



# Systemet

Software Version: 10.04

Windows and Linux Operating System

# Customization Editor User Guide

Document Release Date: July 2017

Software Release Date: July 2017



**Hewlett Packard**  
Enterprise

## Legal Notices

### Warranty

The only warranties for Hewlett Packard Enterprise 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. Hewlett Packard Enterprise 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 Hewlett Packard Enterprise 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

© 2003-2017 Hewlett Packard Enterprise Development LP

### Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

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

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

## Documentation Updates

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

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** 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 HPE sales representative for details.

## Support

Visit the HPE Software Support site at: <https://softwaresupport.hpe.com/>.

Most of the support areas require that you register as an HP Passport user and to sign in. Many also require a support contract. To register for an HP Passport ID, click **Register** on the HPE Support site or click **Create an Account** on the HP Passport login page.

To find more information about access levels, go to: <https://softwaresupport.hpe.com/web/softwaresupport/access-levels>.

**HPE Software Solutions Now** accesses the HPSW Solution and Integration Portal website. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this website is <https://softwaresupport.hpe.com/km/KM01702731>.

## About this PDF Version of Online Help

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

# Contents

Chapter 1: Customization Editor Guide .....	6
Chapter 2: Customization Editor .....	7
Overview .....	7
User Interface .....	8
Extension Explorer .....	9
Extension Name .....	9
Artifact Types .....	9
SDM Details .....	10
Property Descriptors .....	10
Property Groups .....	11
Menu Options .....	12
Server Explorer .....	13
SDM Elements .....	14
Artifacts .....	14
Properties .....	14
Property Groups .....	16
Taxonomies .....	16
Chapter 3: Getting Started .....	17
Creating an Extension Project .....	17
Sharing an Extension Project .....	19
Referencing Other HPE Systinet Workbench Projects .....	20
Saving Modifications .....	20
Searching the Extension .....	20
Chapter 4: Manipulating Artifact Types .....	22
Creating an Artifact Type or Package .....	22
Modifying the Attributes of an Artifact Type .....	23
Modifying the Properties of an Artifact Type .....	23
Mapping an Artifact Type to a Registry .....	25
Adding a Registry Mapping for an Artifact Type .....	26
Adding a Registry Mapping for a Property .....	26

Example: Adding the Department Property to Services .....	27
<b>Chapter 5: Manipulating Properties .....</b>	<b>28</b>
Creating a Property .....	28
Modifying a Property .....	29
Creating a Property Group .....	29
Modifying a Property Group .....	30
<b>Chapter 6: Creating and Using Components .....</b>	<b>31</b>
Creating a Component .....	31
Creating a Task .....	32
Developing a Component .....	32
<b>Chapter 7: Modifying Taxonomies .....</b>	<b>33</b>
<b>Chapter 8: Deploying Extensions .....</b>	<b>35</b>
Exporting the Extension Project .....	35
Redeploying the EAR File .....	36
Deploying Components to HPE Systinet .....	36
Deploying Extensions Remotely .....	37
<b>Chapter 9: Views .....</b>	<b>38</b>
Artifact Editor .....	38
Artifact Editor: Overview Tab .....	39
Artifact Editor: Properties Tab .....	39
Artifact Editor: Registry Mapping Tab .....	41
Messages View .....	42
Project Editor .....	42
Project Editor: Overview Tab .....	42
Project Editor: Environment Tab .....	43
Project Editor: Dependencies Tab .....	43
Property Editor .....	44
Property Editor: Overview Tab .....	44
Property Editor: Advanced Tab .....	45
Property Group Editor .....	46
Search View .....	47
Tasks Editor .....	47
Taxonomies Editor .....	48
<b>Appendix A: Dialog Boxes .....</b>	<b>49</b>

New Artifact .....	49
New Artifact: Create .....	49
New Artifact: Database Settings .....	50
New Component .....	50
New Extension Project .....	51
New Extension Project: Specify HPE Systinet Server Installation .....	51
New Extension Project: New Server .....	52
New Project: Select Working Extensions .....	52
New Project: Create a New Extension .....	53
New Project: Create Dependencies .....	53
New Property .....	54
New Property: Primitive .....	54
New Property: Relationship .....	54
New Property: Taxonomy .....	55
New Property Group .....	56
New Task .....	56
New Taxonomy .....	56
Registry Mapping .....	57
Registry Mapping: New Artifact Mapping .....	57
Registry Mapping: New Property Mapping .....	57
Search .....	58
Share Project .....	58
Appendix B: Localname Naming Rules .....	59

# Chapter 1: Customization Editor Guide

Welcome to the *Customization Editor User Guide*. This guide explains how to use Customization Editor as part of HPE Systinet.

This guide contains the following chapters:

- ["Overview" on page 7](#)  
Provides an overview of the main features of Customization Editor.
- ["Getting Started " on page 17](#)  
Describes the installation of the main features, and shows you how to create an extension in Customization Editor.
- ["Manipulating Artifact Types " on page 22](#)  
Explains how to create, modify and delete artifacts and packages using Customization Editor.
- ["Creating and Using Components " on page 31](#)  
Shows how to create and use custom components in Customization Editor.
- ["Modifying Taxonomies " on page 33](#)  
Shows how to build and modify taxonomies for an extension project in Customization Editor.
- ["Deploying Extensions " on page 35](#)  
Instructions on how to deploy extensions created in Customization Editor.
- ["Views " on page 38](#)  
Explanation of each view in Customization Editor.
- ["Dialog Boxes " on page 49](#)  
Dialog boxes reference.
- ["Localname Naming Rules " on page 59](#)  
Naming strategies for database tables and columns.

# Chapter 2: Customization Editor

HPE Systinet Customization Editor is a set of HPE Systinet Workbench features that enable you to edit the underlying System Data Model (SDM) configuration of an installation of HPE Systinet.

With the introduction of HPE Systinet 1.00, Web GUI customization was decoupled from SDM customization. Having applied an SDM extension, HPE Systinet administrators must configure the UI visibility of artifacts/properties in the web GUI using the **Administration > Customize** menu. This is no longer part of the HPE Systinet Workbench tool set.

This chapter introduces Customization Editor in the following sections:

- ["Overview" below](#)
- ["User Interface " on the next page](#)
- ["SDM Elements " on page 14](#)

## Overview

HPE Systinet is distributed with a preconfigured data model. For typical deployment scenarios, the model should be customized to fit an organization-specific environment.

HPE Systinet Customization Editor provides a mechanism to customize the model, implement those changes, and keep them isolated in a so-called extension package that can then be re-applied during a re-installation or upgrade of HPE Systinet.

To customize HPE Systinet, follow this process:

1. **Create an extension project.**

For details, see ["Creating an Extension Project " on page 17](#).

2. **Modify the data model.**

For details, see the following:

- ["Manipulating Artifact Types " on page 22](#)
- ["Modifying the Properties of an Artifact Type " on page 23](#)
- ["Creating and Using Components " on page 31](#)

- ["Modifying Taxonomies " on page 33](#)

### 3. Deploy the customization to HPE Systinet.

For details, see ["Deploying Extensions "](#).

Be very careful when modifying artifacts that already have instances in the repository. For example, a new installation already contains the person artifact *administrator*. Removing properties from an artifact type with instances, or adding a new required property without a default value, may cause instability in HPE Systinet. Customization Editor should prevent this if the HPE Systinet server is running during modification.

## User Interface

The default **Extension** perspective is split into a number of sections with menu options across the top.

The perspective consists of the following elements:

- **Project Explorer**

The tree view of the SDM configuration represented by your project.

For details, see ["Extension Explorer " on the next page.](#)

- **Server Explorer**

The view listing HPE Systinet server connections to HPE Systinet Workbench.

