>
<tr>
<td>2</td>
<td>approval.quote.detail.requestedFor</td>
<td>Request For</td>
</tr>
<tr>
<td>3</td>
<td>approval.quote.detail.requestedBy</td>
<td>Request By</td>
</tr>
<tr>
<td>4</td>
<td>approval.quote.detail.status</td>
<td>Status</td>
</tr>
<tr>
<td>5</td>
<td>approval.quote.detail.approvalStatus</td>
<td>Approval Status</td>
</tr>
<tr>
<td></td>
<td>Key</td>
<td>Description</td>
</tr>
<tr>
<td>---</td>
<td>----------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>6</td>
<td>approval.quote.detail.currentPhase</td>
<td>Current Phase</td>
</tr>
<tr>
<td>7</td>
<td>approval.quote.detail.category</td>
<td>Category</td>
</tr>
<tr>
<td>8</td>
<td>approval.quote.detail.subCategory</td>
<td>Subcategory</td>
</tr>
<tr>
<td>9</td>
<td>approval.quote.detail.deliveryDate</td>
<td>Delivery Date</td>
</tr>
<tr>
<td>10</td>
<td>approval.quote.detail.assignedGroup</td>
<td>Assignment Group</td>
</tr>
<tr>
<td>11</td>
<td>approval.quote.detail.priority</td>
<td>Priority</td>
</tr>
<tr>
<td>12</td>
<td>approval.quote.detail.impact</td>
<td>Impact</td>
</tr>
<tr>
<td>13</td>
<td>approval.quote.detail.urgency</td>
<td>Urgency</td>
</tr>
<tr>
<td>14</td>
<td>approval.quote.bundle.description</td>
<td>Description</td>
</tr>
<tr>
<td>15</td>
<td>approval.quote.return</td>
<td>Return to your Approvals</td>
</tr>
<tr>
<td>16</td>
<td>account_view.approve_button</td>
<td>APPROVE</td>
</tr>
<tr>
<td>17</td>
<td>account_view.deny_button</td>
<td>DENY</td>
</tr>
</tbody>
</table>
Customize the Search Results page

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>searchView.catalog</td>
<td>Catalog</td>
</tr>
<tr>
<td>2</td>
<td>searchView.service</td>
<td>Service</td>
</tr>
<tr>
<td>3</td>
<td>searchView.support</td>
<td>Support</td>
</tr>
<tr>
<td>4</td>
<td>searchView.knowledge</td>
<td>Knowledge</td>
</tr>
<tr>
<td>5</td>
<td>searchView.header.service</td>
<td>Service</td>
</tr>
<tr>
<td>6</td>
<td>results.number</td>
<td>Results (Total)</td>
</tr>
<tr>
<td>7</td>
<td>searchView.header.knowledge</td>
<td>knowledge</td>
</tr>
<tr>
<td>8</td>
<td>searchView.header.support</td>
<td>Support</td>
</tr>
<tr>
<td>9</td>
<td>support.generic.request.widget.title</td>
<td>More Support</td>
</tr>
</tbody>
</table>
Customize the request on Behalf Of

1. Click Browse Catalog in the header to select Categories and Sub-Categories. Or, you can type the name of the item or service that you need in the Search box.

2. Remember that services requested for another person are not delivered to you.

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>services.impersonate.welcome.header</td>
<td>Click Browse...</td>
</tr>
<tr>
<td>2.</td>
<td>services.impersonate.instruction</td>
<td>Remember that....</td>
</tr>
</tbody>
</table>
Customize the Shopping Cart

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>header.savedItem.button.label</td>
<td>Cart</td>
</tr>
<tr>
<td>2.</td>
<td>cart_view.cart_empty</td>
<td>Your cart is empty</td>
</tr>
<tr>
<td>3.</td>
<td>cart_view.cart.cart_items_header</td>
<td>Items in Your cart</td>
</tr>
<tr>
<td>4.</td>
<td>cart_view.cart.empty</td>
<td>Your cart is empty</td>
</tr>
<tr>
<td>5.</td>
<td>cart_view.cart.empty_prompt</td>
<td>To request an item ...</td>
</tr>
<tr>
<td>6.</td>
<td>cart_view.continue_shopping</td>
<td>Continue Shopping</td>
</tr>
</tbody>
</table>

Shopping Cart View
<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>cart_view.mouse_hover_header</td>
<td>Newest Item in your cart</td>
</tr>
<tr>
<td>2.</td>
<td>cart_view.items_total.several</td>
<td>* Items total</td>
</tr>
<tr>
<td>3.</td>
<td>cart_view.price_total</td>
<td>Total</td>
</tr>
<tr>
<td>4.</td>
<td>cart_view.view_cart</td>
<td>View Cart</td>
</tr>
<tr>
<td>5.</td>
<td>cart_view.checkout</td>
<td>Submit cart</td>
</tr>
<tr>
<td>6.</td>
<td>cart_view.edit_item_label</td>
<td>Edit</td>
</tr>
<tr>
<td>7.</td>
<td>cart_view.remove_item_label</td>
<td>Remove</td>
</tr>
<tr>
<td>8.</td>
<td>cart_view.item_quantity</td>
<td>Quantity</td>
</tr>
<tr>
<td>9.</td>
<td>quantity.cart.label</td>
<td>Total Items</td>
</tr>
<tr>
<td>10.</td>
<td>button.cart.label</td>
<td>Submit Cart</td>
</tr>
</tbody>
</table>
Customize the SRC settings