For details see, ["Server Explorer " on page 13.](#)

- **Editor Views**

The main area of the perspective contains editor views of project elements. You can open multiple editors. The functionality of each editor varies depending on the project element.

For details, see ["Views " on page 38.](#)

- **Information Views**

Messages and search results open their own views.

For details, see ["Views " on page 38.](#)

- **Menu Items**

The standard Eclipse menu options plus additional HPE Systinet Workbench options.

For details about the additional options for Customization Editor, see ["Menu Options " on page 12.](#)



# Extension Explorer

The Extension Explorer, as shown in “Extension Explorer View”, represents a tree view of the extension project that you are currently working with. It includes all the elements from extensions that it depends on as well.

The tree is split into the extension root and five branches representing different elements of your extension.

Double-click an item in the tree to open the relevant editor, or to open or close a branch.

For details about each editor, see ["Views " on page 38](#).

Right-click an element to open a context menu that contains options for the element.

Each element and its contextual functionality is described in the following sections:

- ["Extension Name " below](#)
- ["Artifact Types " below](#)
- ["SDM Details " on the next page](#)
- ["User Interface " on the previous page](#)
- ["Taxonomies " on page 16](#)
- ["Project Editor " on page 42](#)

## Extension Name

The extension name is the root element of the project.

The context menu contains the option to **Build Extension**, which creates a jar file that contains your extension. This extension is ready to import to HPE Systinet.

For deployment details, see ["Deploying Extensions " on page 35](#).

## Artifact Types

The Artifact Types branch contains the artifact types organized into a hierarchical structure reflecting the System Data Model (SDM). Each element within **Artifact Types** is a type of artifact or artifact package.

**Caution:** HPE Systinet separates the SDM into two models, public and system. Customization Editor hides system artifacts, preventing you from modifying them.

**Tip:** An artifact package is an artifact type that serves as a group of other artifacts. They do not have instances in the repository and just serve to organize a hierarchy of artifact types in the SDM.

For details, see ["Artifacts " on page 14](#).

The context menu for the Artifact Types branch contains the following functionality:

- **New Artifact Type or Package**

Creates an artifact type as described in ["Creating an Artifact Type or Package " on page 22](#).

- **Open Artifact Type or Package**

Opens the artifact editor for the selected element with the name of the element as the editor title.

For details, see ["Artifact Editor " on page 38](#).

- **Deprecate Artifact Type or Package**

Deactivates the artifact type in the extension project and hides it in the HPE Systinet UI.

- **Undeprecate Artifact Type or Package**

Reactivates a deprecated artifact type in the extension project.

- **Delete Artifact Type or Package**

Removes the artifact type from the extension project.

## SDM Details

The SDM Details branch contains two element types described in the following sections:

- ["Property Descriptors " below](#)
- ["Property Groups " on page 16](#)

HPE Systinet separates the SDM into two models, public and system. Customization Editor shows properties and property groups used by system artifacts, but prevents you from modifying them.

## Property Descriptors

Property descriptors are the attributes of artifact types.

For details, see ["Properties " on page 14.](#)

In the Extension Explorer, properties are divided into the following branches:

- Primitive Property Descriptors
- Relationship Property Descriptors
- Taxonomy Property Descriptors

Each property type branch contains one context menu option: New Property.

For details, see ["Creating a Property " on page 28.](#)

Each property type branch contains a list of the properties of that type.

The context menu for properties contains the following options:

- **Open Property**

Opens the property editor for the selected element with the name of the element as the editor title.

For details, see ["Property Editor " on page 44.](#)

- **Deprecate Property**

Deactivates the property in the extension project and hides it in the HPE Systinet UI.

- **Undeprecate Property**

Reactivates the deprecated property in the extension project.

- **Delete Property**

Removes the property from the extension project.

- **Find Usage**

Opens a search view displaying all the artifacts and property groups containing this property.

For details, see ["Search View " on page 47.](#)

## Property Groups

Property groups are sets of properties that can be added collectively to an artifact type.

For details, see ["Property Groups " on page 16.](#)

The context menu of the Property Group branch contains one option: New Property Group.

For details, see ["Creating a Property Group " on page 29.](#)

The context menu for each property group contains the following options:

- **Open Property Group**

Opens the property group editor for the selected element with the name of the element as the editor title.

For details, see ["Property Group Editor" on page 46](#).

- **Delete Property Group**

Removes the property group from the extension project.

- **Find Usage**

Opens a search view displaying all the artifacts that contain this property group.

For details, see ["Search View " on page 47](#).

## Menu Options

The menus in HPE Systinet Workbench are standard Eclipse menus, with the addition of specific functionality for HPE Systinet Workbench editors.

The additional menu options for Customization Editor include:

- **File > New > HPE Extension Project**

Create a new project, as described in ["Creating an Extension Project " on page 17](#).

- **File > New > Artifact Type**

Create an artifact type, as described in ["Creating an Artifact Type or Package " on page 22](#).

- **File > New > Artifact Package**

Create an artifact package, as described in ["Creating an Artifact Type or Package " on page 22](#).

- **File > New > Property**

Create a property, as described in ["Creating a Property " on page 28](#).

- **File > New > Property Group**

Create a property group, as described in ["Creating a Property Group " on page 29](#).

- **File > New > Component**

Create a component, as described in ["Creating a Component " on page 31](#).

- **File > New > Task**

Create a task, as described in ["Creating a Task "](#) on page 32.

- **Navigate > Go To > Open Artifact**

Open an artifact editor, as described in ["Artifact Editor "](#) on page 38.

The keyboard shortcut **Ctrl+Alt+A** also performs this function.

- **Navigate > Go To > Open Property**

Open a property editor, as described in ["Property Editor "](#) on page 44.

The keyboard shortcut **Ctrl+Alt+P** also performs this function.

- **Navigate > Go To > Open Property Group**

Open a property group editor, as described in ["Property Group Editor"](#) on page 46.

The keyboard shortcut **Ctrl+Alt+R** also performs this function.

- **Search > Search**

Open a search dialog box, as described in ["Searching the Extension "](#) on page 20.

- **Help > Dynamic Help**

Access context-sensitive help, directing you to the relevant section of this document based on the active view in Customization Editor.

## Server Explorer

The Server Explorer displays the HPE Systinet servers connected to Workbench. The functionality is shared by all the HPE Systinet Workbench editors.

Right-click a server in the Server Explorer to open the context menu described in the “Server Explorer Context Menu Options” table below.

### Server Explorer Context Menu Options

Option	Function
New Server	Add a server for downloading assertions and taxonomies (Assertion Editor, Taxonomy Editor, and Customization Editor).
Remove Server	Delete a server from the Server Explorer.
Download Taxonomy	Download a taxonomy from a platform server (Taxonomy Editor and Customization Editor).

Option	Function
Download Assertion	Download assertions from a platform server (Assertion Editor).
Download Report	Download reports from a reporting server (Report Editor).
Properties	View and edit the server name, URL, username, and password.

## SDM Elements

Customization Editor manages four types of entities in the SDM configuration of HPE Systinet.

These entity types are described in the following sections:

- ["Artifacts " below](#)
- ["Properties " below](#)
- ["Property Groups " on page 16](#)
- ["Taxonomies " on page 16](#)

## Artifacts

Artifacts are the basic building blocks of Service-Oriented Architecture (SOA). Every entity in the repository is an instance of an artifact. Each type of artifact is defined by an artifact type in the SDM. Artifact types are further categorized into artifact packages. These packages are abstract artifact types that do not have instances in the repository but instead define groups of artifacts.

You can explore artifact types and packages in the Artifact Types branch of the Extension Explorer.

For details, see ["Artifact Types " on page 9](#).

For artifact type procedures, see ["Manipulating Artifact Types " on page 22](#).