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>settings.inbox.title</td>
<td>Application</td>
</tr>
<tr>
<td>2</td>
<td>settings.service.title</td>
<td>Service Settings</td>
</tr>
<tr>
<td>3</td>
<td>settings.approvalDelegation/welcome.header</td>
<td>Approval Delegation</td>
</tr>
<tr>
<td>4</td>
<td>settings.welcome.header</td>
<td>Customize Your Experience</td>
</tr>
<tr>
<td>5</td>
<td>settings.welcome</td>
<td>You can customize ...</td>
</tr>
<tr>
<td>6</td>
<td>settings.instruction</td>
<td>Let the application guide ...</td>
</tr>
</tbody>
</table>

You can customize your catalog experience when you request services and support. The left navigation shows what your profile will let you change.

Let the application guide you through changing default settings to a personalized version that focuses on what is important to you.
Customize the Approval Delegation Details page

1. Approval Delegation
   - If you are out of the office, you can still keep everything on schedule by delegating your approval tasks to another person. You can choose who will take over your approval duties and when they will approve on your behalf.

2. Scheduled Approval Delegations
   - You have no delegated approvals scheduled.

3. New Approval Delegation
   - Please specify the following information to designate an Approver, and then click Schedule to save your choices.

4. Start Date
   - 10/29/2014

5. End Date
   - 10/30/2014

6. Choose Type
   - Service

7. Delegate To
   - [Dropdown]

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>settings.approvalDelegation.welcome.header</td>
<td>Approval Delegation</td>
</tr>
<tr>
<td>2.</td>
<td>settings.approvalDelegation.welcome</td>
<td>If you are out of office ...</td>
</tr>
<tr>
<td>3.</td>
<td>settings.approvalDelegation.list.header</td>
<td>Scheduled Approval ...</td>
</tr>
<tr>
<td>4.</td>
<td>settings.approvalDelegation.list.no.delegation</td>
<td>You have no ...</td>
</tr>
<tr>
<td>5.</td>
<td>settings.approvalDelegation.create</td>
<td>New Approval Delegation</td>
</tr>
<tr>
<td>6.</td>
<td>settings.approvalDelegation.form.description</td>
<td>Please specify the ...</td>
</tr>
<tr>
<td></td>
<td>settings.approvalDelegation.startDate.label</td>
<td>Start Date</td>
</tr>
<tr>
<td>---</td>
<td>------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>8</td>
<td>settings.approvalDelegation.startTime.label</td>
<td>Start Time</td>
</tr>
<tr>
<td>9</td>
<td>settings.approvalDelegation.endDate.label</td>
<td>End Date</td>
</tr>
<tr>
<td>10</td>
<td>settings.approvalDelegation.endTime.label</td>
<td>End Time</td>
</tr>
<tr>
<td>11</td>
<td>settings.approvalDelegation.chooseType</td>
<td>Choose Type</td>
</tr>
<tr>
<td>12</td>
<td>settings.approvalDelegation.delegateTo.label</td>
<td>Delegate To</td>
</tr>
<tr>
<td>13</td>
<td>settings.approvalDelegation.createButton.label</td>
<td>Schedule</td>
</tr>
</tbody>
</table>
## Customize the Recurring Cost labels

<table>
<thead>
<tr>
<th>Request Type</th>
<th>Recurring Period</th>
<th>Service</th>
<th>Fixed Cost</th>
<th>Additional Fees</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>All</td>
<td>Business Call from Home</td>
<td>$0.00</td>
<td>Yearly Fee: $8.00</td>
</tr>
<tr>
<td>Regular</td>
<td>Hourly</td>
<td>Basic PC Package</td>
<td>$3.00</td>
<td>Weekly Fee: $3.00</td>
</tr>
<tr>
<td>Info-Only</td>
<td>Daily</td>
<td>Budgeting</td>
<td>$0.00</td>
<td>Monthly Fee: $4.00</td>
</tr>
<tr>
<td>Fast Track</td>
<td>Weekly</td>
<td>Audio Conferencing</td>
<td>$0.00</td>
<td>Daily Fee: $2.00</td>
</tr>
<tr>
<td></td>
<td>Monthly</td>
<td>Accounts Payable</td>
<td>$0.00</td>
<td>Hourly Fee: $1.00</td>
</tr>
<tr>
<td></td>
<td>Yearly</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Non Recurring</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*See all Application Support cate...*  
*See all Telecommunications cate...*  
*See all Hardware Bundles category items*  
*See all Finance and Accounting ca...*  
*See all Conferencing and Collabor...*  
*See all Finance and Accounting co...*
### Customize the login panel