## Properties

Properties are the attributes of artifacts, the labels that distinguish one instance of an artifact from another. For example, all artifact instances have the *name* property, but every artifact has a different

name.

There are three categories of property:

- **Primitive Properties**

The basic labels for artifacts such as name and description.

Basic properties are of the following types:

Property Type	Description
addressPropertyType	A full postal address.
booleanPropertyType	True or false.
categoryBagPropertyType	A taxonomic categorization.
dailyIntervalPropertyType	A time period with a start and end day, for example Monday to Friday.
dateTimePropertyType	A specific date and time.
doublePropertyType	A double precision floating point number.
encryptedPasswordPropertyType	An encrypted password.
identifierBagPropertyType	A taxonomic identification.
integerPropertyType	An integer number.
nameUrlPairPropertyType	A URL with an optional name assigned.
nameValuePairPropertyType	A name and value pair.
plainTextPropertyType	One-line text, suitable for properties such as name.
textareaPropertyType	Multi-line text, suitable for properties such as description.

- **Relationship Properties**

The nature and direction of a relationship with another artifact such as ImplementationOf.

- **Taxonomy Properties**

The categories used to define an artifact based on taxonomies such as the High category in the Failure Impact taxonomy.

You can explore properties in the SDM Details branch of the Extension Explorer.

For details, see ["SDM Details " on page 10](#).

For property procedures, see ["Modifying the Properties of an Artifact Type " on page 23](#)

## Property Groups

You can organize properties into groups, and add them to artifacts collectively instead of individually. For example, you can organize name, address, and telephone number into a contact details property group. Then, whenever a new artifact type is created that represents a person, the group can be added to it instead of the individual properties.

You can explore property groups in the SDM Details branch of the Extension Explorer.

For details, see ["SDM Details " on page 10](#).

For property group procedures, see ["Modifying the Properties of an Artifact Type " on page 23](#)

## Taxonomies

Taxonomies are categorization groups, each of which contain a set of values within a single category. For example, office location can be a taxonomic group containing a set of values representing each office in an organization.

You can explore taxonomies in the Taxonomies branch of the Extension Explorer.

For taxonomy procedures, see ["Modifying Taxonomies " on page 33](#).



# Chapter 3: Getting Started

You can use Customization Editor to create extensions to the core HPE configuration. The core configuration itself is not modified. Instead, HPE Systinet includes a utility for adding extensions to the configuration.

This chapter contains the following sections:

- ["Creating an Extension Project " below](#)
- ["Searching the Extension " on page 20](#)
- ["Saving Modifications " on page 20](#)

## Creating an Extension Project

The first step in customizing your HPE Systinet installation is to create an extension project in the Customization Editor. This is the workspace where you make all your modifications.

The symbol  $\triangleright$  supports the separation of projects based on their type. The following extension project types are supported:

- **model**  
Extensions for changes to the data model.
- **code**  
Extensions for custom components.
- **mixed**  
Combined model and code extensions.

HPE Software recommends separating extensions into different types because HPE Systinet does not guarantee backward compatibility for the Java API. This may make it difficult to migrate custom components to newer versions.

**model** extensions are easily migrated by opening them in the latest version of Customization Editor, building a new extension and redeploying them.

To create an extension project:

1. Select **File > New > Project**.

The New Project wizard opens.

2. Expand **HPE Systinet Extension Project** and click **Next**.
3. In the New Extension Project dialog box, specify how to connect to HPE Systinet to manage extensions (connect to remote HPE Systinet installation or HPE Systinet running on local computer)

**Note:** Creating an extension project based on a remote installation of Systinet, is not available in Production installation mode.

4. Click **Next** and follow one of the below steps:
  - a. If you connect to remote HPE Systinet installation, select or create a server. If no servers are configured currently, continue to [Step 6](#).
  - b. If you connect to the HPE Systinet running on local computer, specify the HPE Systinet server installation parameters and then click **Next** and go to [Step 7](#).

For parameter descriptions, see "[New Extension Project: Specify HPE Systinet Server Installation](#)" on page 51.

5. Do one of the following:
  - a. Select **Create a New Server** and then click **Next**. Go to [Step 6](#).
  - b. Select **Use an Existing Server**, select the server from the list and input its credentials, and then click **Next**. Go to [Step 7](#).

6. In the New Server dialog box, add the parameters you want, and then click **Next**.

For parameter descriptions, see "[New Extension Project: New Server](#)" on page 52.

7. Do one of the following:
  - a. Select **Create a New Extension Project from Scratch** to create an empty extension containing no elements at all. Click **Next**.
  - b. Select **Edit an Existing Extension** to open an existing extension for modification.
    - i. Select an extension from the extension folder, or use **Add Extension** to select one from another location.  
  
Click **Next** to continue to [Step 10](#).
  - c. Select **Create a New Extension from an Existing One** to use an existing extension as the basis of a new one.

- i. Select an extension from the extension folder or use **Add Extension** to select one from another location. Click **Next**.

If you use this option, you must remove the existing extension from the extensions folder before you deploy the new modified extension.

8. In the Create Extension or Project dialog box, add the parameters you want, and then click **Next**.

For parameter descriptions, see ["New Project: Create a New Extension " on page 53](#).

9. In the Add Dependencies dialog box, optionally select or add extensions to depend on, and then click **Next**.

10. In the HPE Systinet Extension Project dialog box, add the parameters you want, and then select available projects to reference.

Adding referenced projects enables you to use assertions or taxonomies from the referenced projects without the need to copy them manually to the extension project.

11. Click **Finish** to open the project.

In the Platform perspective, the Project Explorer now displays a view of your extension that contains not only the elements contained in your extension project but all the elements from any other extensions that your project depends on.

For details about the contents of your extension project, see ["Extension Explorer " on page 9](#).

## Sharing an Extension Project

HPE Systinet enables the sharing of an extension project in the CVS repository.

To share an extension project:

1. In Extension Explorer, right-click the project you need to open its context menu, and select **Team > Share Project** to open the **Share Project** dialog box. For details, see ["Share Project " on page 58](#).
2. Enter the required parameters and click **Finish**.

The project is published to the CVS repository.

# Referencing Other HPE Systinet Workbench Projects

You can include assertions and taxonomies from an Assertion or Taxonomy Project in your current Extension Project.

To reference an Assertion or Taxonomy Project:

1. From the platform perspective, right-click your extension project and select **Properties** to open the **Preferences** dialog box, and then select **Project References** to view projects available to reference.
2. Select the projects you want to reference and click **OK**.

The assertions or taxonomies from the referenced project are copied to your extension project.

## Saving Modifications

As you modify an entity in your extension project, the tab label of the entity editor is marked with an asterisk (\*).

To save your changes to the project, select **File > Save** from the menu, or use keyboard shortcut **Ctrl+S**.

If you close an editor or the application with unsaved changes, you are prompted to save these changes.

Your modifications are made to your extension project and not to the configuration of HPE Systinet. To deploy your customization to HPE Systinet, see "[Deploying Extensions](#) " on page 35.

## Searching the Extension

The Customization Editor contains a customized search function, which enables you to find entities in your extension project.

To search your extension project:

1. From the menu, select **Search > Search**.

The Search dialog box is displayed.

- 
2. In the SDM Search tab of the Search dialog box, add the parameters for your search.

For parameter descriptions, see ["Search " on page 58](#).

- 
- 
3. Click **Search**.

The search results is displayed in the Search view.

For details, see ["Search View " on page 47](#).

# Chapter 4: Manipulating Artifact Types

You can use Customization Editor to create, modify, and delete artifact types and packages in your extension project. You can also change the format of artifact pages in the HPE Systinet UI.

HPE Systinet separates the SDM into two models, public and system. Customization Editor hides system artifacts, preventing you from modifying them.

For more details about artifact types and packages, see ["Artifacts " on page 14](#).

The chapter explains the following procedures:

- ["Creating an Artifact Type or Package " below](#)
- ["Modifying the Attributes of an Artifact Type " on the next page](#)
- ["Modifying the Properties of an Artifact Type " on the next page](#)
- ["Mapping an Artifact Type to a Registry " on page 25](#)
- ["Example: Adding the Department Property to Services " on page 27](#)

## Creating an Artifact Type or Package

Customization Editor enables you to create a new artifact type or package in your extension project.

To create an artifact type or package:

1. Do one of the following:
  - a. From the main menu, select **New > Artifact Type** or **Package**.
  - b. In the Extension Explorer, open the context menu of an artifact package branch and select **New Artifact Type** or **Package**.

The New Artifact Type or New Artifact Package dialog box is displayed.

2. In the dialog box, add the artifact parameters you want.

For parameter descriptions, see ["New Artifact: Create " on page 49](#).

For localname naming conventions, see ["Localname Naming Rules " on page 59](#).

3. Do one of the following:

- a. Click **Finish** to create the artifact type or package and exit the dialog box.
  - b. Click **Next** and continue to [Step 4](#) to set database parameters.
4. In the dialog box, add the database parameters you want.
- For parameter descriptions, see ["New Artifact: Database Settings "](#) on page 50.
5. Click **Finish** to create the artifact type or package.

## Modifying the Attributes of an Artifact Type

Customization Editor enables you to edit the main attributes of artifact types.

To edit the attributes of an artifact type:

1. Open the artifact editor and select the **Overview** tab.
2. In the Overview tab, make any changes you want.

You cannot change the local name of the artifact if the extension project is in production mode.

For a description of the Overview tab, see ["Artifact Editor "](#) on page 38.

3. Press **Ctrl+S** to save your changes.

## Modifying the Properties of an Artifact Type

The main attributes of artifact types are properties. You can modify these properties, add them individually, or add them as a group within the artifact editor.

To add or modify the properties and property groups of an artifact type:

1. Open the artifact editor and select the **Properties** tab.
2. In the Properties tab, make any changes you want.

For a description of the Properties tab, see ["Artifact Editor: Properties Tab "](#) on page 39.

3. Press **Ctrl+S** to save your changes.

You can change the local name of the property descriptor defined in the current extension project.

To change the local name of a property descriptor:

1. From the tree menu of Extension Explorer, expand **SDM Details > Property Descriptors**.
2. Browse for the descriptor you want, and double-click to open the **Overview** view.
3. In the **Local Name** field, enter the parameters you require, and then save your changes.

You can also change the cardinality of an artifact defined in the current extension project.

**Note:** You cannot change the local name of the property descriptor if the extension project is in production mode.

To change the cardinality of an artifact type:

1. Browse for the artifact you need and open the Editor view, and then switch to the **Properties** tab.
2. In the **Cardinality** field, click to open the drop-down menu, and select the cardinality you need.

**Note:** For taxonomic properties, the cardinality affects the way the property displays in the HPE Systinet UI.

- Required taxonomic properties display as an **Add Category** function.
- Optional taxonomic properties display as a drop-down menu.
- Multiple taxonomic properties display as a list box.

**Tip:** You cannot change the cardinality of an artifact if the extension project is in production mode.

You can also change the direction of relationship of a property descriptor which is defined in the current extension project.

To change the direction of a relationship property descriptor:

1. In Extension Explorer, expand **SDM Details > Property Descriptors > Relationship Property Descriptors** and double-click the descriptor you need to open its Editor view.
2. In the **Direction** field, click the link next to **Inverse Display Name** to toggle the relationship between **incoming** or **outgoing**, and then save your changes.

HPE Systinet also enables you to modify the taxonomical properties of an artifact type.

To modify the taxonomical properties of an artifact:

1. In Extension Explorer, browse for the artifact you need and open its Editor view, and then switch to the **Properties** tab.



2. In the Properties window, click **New** to open the **New Property** dialog box, and select **Taxonomy Property**, and then click **Next**.

The **New Taxonomy Property** dialog box is displayed. For details, see "[New Property: Taxonomy](#)" on page 55.

3. Enter general parameters and in the **Taxonomy Name** field, do one of the following:
  - a. Click **Browse** to browse for and select from a list of available taxonomies.
  - b. Click **Create** to open the Create a New Taxonomy dialog box. For details, see "[New Taxonomy](#)" on page 56.
4. Enter the parameters you want and click **Finish**.

The created taxonomy is now visible in the **New Taxonomy Property** dialog box.

5. Click **Finish**.

## Mapping an Artifact Type to a Registry

You can map artifacts in HPE Systinet to registry entities in a UDDI registry.

The artifact types listed in "Predefined Registry Mapping" have predefined mappings to registry and cannot be amended.

### Predefined Registry Mapping

SDM Artifact	UDDI Entity	Direction
Organizational Unit	BusinessEntity	Bi-directional
Business Service	tModel	Bi-directional
Implementation Artifacts	Business Service	Bi-directional
Endpoint	BindingTemplate	Bi-directional

The mapping of an artifact to a registry entity is illustrated in "Registry Mapping".

To add or modify artifact type mappings to registry entities:

1. Open the artifact editor and select the **Registry Mapping** tab.
2. In the Registry Mapping tab, make changes as per the requirement.

For a description of the Registry Mapping tab, see "[Artifact Editor: Registry Mapping Tab](#)" on page 41.

For details on how to add an artifact mapping, see ["Adding a Registry Mapping for an Artifact Type" below](#)

For details on how to add a property mapping, see ["Adding a Registry Mapping for a Property" below](#).

3. Press **Ctrl+S** to save your changes.

## Adding a Registry Mapping for an Artifact Type

You can map artifact types to registry entities.

To add an artifact mapping:

1. Open the Artifact Editor and select the **Registry Mapping** tab.
2. In the Artifact Mapping pane, click **Add**.

The New Artifact Mapping dialog box is displayed.

3. In the New Artifact Mapping dialog box, add the parameters you want.

For parameter descriptions, see ["Registry Mapping: New Artifact Mapping" on page 57](#).

4. Click **Finish** to add the mapping.

## Adding a Registry Mapping for a Property

You can map properties to registry entity keyed references.

To add a property mapping:

1. Open the Artifact Editor and select the **Registry Mapping** tab.
2. In the Property Mapping pane, click **Add**.

The New Property Mapping dialog box is displayed.

3. In the New Property Mapping dialog box, add the parameters you want.

For parameter descriptions, see ["Registry Mapping: New Property Mapping" on page 57](#).

4. Click **Finish** to add the mapping.

# Example: Adding the Department Property to Services

In this example, you add a new taxonomic property, department, to the service artifact.

To follow this example, you must first do the following:

- In Customization Editor, create a new extension dependent on the core extension, as described in ["Creating an Extension Project " on page 17](#).
- In Taxonomy Editor, create a new department taxonomy and deploy it to HPE Systinet, as described in the section *Example: Creating and Publishing a Department Taxonomy in HPE Systinet Taxonomy Editor Guide*.

To add the department property to services:

1. Open the Service editor and select the **Properties** tab.
2. In the Properties segment, click **New**.
3. Select **Taxonomy Property** and click **Next**.
4. Input **Department** as the **Display Name**.
5. For the **Taxonomy**, click **Browse** and import the **Departments** taxonomy.
6. Press **Ctrl+S** to save your changes to the business service artifact type.
7. Deploy your extension, as described in ["Deploying Extensions " on page 35](#).
8. Start your installation of HPE Systinet.

Add the property to the artifact in the user interface.

# Chapter 5: Manipulating Properties

Customization Editor enables you to create, modify, and delete properties and property groups in your extension project.

**Caution:** HPE Systinet separates the SDM into two models, public and system. Customization Editor shows properties and property groups used by system artifacts, but prevents you from modifying them.