#### Id | Key | Value
--- | --- | ---
1. | itemList.recurringPrice.yearly.sm | + Yearly Fee
2. | itemList.recurringPrice.weekly.sm | + Weekly Fee
3. | itemList.recurringPrice.monthly.sm | + Monthly Fee
4. | itemList.recurringPrice.daily.sm | + Daily Fee
5. | itemList.recurringPrice.hourly.sm | + Hourly Fee

---

#### Id | Key | Value
--- | --- | ---
1. | login.label.username | User Name
2. | login.label.password | Password
3. | login.label.remember | Keep me signed in.
4. login.label.rememberNote  Do not select if …
5. login.label.submit       Sign in
6. login.label.reset       Clear

Customize the user account settings

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>header.user_account.approvals</td>
<td>Approvals</td>
</tr>
<tr>
<td>2.</td>
<td>header.user_account.requests</td>
<td>Request History</td>
</tr>
<tr>
<td>3.</td>
<td>header.user_account.settings</td>
<td>Settings</td>
</tr>
</tbody>
</table>
Customize the Request Summary panel

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>quantity.request.label</td>
<td>Quantity</td>
</tr>
<tr>
<td>2</td>
<td>button.add.cart.label</td>
<td>Add to Cart</td>
</tr>
<tr>
<td>3</td>
<td>button.requestNow.label</td>
<td>Request Now</td>
</tr>
<tr>
<td>4</td>
<td>itemdetails.cost.fixed</td>
<td>Fixed Cost</td>
</tr>
<tr>
<td>5</td>
<td>button.return.to.detail</td>
<td>Return to Service Detail</td>
</tr>
</tbody>
</table>

Customize the Submission panel

1. Total Items: 2
2. Submit Cart
Server customization reference

The following user interface labels can be customized in SRC Server customization file, which by default, you can find in the following location:

/resources/server/xx/CustomServer_xx.properties

Note: In this example path, the xx placeholder represents the language code of your SRC installation.

Customize the Search Results

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>quantity.cart.label</td>
<td>Total Items</td>
</tr>
<tr>
<td>2.</td>
<td>button.checkout.label</td>
<td>Submit Cart</td>
</tr>
<tr>
<td>1.</td>
<td>Configure in client custom.properties file</td>
<td>Catalog</td>
</tr>
<tr>
<td>2.</td>
<td>Configure in client custom.properties file</td>
<td>Service</td>
</tr>
<tr>
<td>3.</td>
<td>Configure in client custom.properties file</td>
<td>Support</td>
</tr>
<tr>
<td>4.</td>
<td>search_facet.service.item_type</td>
<td>Item Type</td>
</tr>
</tbody>
</table>
5. search_facet.facet_value_all | All
6. search_facet.service.item_type.item | Individual Items
7. search_facet.service.item_type.bundle | Bundles

8. Request Type
   - All
   - Regular
   - Info-Only
   - Fast Track

9. System Monitoring
   - Fixed Cost $0.00
   - Provide ongoing monitoring services for a server. Current monitoring capabilities include metrics such as availability load/performance.

10. Recurring Period
    - All
    - Hourly
    - Daily
    - Weekly
    - Monthly
    - Yearly
    - Non Recurring

11. Streaming Media
    - Fixed Cost $0.00
    - Provides the ability to publish streaming media files accessible through the internal network.

12. Technical Project Management
    - Consulting services to assist you with management of an application roll-out project.

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>search_facet.service.item_status</td>
<td>Request Type</td>
</tr>
<tr>
<td>9</td>
<td>search_facet.facet_value_all</td>
<td>All</td>
</tr>
<tr>
<td>10</td>
<td>search_facet.service.item_status.standard</td>
<td>Regular</td>
</tr>
<tr>
<td>11</td>
<td>search_facet.service.item_status.infoOnly</td>
<td>Info-Only</td>
</tr>
<tr>
<td>12</td>
<td>search_facet.service.item_status.nonCart</td>
<td>Fast Track</td>
</tr>
<tr>
<td>13</td>
<td>search_facet.service.recurring_period</td>
<td>Recurring Period</td>
</tr>
<tr>
<td>14</td>
<td>search_facet.facet_value_all</td>
<td>All</td>
</tr>
<tr>
<td>15</td>
<td>search_facet.service.recurring_period.hour</td>
<td>Hourly</td>
</tr>
<tr>
<td>16</td>
<td>search_facet.service.recurring_period.day</td>
<td>Daily</td>
</tr>
<tr>
<td>Id</td>
<td>Key</td>
<td>Value</td>
</tr>
<tr>
<td>----</td>
<td>-----------------------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>1</td>
<td>group_info.order_info.label</td>
<td>Order Information</td>
</tr>
<tr>
<td>2</td>
<td>order_info.requested_for.label</td>
<td>This request is for</td>
</tr>
<tr>
<td>3</td>
<td>order_info.delivery_sla.label</td>
<td>Delivery Targets</td>
</tr>
</tbody>
</table>

Customize the order information

Name Change

```
1. Order Information
2. * This request is for
   FALCON, JENNIFER
3. * Delivery Targets
   6 Business Hours
4. * Service Level Package
   Bronze Service
```

Item Options

```
Old Name
New Name
```
### Customize the Subscriptions page

#### Your Subscriptions

Track and modify your subscriptions here.