For more details about properties and property groups, see ["Properties " on page 14](#) and ["Property Groups " on page 16](#).

This chapter describes the following procedures:

- ["Creating a Property " below](#)
- ["Modifying a Property " on the next page](#)
- ["Creating a Property Group " on the next page](#)
- ["Modifying a Property Group " on page 30](#)

In addition, ["Manipulating Artifact Types " on page 22](#) includes the following procedures related to properties:

- ["Modifying the Properties of an Artifact Type " on page 23](#)
- ["Mapping an Artifact Type to a Registry " on page 25](#)

## Creating a Property

You can create properties individually and then add them to artifact types later.

To create a property:

1. Do one of the following:
  - a. From the menu, select **File > New > Property**.
  - b. In the Extension Explorer, open the context menu for the Property Descriptors branch, select **New Property**, and then select the property type.

- c. In the Extension Explorer, open the context menu for a property type branch and select **New Type Property** to create a property of that type.

The New Property dialog box is displayed.

2. In the New Property dialog box, add the property parameters you want.

For parameter descriptions of each property type, see the following:

- o ["New Property: Primitive " on page 54](#)
- o ["New Property: Relationship " on page 54](#)
- o ["New Property: Taxonomy " on page 55](#)

For localname naming conventions, see ["Localname Naming Rules " on page 59](#).

3. Click **Finish** to create the property.

## Modifying a Property

Customization Editor enables you to modify properties.

To edit a property:

1. Open the Property Editor.
2. In the Property Editor, make the required changes.

For a description of the Property Editor, see ["Property Editor " on page 44](#).

3. Press **Ctrl+S** to save your changes.

## Creating a Property Group

Many artifact types share sets of the same properties. It is useful to create groups of properties and add these to artifact types instead of adding each property individually.

To create a property group:

1. Do one of the following:
  - a. From the menu, select **File > New > Property Group**.
  - b. In the Extension Explorer, open the context menu for the Property Groups branch, and select

### **New Property Group.**

- c. Open the Artifact Editor and select the Properties tab. In the Property Groups pane, click **New** to create a new property group as a set of attributes for an artifact type.

The New Property Group dialog box is displayed.

2. In the New Property Group dialog box, add the parameters you want.

For parameter descriptions, see "[New Property Group](#)" on page 56.

3. Click **Finish** to create the property group.

## Modifying a Property Group

Customization Editor enables you to modify property groups.

To edit a property group:

1. Open the property group editor.
2. In the Property Group Editor, make the required changes.

For a description of the Property Group Editor, see "[Property Group Editor](#)" on page 46.

3. Press **Ctrl+S** to save your changes.

# Chapter 6: Creating and Using Components

Customization Editor enables you to create custom components, for use in the customizable HPE Systinet user interface.

To create and use components:

1. Create a new extension project, selecting **Develop JSPs**, and setting the appropriate application server settings.

For details, see as described in ["Creating an Extension Project " on page 17.](#)

2. Create a component.

For details, see ["Creating a Component " below.](#)

3. Create a task to use your component.

For details, see ["Creating a Task " on the next page.](#)

4. Develop the Java and JSP that the component uses.

For details, see ["Developing a Component " on the next page.](#)

5. Deploy the code extension to HPE Systinet.

For details, see ["Deploying Extensions " on page 35.](#)

6. Perform the appropriate UI customization to use your component in HPE Systinet.

## Creating a Component

Components are functional elements that you can call using a number of different elements in the HPE Systinet UI.

To create a component:

1. Do one of the following:
  - a. In the Extension Explorer, open the context menu for the Components branch and select **New Component**.

- b. In the Components Editor, click **Add**.

The New Component dialog box is displayed.

2. In the New Component dialog box, add the parameters you want.

For parameter definitions, see "[New Component](#)" on page 50.

3. Click **Finish** to create the component.

## Creating a Task

Tasks are top level components accessible via their URIs. The addition of a URI allows you to access the component in the HPE Systinet UI, and to create more sophisticated multi-layered components.

To create a task:

1. Do one of the following:
  - a. In the Extension Explorer, open the context menu for the Tasks branch and select **New Task**.
  - b. In the Tasks editor, click **Add**.

The New Task dialog box is displayed.

2. In the New Task dialog box, add the parameters you want.

For parameter descriptions, see "[New Task](#)" on page 56.

3. Click **Finish** to create the task.

## Developing a Component

When you create a component, as described in "[Creating a Component](#)" on the previous page, the templates for the Java class and JSP page are created automatically.

To view this code, switch to the Java or Debug perspective.

The project contains two source folders containing the Java and JSP files, respectively.

To open an edit view, double-click the source file you want to edit. In the edit view, you can write a custom component.



# Chapter 7: Modifying Taxonomies

Customization Editor controls which taxonomies are available in the extension project. You can create and modify taxonomies from your extension project.

Creating and modifying taxonomies uses the Taxonomy Editor functionality.

For details, see the *HPE Systinet Taxonomy Editor Guide*.

To control taxonomies in your extension project:

1. In the Extension Explorer, double-click the Taxonomies branch to open the Taxonomies editor.
2. To modify the available taxonomies in your extension project, do one of the following:

- a. Open the taxonomy from the extension project, and edit the parameters as required.

The modified taxonomy is then indicated by a > next to the taxonomy.

- b. To import taxonomies from HPE Systinet and add them to the extension, click **Import**. Select multiple taxonomies by holding the **Ctrl** key.

You can also import taxonomies in the Server Explorer view. For more details, see "[Server Explorer](#)" on page 13.

- c. To update taxonomies in your extension to newer versions from HPE Systinet, click **Update**. Select multiple taxonomies by holding the **Ctrl** key.

Alternatively, in the Extension Explorer, expand the branches in the Taxonomy branch, select the taxonomy (use the **Ctrl** key to select multiple taxonomies), open the context menu, and select **Update Taxonomy**.

The selected taxonomies are added as separate items for modification under the Taxonomy branch.

- d. To remove a taxonomy from the extension, select the taxonomy and click **Remove**. Select multiple taxonomies by holding the **Ctrl** key.

To enable the import or update of taxonomies from HPE Systinet, you must specify the correct server URL during configuration.

For details, see "[Creating an Extension Project](#)" on page 17.

You can change the taxonomy in a taxonomy based property descriptor which is used in the current extension project.

To change the taxonomy in a taxonomy based property descriptor:

1. In the tree menu of Extension Explorer, expand **SDM Details > Taxonomy Property Descriptors** and double-click the descriptor you need to open its Editor in the Overview tab.
2. In the **Taxonomy Name** field, click **Browse**.
3. Select the taxonomy you want, and click **Select**.

The new taxonomy overrides the previous taxonomy in the property descriptor.

You cannot change the taxonomy in a taxonomy based property descriptor if the extension project is in production mode.

# Chapter 8: Deploying Extensions

Deploying an extension to HPE Systinet is a three-part process:

1. ["Exporting the Extension Project "](#) below
2. ["Redeploying the EAR File "](#) on the next page
3. ["Deploying Components to HPE Systinet "](#) on the next page
4. ["Deploying Extensions Remotely"](#) on page 37

Alternatively, if you are developing components in Customization Editor, you can directly deploy an extension from Customization Editor.

## Exporting the Extension Project

Customization Editor displays the entire configuration of your extension and other dependent extensions. However, your extension consists only of any modifications and additions that you have made. Only these changes are stored when you save your project as an extension JAR file, and only these changes are deployed to HPE Systinet.

To create your extension package:

1. In the Extension Explorer or Project Explorer, open the context menu of the extension name, and select **HPE Systinet > Build Extension**.

The Export Extension dialog box is displayed.

2. Enter file name and location to save. Click **Save**.

By default, the Extension Folder set during configuration is selected.