#### Subscriptions

1. **Personal**
2. **Departmental**
3. **Subscription Status**
   - **Requested**
   - **Active**
   - **Cancellation Requested**
   - **Canceled**
   - **Denied**

#### Personal - All

<table>
<thead>
<tr>
<th>Name</th>
<th>Subscriber</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-mail / Webmail (North America)</td>
<td>advantage/North America - HR &amp; Administration</td>
<td>10/28/2014</td>
</tr>
<tr>
<td>Printing (North America)</td>
<td>advantage/North America - HR &amp; Administration</td>
<td>10/28/2014</td>
</tr>
<tr>
<td>Internet / Internet (North America)</td>
<td>advantage/North America - HR &amp; Administration</td>
<td>10/28/2014</td>
</tr>
<tr>
<td>MyDevices</td>
<td>advantage/North America - HR &amp; Administration</td>
<td>10/28/2014</td>
</tr>
<tr>
<td>Handheld, PDA &amp; Telephony</td>
<td>advantage/North America - HR &amp; Administration</td>
<td>10/28/2014</td>
</tr>
<tr>
<td>Applications</td>
<td>advantage/North America - HR &amp; Administration</td>
<td>10/28/2014</td>
</tr>
</tbody>
</table>

#### Id | Key                                                                 | Value                        |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>subscription.inbox.group.displayname</td>
<td>Subscriptions</td>
</tr>
<tr>
<td>2</td>
<td>subscription.inbox.group.myuse.displayname</td>
<td>Personal</td>
</tr>
<tr>
<td>3</td>
<td>subscription.inbox.group.department.displayname</td>
<td>Departmental</td>
</tr>
<tr>
<td>4</td>
<td>subscription.inbox.status.displayname</td>
<td>Subscription Status</td>
</tr>
<tr>
<td>5</td>
<td>subscription.status.all</td>
<td>All</td>
</tr>
<tr>
<td>6</td>
<td>Dynamic value</td>
<td>Personal - All</td>
</tr>
</tbody>
</table>
Customize the Knowledge Management results

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>search_facet.km.doctype</td>
<td>Doc Type</td>
</tr>
<tr>
<td>2</td>
<td>search_facet.km.category</td>
<td>Category</td>
</tr>
<tr>
<td>3</td>
<td>search_facet.facet_value_all</td>
<td>All</td>
</tr>
<tr>
<td>4</td>
<td>search_facet.facet_value_all</td>
<td>All</td>
</tr>
</tbody>
</table>
Customize Your Approvals

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>search_facet.approval.type</td>
<td>Approval</td>
</tr>
</tbody>
</table>

Id  Key                       Value
---  --------------------------  ------------
1.   search_facet_km.last_updated     Date
2.   search_facet_facet_value_all              All
3.   search_facet_km.last_updated_week       Past Week
4.   search_facet_km.last_updated_month     Past Month
5.   search_facet_km.last_updated_year      Past Year
<table>
<thead>
<tr>
<th></th>
<th>search_facet.facet_value_all</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.</td>
<td>search_facet.approval.type.service</td>
<td>Service</td>
</tr>
<tr>
<td>4.</td>
<td>search_facet.approval.type.change</td>
<td>Change</td>
</tr>
<tr>
<td>5.</td>
<td>search_facet.approval.type.quote</td>
<td>Request</td>
</tr>
</tbody>
</table>

**Note:** The Approvals Page also requires some customization on the SRC client side. For more information, see "Customize the Approvals page" on page 115
Customize the Request For panel

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>group_info.order_info_support.label</td>
<td>Support Recipient (Support catalog/General support)</td>
</tr>
<tr>
<td>1</td>
<td>group_info.order_info.label</td>
<td>Order Information (Service catalog)</td>
</tr>
<tr>
<td>2</td>
<td>order_info.requested_for.label</td>
<td>This request is for</td>
</tr>
</tbody>
</table>
SRC Service Manager customization reference

You must customize the following fields in Service Manager. To do this, edit the SRC checkout panels by navigating to Tailoring > SRC Tailoring in Service Manager. For more information on how to customize and configure the SRC checkout panels, see “Tailoring check-out panels” on page 23.

<table>
<thead>
<tr>
<th></th>
<th>support_attachments.support_attachments_label</th>
<th>Attachments (Support catalog/General support)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.</td>
<td>attachments_info.attachments_info_label</td>
<td>Attachments (Service catalog)</td>
</tr>
</tbody>
</table>
# Delivery Information

1. **Title**

2. **Description**

# Contact Information

1. **Contact Name**
   - FALCON, JENNIFER

2. **Contact Method**
   - E-mail

# Additional Information

1. **Urgency**
   - Select one

# Attachments

<table>
<thead>
<tr>
<th>Id</th>
<th>Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Delivery Information</td>
</tr>
<tr>
<td>2</td>
<td>Title</td>
</tr>
<tr>
<td>3</td>
<td>Description</td>
</tr>
<tr>
<td>4</td>
<td>Contact Information</td>
</tr>
<tr>
<td></td>
<td>Contact Name</td>
</tr>
<tr>
<td>---</td>
<td>----------------------</td>
</tr>
<tr>
<td>6</td>
<td>Contact Method</td>
</tr>
<tr>
<td>7</td>
<td>Additional Information</td>
</tr>
<tr>
<td>8</td>
<td>Urgency</td>
</tr>
</tbody>
</table>
Style Service Request Catalog only for the desktop

The following graphics and keys show the different labels of the SRC user interface that you can customize Service Request Catalog for a desktop only experience.

Customize colors and graphics

You can change the interface colors and introduce customization that echoes your company's signature branding and color palette.

Service Request Catalog packages files that you can edit to change the basic application design and color palette. You can also redesign the backgrounds for the header, main, and footer areas. You do not need to be a designer or a programmer to make these changes. Obtain the required tools and follow the directions in this section.

Before you begin

Customization of most of the graphics and images of Service Request Catalog is performed by modifying the symbols in the panel.fla file and replacing the panel.swf used in SRC. The symbols in the panel.fla file each correspond to a graphic, image, or text area in the SRC.

You can find the panel.swf file and the panel.fla in the following location:

< SRC_R oot_Folder > \secure\configuration\default

Note:

- < SRC_R oot_Folder > refers to the Web application server where you installed SRC.

- Before you attempt to replace the panel.swf file, we recommend that you make a copy and store it in a convenient location. You should perform all your adjustments on the copy first.

- You need to obtain Adobe® Flash® Professional CS5.x (or a later release) to edit the relevant files. You may also need an image editing application, such as Adobe® Illustrator® or Adobe® Photoshop®.

- Make sure you have your corporate branding color values (expressed in hexadecimal numbers) handy.
Modifying panel.swf

To modify panel.swf and customize Service Request Catalog, follow these steps:

**Note:** You cannot change the name of the panel.swf file. SRC will search only for the name panel.swf.

1. Open panel.fla in Flash Professional CS5.x, and note the various symbols. You should see symbols for the background, header, footer, logo, and other elements of the SRC user interface.

2. Select a symbol that you wish to modify.

3. By default, Flash Professional CS 5.x provides numerous panels on the right side of the screen. These panels allow you to access the properties of the symbol that you selected. You can choose to adjust the color, gradient of the various symbols. For example, you may click on the symbol for the background, and use the color panel to specify a background that matches your organization's branding.

4. When you have finished modifying the symbols, make sure that all symbols are under the "Illustrator Symbols" folder in the Library panel in Flash Professional. During the process of manipulating the symbols, Flash Professional may have moved the symbols from their original hierarchy in the Library Panel.

5. Export the file as a panel.swf file.

6. Stop any running instances of SRC.

7. Copy the panel.swf file that you exported in Step 4, and then navigate to where the current panel.swf is installed. By default, this folder in the following location:

   `<SRC_Root_Folder>\secure\configuration\default`

8. Rename the current panel.swf to panel.OLD to ensure that you maintain an original copy, and then paste your modified panel.swf into the same directory. Move the panel.OLD file to a safe location.

9. Restart Service Request Catalog. You should see the modifications you made to the SRC user interface.

**Note:** By manipulating the symbols in the panel.swf file, you can customize almost all of the
graphical aspects of SRC. This includes the your corporate logo, which you can adjust by replacing the current logo in panel.swf. However, you should be careful not to adjust the following:

- **Size**: Adjusting the size of the symbols can lead to unpredictable results and may cause a poor user experience. SRC is designed with most elements at specific sizes.
- **Nine scale slicing**: Manipulating Nine-scale slicing can cause some symbols to become distorted in unintended ways, so that the symbols may be too large or small, or distorted.
- **Names**: Do not change the names of any of the symbols or their corresponding associations.

### Handling empty symbols

Some symbols in panel.swf may be blank or empty by default. For example, the backgroundImage symbol contains no image by default. However, you may wish to add a picture to that will appear as the backdrop to your users’ SRC experience.

Similarly, you may wish to remove some symbols. To do this, simply reset the symbol with a blank graphic.