For configuration details, see ["Creating an Extension Project "](#) on page 17.

If the extension already exists, Customization Editor prompts you to confirm that you want to overwrite it.

## Redeploying the EAR File

After using the Setup Tool to apply extensions or updates, you must redeploy the EAR file to the application server. For JBoss, you can do this using the Setup Tool.

To redeploy the EAR file to JBoss:

1. Stop the application server.
2. Start the Setup Tool by executing the following command:

**SYSTINET\_HOME/bin/setup.bat(sh).**

3. Select the **Advanced** scenario and click **Next**.
4. Scroll down to select **Department** and click **Next**.

When the Setup Tool validates the existence of the JBoss Deployment folder, click **Next**.

5. Click **Finish** to close the Setup Tool.
6. Restart the application server.

## Deploying Components to HPE Systinet

If you are creating components in Customization Editor, it is not convenient to continually build and apply extensions to HPE Systinet. Customization Editor enables you to deploy your extension directly to HPE Systinet.

Only HPE Systinet deployed to a JBoss application server is supported by this release.

To directly deploy an extension from Customization Editor:

1. Stop the application server.
2. Switch to the Debug perspective in Customization Editor.
3. From the main menu, select **Run > External Tools > Open External Tools Dialog**.

The External Tools dialog box is displayed.

4. Select **Ant Build > Customization Editor – Apply Extension**.
5. Click **Run** to apply the extension.

The Console view displays the output of the deployment process.

# Deploying Extensions Remotely

In addition to manual deployment, you can deploy an extension remotely from Workbench to Systinet server.

To deploy the extension, follow these steps:

1. Open the extension project.
2. In the Extension Explorer or Project Explorer, open the context menu of the extension name and select **HPE Systinet > Apply To Server**.

You can also apply the extension to other servers. Right-click the extension project name in the Extension Explorer or Project Explorer to open the context menu and select **HPE Systinet > Apply To Other Server**.

3. Follow the subsequent screens to deploy the extension to Systinet server.

**Note:** Remote deployment requires administrative rights. Thus, you are prompted to provide administrative credentials.

# Chapter 9: Views

Each Customization Editor view in the main and bottom-right section of the editor is described in the following sections:

- ["Artifact Editor " below](#). Manage your SOA artifacts and customize their properties, appearance, and registry mapping.
- ["Messages View " on page 42](#). View the action log as you customize your extension project.
- ["Project Editor " on page 42](#). Manage your extension project.
- ["Property Editor " on page 44](#). Manage your SOA properties and which perspectives can see and edit them.
- ["Property Group Editor" on page 46](#). Organize your properties into groups.
- ["Search View " on page 47](#). View the results of usage and entity searches.
- ["Tasks Editor " on page 47](#). Manage tasks in HPE Systinet.
- ["Taxonomies Editor " on page 48](#). Customize the taxonomies available in HPE Systinet.

## Artifact Editor

Artifacts are the basic building blocks of SOA and represent all types of entity. This editor enables you to modify the parameters, and properties of an artifact type within HPE Systinet. Each artifact type and package has an editor with the artifact name as the title.

Each editor contains the following tabs described in the following sections:

- ["Artifact Editor: Overview Tab " on the next page](#)  
The main attributes of the artifact type.
- ["Artifact Editor: Properties Tab " on the next page](#)  
The properties of the artifact type.
- ["Artifact Editor: Registry Mapping Tab " on page 41](#)  
The mapping of the artifact to registry entities.

# Artifact Editor: Overview Tab

The **Overview** tab displays the general attributes of the artifact and allows you to modify some of them.

The Overview tab contains the following segments:

- **Artifact General Information.** The following parameters related to artifacts shown in this pane. They are described in the table below:

Parameter	Definition
URI	The identifier for the artifact descriptor in the configuration.
Display Name	The name of the artifact as it appears in the HPE Systinet UI.
Local Name	The name of the artifact as it is stored in the extension.
Collection Display Name	The plural version of the name as it appears in the HPE Systinet UI.
Artifact Icon	Displays the icon used for the artifact in the UI with an option to change it.
Package	The parent artifact type that this artifact type belongs to.
Extends Type	The artifact type that this artifact type inherits the properties of.
Description	The description of the artifact type as it appears in the HPE Systinet UI.
Deprecated	A check box indicating whether the artifact is currently active in the extension.
Revisionless	Indicates whether the artifact creates a new revision when it updates.
Abstract	A check box indicating whether the artifact is an artifact package which does not have actual instances in the repository.
Data Attachment	Indicates the type of data content that may be attached to the artifact.

- **Properties**

A summary of different properties that the artifact has. Each of the links opens the **Properties** tab described in "[Artifact Editor: Properties Tab](#)" below.

# Artifact Editor: Properties Tab

The **Properties** tab displays all the properties of the artifact and enables you to add and remove them.

This tab contains the following panes:

- **Properties**

This is the list of individual properties of the artifact. On the right are the following buttons:

- **New** opens the **New Property** dialog described in ["Creating a Property "](#) on page 28.
- **Add** adds a property to the artifact type from the list of available property descriptors, set its cardinality, and its location on the **View Artifact** page.
- **Deprecate** deactivates the property in the artifact type.
- **Undeprecate** reactivates the property in the artifact type.
- **Remove** removes the selected property from the artifact type.

- **Property Groups**

This is the list of the property groups of the artifact. On the right are the following buttons:

- **New** opens the **New Property Group** dialog. For details, see ["Creating a Property Group "](#) on page 29.
- **Add** adds a property group to the artifact type from the list of available property groups.
- **Remove** removes the selected property group from the artifact type.

- **Inherited Groups and Properties**

If the artifact is based on another artifact then this segment displays the inherited artifact and its properties and groups.

The following property parameters are displayed in these panes:

Parameter	Definition
Name	The name of the property as it appears in the HPE Systinet UI.
Cardinality	The occurrence of the property in an artifact with options: <ul style="list-style-type: none"> <li>◦ <b>Optional</b> The property is not required to be populated.</li> <li>◦ <b>Required</b> The property must be populated.</li> <li>◦ <b>Multiple</b> The cardinality is determined by the MinOccurs/MaxOccurs defined in the <a href="#">"Property Editor: Advanced Tab "</a> on page 45.</li> </ul>
Extension	The extension to which this property is part of.



# Artifact Editor: Registry Mapping Tab

The **Registry Mapping** tab displays any mappings between the artifact and entities in registries and allows you to create, edit, and remove them.

This tab has the following panes:

- **Artifact Mapping**

This is the list of registry entities that represent the artifact in registries. On the right are the following buttons:

- **Add** adds a registry mapping, as described in ["Mapping an Artifact Type to a Registry " on page 25](#)
- **Edit** edits the mapping, as described in ["Mapping an Artifact Type to a Registry " on page 25](#)
- **Remove** removes the mapping from the artifact.

The following registry mapping parameters are displayed in this pane:

Parameter	Definition
Registry Name	The registry that the artifact type is mapped to.
TModel Key	The taxonomy key used to categorize the UDDI entity in the registry.
Key Name	The name used to categorize the UDDI entity in the registry.
Key Value	The value used to categorize the UDDI entity in the registry.
Extension	The extension to which this mapping is part of.

- **Property Mappings**

This is the list of the properties of the artifact and if they are mapped to a registry. On the right are the following buttons:

- **Add** adds a new registry mapping for the selected property, as described in ["Mapping an Artifact Type to a Registry " on page 25](#).
- **Edit** edits a property mapping, as described in ["Mapping an Artifact Type to a Registry " on page 25](#).
- **Remove** removes a mapping from the selected property.