**Caution**: Do not delete any symbol.

### Modifying the favorite icon

The favorite icon is displayed in the browser tab header. By default, this icon is the HP logo. However, you can replace this logo with your organization's logo by replacing the favicon.ico file located in the SRC root folder. Note that the size for this icon must be 16 x 16 pixels.

### Customize the text format

You may want to make changes to the appearance of text in the application. You can control fonts, text color, and size in the header and footer areas by customizing the style.css file.

### Text manipulation overview

Text styles, font, weight, and other characteristics are controlled by the styles.swf file. You can locate the styles.swf file at the following location:

<SRC_Root_Folder>\src\secure\configuration\default
However, the styles.swf file is an auto-generated .swf file derived from the corresponding styles.css file. Cascading Style Sheets are a well-known standard you can use easily configure attractive looking text according to your organization's branding guidelines. To convert the styles.css file to a styles.swf file, you must used the Adobe Flex Software Development Kit (SDK) compiler. You can download this open-source compiler from the following Web site: www.Adobe.com

**Converting styles.css to styles.swf**

Once you have downloaded the compiler, you can use it to convert the styles.css file to a styles.swf file. To do this, follow these steps:

1. Extract the compiler to an easily accessible folder.
2. Add the location of the mxml compiler to the Path environment variable.
3. Open a command prompt, and then navigate to the location where you stored the styles.css.
4. Enter the following command at the command prompt:

   mxmlc styles.css

   **Note:** If SRC end users do not have the access to the Internet, enter the following command at the command prompt:

   mxmlc styles.css -static-link-runtime-shared-libraries=true

5. When you do this, the compiler should generate the styles.swf file. Copy the styles.swf file, and replace the original styles.swf in the following location:

   <SRC_Root_Folder>/src/secure/configuration/default

   **Note:** Make sure that you make a copy of the original styles.swf file and store it in a safe place.

**Customize the font family**

The out-of-box font style in Service Request Catalog 9.40 is HP Simplified, but you may want to change your current font style to Service Request Catalog 1.40. The following configuration files enable you to make the change:
The out-of-box .css file is only for your reference.

Demonstrates the steps that are required to change the HP Simplified font style to Service Request Catalog 1.40.

The font folder

Includes all the necessary font files.

A .swf file which is converted from the style(modified).css file

There are two circumstances in which you change the current font style to Service Request Catalog 1.40. Therefore, before you make the change, you must decide which of the following circumstances applies to you:

- If your style.css and style.swf are out-of-box files, you only need to rename style (modified).swf as style.swf, and then copy it to replace your old style.swf under the following path:

  `<SRC_Root_Folder>\src\secure\configuration\default`

- If your style.css and style.swf are not out-of-box files, you need to integrate the style (modified).css file into the style.css file. For more information, refer to "Integrate style (modified).css into style.css" below.

**Integrate style(modified).css into style.css**

The style(modified).css file demonstrates all the necessary steps to customize the font style in Service Request Catalog 1.40. Since you have modified your style.css before, you should integrate the style(modified).css file to your style.css file to keep all your previous modifications. To do this, follow these steps:

1. Open the following file in a text editor.

   `<SRC_Root_Folder>\src\secure\configuration\default\style.css`

2. Insert the following lines at the start of the file.
@namespace s "library://ns.adobe.com/flex/spark"
namespace componentsCommon
"com.hp.service.catalog.client.components.common.*";
@namespace lookup "com.hp.siberia.component.formInput.lookup.*";
namespace renderer
"com.hp.siberia.skin.component.renderer.formInput.graphic.*"

@font-face {
  fontFamily: "Collator";
  src: url("font/Collator.otf");
  embedAsCFF: true;
}

@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-Book.otf");
  embedAsCFF: true;
}

@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-BookItalic.otf");
  fontStyle: italic;
  embedAsCFF: true;
}

@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-Medium.otf");
  fontWeight: bold;
  embedAsCFF: true;
}

@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-MediumItalic.otf");
  fontWeight: bold;
  fontStyle: italic;
  embedAsCFF: true;
}

3. Optional. If you no longer need the HP Simplified font style, remove all the @font-face entries that contain HPSimplified.

   **Note:** This step is not demonstrated in the style(modified).css file.

4. Change the font-family of existing selectors to Collator or Gotham to align with the original style of Service Request Catalog 1.40.
5. Insert the following lines at the end of the file.

```
s|ButtonBase.small, s|ButtonBase.basic,
s|ButtonBase.gelBlue, s|ButtonBase.gelGreen, s|ButtonBase.gelOrange,
s|ButtonBase.actionButton,
s|ButtonBase.widgetButton,s|ButtonBase.unsubscribeBtn {
  font-family: "Gotham";
}
s|TextInput,
s|TextInput.graphic, lookup|Lookup #input,
s|TextArea.graphic,
s|NumericStepper.graphic,
s|DropDownList.graphic,
s|ToggleButtonBase.graphic,
s|NumericStepper.graphic,
s|CheckBox.graphic,
s|RadioButton.graphic,
renderer|GraphicBackgroundItemRenderer,
componentsCommon|TimeSpinner.graphic,
s|CheckBox.optional {
  font-family: "Gotham";
}

.attachmentLink, .attachmentLinkHover,
.attachmentButton|ButtonTextWhite,
.pageRangesPaginatorButton, .pageRangesPaginatorSkin,
.pageRangesPaginatorPageRangeContainer, .pageRangeNormal {
  font-family: "Gotham";
}

s|TextBase.h1, s|RichEditableText.h1, s|ButtonBase.h1 {
  font-family: "Collator";
}
s|TextBase.h2, s|RichEditableText.h2, s|ButtonBase.h2 {
  font-family: "Collator";
}
s|TextBase.h3, s|RichEditableText.h3, s|ButtonBase.h3 {
  font-family: "Gotham";
}
s|TextBase.h4, s|RichEditableText.h4, s|ButtonBase.h4 {
  font-family: "Gotham";
}
s|TextBase.h5, s|RichEditableText.h5, s|ButtonBase.h5 {
  font-family: "Gotham";
}
s|TextBase.h6, s|RichEditableText.h6, s|ButtonBase.h6 {
  font-family: "Gotham";
}
```
6. Save your modifications, and then convert your modified style.css file into a style.swf file. For more information, see the "Converting styles.css into styles.swf" section in "Customize the text format" on page 146.
7. Replace the original style.swf file with the newly converted one under the following path:

```
<SRC_Root_Folder>/src /secure/configuration/default
```

8. Restart Service Request Catalog. You should see the modifications you made to the Service Request Catalog user interface.

**Note:** If the fonts have not changed, clear your browser cache and then restart Service Request Catalog again.

## The style.css file

The content below is provided as a convenient reference for the default style.css file that is included with SRC.

**Note:** This file is only used for customizing the .swf files for Service Request Catalog on the desktop.

```css
/* CSS file */

@font-face {
    fontFamily: HPSimplified;
    src: url("font/HPSimplified_Rg.ttf");
    embedAsCFF: true;
}

@font-face {
    fontFamily: HPSimplified;
    src: url("font/HPSimplified_It.ttf");
    fontStyle: italic;
    embedAsCFF: true;
}

@font-face {
    fontFamily: HPSimplified;
    src: url("font/HPSimplified.ttf");
    fontWeight: bold;
    embedAsCFF: true;
}

@font-face {
    fontFamily: HPSimplified;
    src: url("font/HPSimplified_BdIt.ttf");
    fontWeight: bold;
    fontStyle: italic;
    embedAsCFF: true;
}
```
/*
text shape
*/

.headerTitle {
  font-size: 21;
  color: #333333;
  font-family: HPSimplified;
}

.headerCartLink{
  font-size: 14;
  color: #333333;
  textDecoration: "underline";
  font-family: HPSimplified;
}

.headerUserDropdownLabel{
  font-size: 14;
  color: #333333;
  textDecoration: "underline";
  font-family: HPSimplified;
}

.launcherTitle{
  font-size: 24;
  color: #435057;
  font-family: HPSimplified;
}

.launcherSubtitle{
  font-size: 11;
  color: #71828C;
  font-family: HPSimplified;
}

.bodyTitle{
  font-size: 24;
  color: #0096d6;
  font-family: HPSimplified;
}

.bodySubtitle{
  font-size: 12;
  color: #435057;
  font-family: HPSimplified;
}
The style(modified).css file

The content below is provided as a convenient reference for the default style(modified).css file that is included with SRC.

Note: This file is only used for customizing the .swf files for Service Request Catalog on the desktop.

/* CSS file */
@namespace s "library://ns.adobe.com/flex/spark";
@namespace componentsCommon "com.hp.service.catalog.client.components.common.*";
@namespace lookup "com.hp.siberia.component.formInput.lookup.*";
@namespace renderer "com.hp.siberia.skin.component.renderer.formInput.graphic.*";

/*
  embed font files
*/
@font-face {
  fontFamily: "Collator";
  src: url("font/Collator.otf");
  embedAsCFF: true;
}
@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-Book.otf");
  embedAsCFF: true;
}
@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-BookItalic.otf");
  fontStyle: italic;
  embedAsCFF: true;
}
@font-face {
  fontFamily: "Gotham";
  src: url("font/Gotham-BookItalic.otf");
  fontStyle: italic;
  embedAsCFF: true;
}
fontFamily: "Gotham";
src: url("font/Gotham-Medium.otf");
fontWeight: bold;
embedAsCFF: true;
}

@font-face {
fontFamily: "Gotham";
src: url("font/Gotham-MediumItalic.otf");
fontWeight: bold;
fontStyle: italic;
embedAsCFF: true;
}

/*
You can remove the four @font-face below
if you no long need "HPSimplified"
*/