The following registry mapping parameters are displayed in this pane:

Parameter	Definition
Property Name	The name of the property.
Registry Name	The registry that the artifact type property is mapped to.
TModel Key	The taxonomy key used to categorize the UDDI entity in the registry.
Key Name	The name used to categorize the UDDI entity in the registry.
Extension	The extension to which this property is part of.

## Messages View

The **Messages** view tracks the changes that you make to extension entities and displays any warnings and problems that may occur as a result of those actions:

Double-click an item in the tree to open the relevant editor for that object.

## Project Editor

The project editor allows you to configure your extension project. It contains three tabs described in the following sections:

- ["Project Editor: Overview Tab " below](#)
- ["Project Editor: Environment Tab " on the next page](#)
- ["Project Editor: Dependencies Tab " on the next page](#)

## Project Editor: Overview Tab

The **Overview** tab enables you to view and amend the basic parameters of your extension project.

The **Overview** tab contains the following collapsible panes:

- **Project details** displays the following parameters for the extension project:

Parameter	Definition
Project name	The name of the extension project.
Project type	The project is an extension project.
Project location	The workspace folder that contains the extension project.
Last modified	The last time the project was changed.

- **Extension details** enables you to configure some of the extension project parameters:

Parameter	Definition
Name	The display name of the extension.
Version	The version number of the extension project.
Description	A description of the extension project.
URI	The identifier for the extension in the configuration (not editable).
Namespace	The prefix used for the URI when you create a new artifact type or property.

## Project Editor: Environment Tab

The **Environment** tab displays the parameters of the HPE Systinet server that the Customization Editor is configured for.

The **Environment** tab contains the following parameters:

Parameter	Definition
Platform Home	Your HPE Systinet installation folder.
Extension Folder	The location of the extension folder in your HPE Systinet installation.
Server URL	The URL used to access HPE Systinet.
Deploy JSPs	Indicates if component and JSP development is configured for your extension project.
JBoss Deploy Directory	Deployment directory for your JBoss application server.
JBoss JNDI Port	JNDI port configured in your JBoss application server.

## Project Editor: Dependencies Tab

The **Dependencies** tab displays general information about your extension project and any dependencies that it has:

The **Dependencies** tab contains the following panes:

- **Extension dependencies** displays a tree of extensions that the extension project is dependent on.
- **Project and extension general information** displays parameters of the extension project:

Parameter	Definition
Name	The name of the extension.
Version	The version number of the extension project.
Description	A description of the extension project.
URI	The identifier for the extension project in the configuration (not editable).
Buildtime	The creation date and time of the extension.

## Property Editor

Properties are attributes of artifacts. The property editor allows you to modify the properties in your extension project. Each property has an editor with the property name as the title containing the tabs described in the following sections:

- ["Property Editor: Overview Tab "](#) below  
The main attributes of the property.
- ["Property Editor: Advanced Tab "](#) on the next page  
The advanced attributes of the property.

## Property Editor: Overview Tab

The **Overview** tab displays the general attributes of the artifact and enables you to modify some of them.

This tab contains the following panes:

- **General**

The following parameters are displayed in General pane:

Parameter	Definition
Display Name	The name of the property as it appears in the HPE Systinet UI.
Local Name	The name of the property as it is stored in the extension.
URI	The identifier for the property descriptor in the configuration.
Description	The description of the property as it appears in the HPE Systinet UI.
Collection Display Name	The plural version of the name as it appears in the HPE Systinet UI.
Deprecated	A check box indicating whether the property is currently active in the extension.

- **Property Type, Relationship, or Taxonomy**

There are three types of property and this segment is specific to each:

- **Property Type**

The property type for primitive properties.

- **Relationship**

This pane displays the source and target artifacts that this relationship links and some parameters of the inverse relationship. On the right are two sets of two buttons for altering the source and target artifacts:

- **Add** selects an artifact in the extension to source or target.
- **Remove** removes the relationship from a source or target artifact.

- **Taxonomy**

The taxonomy that contains the available options for this property with **Browse** to select a new taxonomy from the extension project.

## Property Editor: Advanced Tab

The **Advanced** tab displays the advanced attributes of the property and allows you to modify them.

This tab contains the following panes:

- **Multiple Cardinality:**

The following are the parameters of the multiple cardinality pane:

Parameter	Definition
MinOccurs	If a property has multiple cardinality, this parameter is the minimum occurrences of the property in an artifact.
MaxOccurs	If a property has multiple cardinality, this parameter is the maximum occurrences of the property in an artifact. Use <b>unbounded</b> if there is no limit.

**Caution:** Properties that inherit `urn:com:systinet:soa:model:propertyTypes:text`, such as **Name**, cannot have multiple instances. If you need a text property with multiple cardinality, use one that inherits `urn:com:systinet:soa:model:propertyTypes:plainText`.

**Note:** Cardinality for incoming relationship properties cannot be changed.

- **Default Values:**

The following are the parameters of the default values pane:

Parameter	Definition
Default Value	An XML extract defining the default value.
Multiple Default Value	An XML extract defining default values in the case of multiple cardinality.

- **Database Sizes**

Displays optional database sizing for property elements with **Add**, **Edit**, and **Remove** functionality.

## Property Group Editor

Properties can be organized into groups. The property group editor allows you to add and remove properties from a property group. Each property group has an editor with the property group name as the title.

The Property Group Editor has the following panes:

- **General** enables you to change the **Display Name** and view the **URI** of the property group.
- **Properties** enables you to manage the property group.

The following are the available options on the right pane:

- **New** creates a new property, as described in ["Creating a Property Group "](#) on page 29.
- **Add** adds a property to the group from the list of available properties.
- **Remove** removes the selected property from this property group.

The following property parameters are displayed in Property Group Editor:

Parameter	Definition
Name	The name of the property.
Cardinality	The occurrence of the property in an artifact with options: <ul style="list-style-type: none"> <li>◦ <b>Optional</b> The property is not required to be populated.</li> <li>◦ <b>Required</b> The property must be populated.</li> <li>◦ <b>Multiple</b> The property can occur multiple times with different values.</li> </ul>
Extension	The extension to which this property is part of.

## Search View

The **Search** view displays the results of a search or find usage query.

The results are shown as a tree of artifact types, properties, and property groups containing the requested item.

Double-click an artifact type, property, or group to open the relevant editor.

## Tasks Editor

Tasks are top level HPE Systinet UI components accessible via their URIs. The tasks editor allows you to create, edit, and remove tasks from your extension project.

The following are the available options on the right pane:

- **New** opens the **New Task** dialog box, as described in ["Creating a Task "](#) on page 32.
- **Edit** opens the **Edit Task** dialog box with the same parameters described in ["Creating a Task "](#) on

[page 32](#).

- **Remove** deletes the selected task from your extension project.

The following parameters are displayed in Task Editor:

Parameter	Definition
URI	The identifier for the task.
Component	The component the task uses.
Caption	A name for the task.
Extension	The extension to which this property is part of.

## Taxonomies Editor

Taxonomies are category groups that allow you to organize your services. The taxonomies editor allows the import and removal of taxonomies from your extension project.

The following are the available options on the right pane:

- **Import** imports taxonomies from the active HPE Systinet server.  
The referenced HPE Systinet server must be running during import.
- **Update** refreshes the taxonomy list with any changes from the HPE Systinet server.
- **Remove** deletes the selected taxonomy from the extension project.

The following parameters are displayed in Taxonomies editor:

Parameter	Definition
Name	The name of the taxonomy as it is stored in the extension.
TModel Key	The taxonomy key identifier.
Extension	The extension to which this property is part of.



# Appendix A: Dialog Boxes

Each Customization Editor input dialog is described in the following sections:

- ["New Artifact " below](#). Create a new artifact type or package.
- ["New Component " on the next page](#). Create a new component.
- ["New Extension Project " on page 51](#). Create an extension project.
- ["New Property " on page 54](#). Create a new property.
- ["New Property Group " on page 56](#). Create a new property group.
- ["New Task " on page 56](#). Create a new task.
- ["Registry Mapping " on page 57](#). Add registry mapping.
- ["Search " on page 58](#). Search your extension project.