@font-face {
fontFamily: "HPSimplified";
src: url("font/HPSimplified_Rg.ttf");
embedAsCFF: true;
}

@font-face {
fontFamily: "HPSimplified";
src: url("font/HPSimplified_It.ttf");
fontStyle: italic;
embedAsCFF: true;
}

@font-face {
fontFamily: "HPSimplified";
src: url("font/HPSimplified.ttf");
fontWeight: bold;
embedAsCFF: true;
}

@font-face {
fontFamily: "HPSimplified";
src: url("font/HPSimplified_BdIt.ttf");
fontWeight: bold;
fontStyle: italic;
embedAsCFF: true;
}

/*
text shape
*/

.headerTitle {
font-family: "Collator";
font-size: 21px;
font-size: 11px;
color: #666666;
}

.navigationLabelUp{
font-family: "Gotham";
font-size: 17px;
font-weight: normal;
color: #666666;
text-drop-shadow-alpha: 0.25;
}

.navigationLabelSelected{
font-family: "Gotham";
font-size: 17px;
font-weight: normal;
color: #ffffff;
text-drop-shadow-alpha: 0.25;
}

.navigationLabelOver{
font-family: "Gotham";
font-size: 17px;
font-weight: normal;
color: #000000;
text-drop-shadow-alpha: 0.25;
}

.navigationLabelDisabled{
font-family: "Gotham";
font-size: 17px;
font-weight: normal;
color: #CCCCCC;
text-drop-shadow-alpha: 0.25;
}

/*
override built-in font styles
*/

s|ButtonBase.small, s|ButtonBase.basic,
s|ButtonBase.gelBlue, s|ButtonBase.gelGreen, s|ButtonBase.gelOrange,
s|ButtonBase.actionButton, s|ButtonBase.widgetButton,s|ButtonBase.unsubscribeBtn {
font-family: "Gotham";
}

s|TextInput,
s|TextInput.graphic,
lookup|Lookup #input,
s|TextArea.graphic,
s|NumericStepper.graphic,
s|DropDownList.graphic,
s|ToggleButtonBase.graphic,
Style Service Request Catalog for tablets and the desktop

The following graphics and keys show the different labels of the SRC user interface that you can customize for a tablet or desktop experience.

Customize the background, images, and icons

To modify a background image or icon, follow these steps:

1. Unzip the image.zip file from the Service Request Catalog .war file to some location on your computer.

2. Replace the image of your choice with your customized image using the same file name.
3. Re-zip the `image.zip` file, and overwrite the version embedded in the Service Request Catalog `war` file.

Reference: Background, images and icons

1. `header.png`
2. `body.png`
3. `footer.png`
4. `backgroundImage.png`
5. `companyLogo.png`
6. `headerShoppingCartIcon.png`
Customize the font styles

To modify a font style, follow these steps:

1. Locate the selector you want to customize in the font.css file.
2. Modify the CSS properties for color, size, style, and so on.
3. Save the font.css file.
Font styles
13. footerLegalNoticeLabel

14. navigationLabelUp

    navigationLabelSelected

    navigationLabelOver (Added in Service Request Catalog 9.33 for compatibility purposes. It is used only for the desktop.)

15. backLabel

The font.css file

The content below is provided as a convenient reference for the default font.css file that is included with SRC.

Note: This file is only used for customizing the .zip files method for Service Request Catalog for tablets and on the desktop.

/* CSS file */
/* For Example <Start>*/
.text {
    color: #FF0000;
    font-size: 18;
    font-weight: bold;
    font-style: italic;
    text-decoration: underline;
}

font-family OOB default: HPSimplified
font-family list for IOS:
    Heiti SC;
    Helvetica Neue;
font-family list for Android:
    DroidSans;

<End>*/

.headerTitle {
    font-size: 21;
    color: #333333;
    font-family: HPSimplified;
}

.headerCartLink{
    font-size: 14;
Customization Guide
Appendix D: Styling the interface

```css
.color: #333333;
text-decoration: underline;
font-family: HPSimplified;
}

.headerUserDropdownLabel{
    font-size: 14;
    color: #333333;
    text-decoration: underline;
    font-family: HPSimplified;
}

.launcherTitle{
    font-size: 24;
    color: #435057;
    font-family: HPSimplified;
}

.launcherSubtitle{
    font-size: 11;
    color: #71828C;
    font-family: HPSimplified;
}

.bodyTitle{
    font-size: 24;
    color: #0096d6;
    font-family: HPSimplified;
}

.bodySubtitle{
    font-size: 12;
    color: #435057;
    font-family: HPSimplified;
}

.backLabel{
    font-size: 14;
    color: #0096d6;
    font-family: HPSimplified;
}

.footerTitle{
    font-size: 18;
    color: #333333;
    font-family: HPSimplified;
}

.footerSubtitle{
```
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 Customization Guide (Service Request Catalog 9.40)

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 ovdoc-itsm@hp.com.

We appreciate your feedback!