## New Artifact

The New Artifact Type and New Artifact Package dialogs are identical and consist of the following ordered stages:

1. ["New Artifact: Create " below](#)
2. ["New Artifact: Database Settings " on the next page](#)

## New Artifact: Create

The following are the parameters to create the new artifact:

Parameter	Definition
Display Name	The name of the artifact as it appears in the HPE Systinet UI.
Local Name	The name of the artifact as it is stored in the extension.
URI	The identifier of the artifact descriptor in the configuration.

Parameter	Definition
Collection Display Name	The plural name of the artifact as it appears in the HPE SystinetM UI.
Package	Click <b>Browse</b> to select a parent artifact package to which the artifact belongs. The default is Systinet if the new artifact is not created from within an existing artifact package.
Extends	If you select <b>Extends</b> , the new artifact inherits any future changes made to the chosen parent artifact.  Click <b>Browse</b> to select an artifact type to inherit the properties from
Description	The description of the artifact type as it appears in the HPE Systinet UI.
Data Attachment	If the artifact type normally has attached data content, select the content type from the drop-down list.
Revisionless	Specify the artifact to be revisionless. These type of artifacts update without creating a new revision.

## New Artifact: Database Settings

The following are the parameters for database settings of the new artifact:

Parameter	Definition
Collection name	The name of collection where the artifact instances are stored.
Database table name	The name of the database table where instances of this artifact are stored.

**Note:** You cannot change these parameters if the new artifact extends an artifact type that already defines them.

## New Component

The following are the general parameters for the new component:

Parameter	Definition
Name	The name of the component as it is stored in the extension.

Parameter	Definition
Page	The JSP used by the component.
Class name	The Java class used by the component.
Component parameters	Use <b>Add</b> and <b>Remove</b> to select parameters to use with the component.

## New Extension Project

The New Extension Project dialog consists of the following stages depending on the options you select:

1. ["New Extension Project: Specify HPE Systinet Server Installation "](#) below
2. ["New Extension Project: New Server "](#) on the next page
3. ["New Project: Select Working Extensions "](#) on the next page
4. ["New Project: Create a New Extension "](#) on page 53
5. ["New Project: Create Dependencies "](#) on page 53

## New Extension Project: Specify HPE Systinet Server Installation

The following are the parameters for the HPE Systinet Server:

Parameter	Definition
Platform Home	Use <b>Browse</b> to select the HPE Systinet platform installation folder. If Workbench is installed on a different machine other than the Systinet server, select an empty folder.
Extension Folder	Use <b>Browse</b> to select the extension folder in your HPE Systinet installation (populated by default based on the <b>Platform Home</b> input). If Workbench is installed on a different machine other than the Systinet server, copy the core extension file from HPE Systinet installation folder (SYSTINET_HOME/extensions/) to this folder.
Production Mode	Select this check-box to create the project in production mode (selected by default). This disables customizations

Parameter	Definition
	that would require the deletion of server data during the application of the extension to HPE Systinet.
Project Type	Select from the following options: <ul style="list-style-type: none"> <li>• <b>model</b> for changes to the SDM model.</li> <li>• <b>code</b> for custom code components and UI customizations.</li> <li>• <b>mixed</b> for both types of changes (not recommended).</li> </ul>
JBoss Deploy Directory	Use <b>Browse</b> to select the deployment directory for your application server (JBOSS_HOME\standalone\deployments).
JBoss JNDI Port	Input the JNDI Port for your application server.

Only JBoss is supported for the development of components and tasks in this release.

## New Extension Project: New Server

The following are the parameters for the HPE Systinet new server:

Parameter	Definition
Name	The name you want to use for the new server.
URL	The endpoint URL of the server you want to use.
Username and Password	Your login credentials for the specified server.
Save password	Select this check-box to store your login credentials.
Validate connection	Select this check-box to validate connection to the server. Otherwise, you can validate connection the next time you log in.

## New Project: Select Working Extensions

The following are the parameters to create working extensions:

Selection	Action
Create a new extension from	Click <b>Next</b> .

Selection	Action
scratch	
Edit an existing extension	<p>Do one of the following:</p> <ol style="list-style-type: none"> <li>1. Select from the available extensions and click <b>Next</b>.</li> <li>2. Click <b>Add extension</b> to browse and select a different extension, and then click <b>Next</b>.</li> </ol> <p>You cannot edit the core extension.</p>
Create a new extension from an existing one	<p>Do one of the following:</p> <ol style="list-style-type: none"> <li>1. Select from the available extensions and click <b>Next</b>.</li> </ol> <p>You cannot use the core extensions.</p> <ol style="list-style-type: none"> <li>2. Click <b>Add extension</b> to browse and select a different extension, and then click <b>Next</b>.</li> </ol>

## New Project: Create a New Extension

The following are the general parameters to create new extension:

Parameter	Definition
Name	The name of your extension project.
Namespace	The prefix used for the URI when creating a new artifact type or property.
Description	A description of your extension project.
Version	The extension version number.
URI	The unique identifier of the extension.

## New Project: Create Dependencies

Select extensions to declare dependencies.

**Tip:** Your changes are stored in your extension. You can change the labels and descriptions of elements from dependent extensions but you cannot delete them or make any other changes. If you want to delete an element from a dependent extension then deprecate it.

# New Property

The New Property dialog contains parameters according to the property type:

- ["New Property: Primitive " below](#)
- ["New Property: Relationship " below](#)
- ["New Property: Taxonomy " on the next page](#)

For details about property types, see ["Properties " on page 14](#).

## New Property: Primitive

The following are the general parameters for the new primitive property:

Parameter	Definition
Display Name	The name of the property as it appears in the HPE Systinet UI.
Local Name	The name of the property as it is stored in the extension.
URI	The identifier for property in the configuration.
Collection Display Name	The plural name of the property as it appears in the HPE Systinet UI.
Description	The description of the property as it appears in the HPE Systinet UI.
Property Type	Use <b>Browse</b> to select a property from the available primitive property types. For details, see <a href="#">"Properties " on page 14</a> .

## New Property: Relationship

The following are the general parameters for the new relationship property:

Parameter	Definition
Display Name	The name of the property as it appears in the HPE Systinet UI.
Local Name	The name of the property as it is stored in the extension.
URI	The identifier for the property in the configuration.

Parameter	Definition
Collection Display Name	The plural name of the property as it appears in the HPE Systinet UI.
Description	The description of the property as it appears in the HPE Systinet UI.
Relation Type	Select a type from the drop-down list.
From Entity	Use <b>Browse</b> to select the source artifact type or property group of the relationship.
To Entity	Use <b>Browse</b> to select the target artifact type or property group of the relationship.
Inverse Display Name	The name of the inverse relationship property as it appears in the HPE Systinet UI.
Inverse Local Name	The name of the inverse relationship property as it is stored in the extension.
Inverse URI	The identifier for the inverse property descriptor in the configuration.
Inverse Collection Display Name	The plural name of the inverse relationship property as it appears in the HPE Systinet UI.
Inverse Relationship Description	The description of the inverse relationship property as it appears in the HPE Systinet UI.

## New Property: Taxonomy

The following are the general parameters for the new taxonomy property:

Parameter	Definition
Display Name	The name of the property as it appears in the HPE Systinet UI.
Local Name	The name of the property as it is stored in the extension.
URI	The identifier for the property in the configuration.
Collection Display Name	The plural name of the property as it appears in the HPE Systinet UI.
Description	The description of the property as it appears in the HPE Systinet UI.
Taxonomy	Use <b>Browse</b> to select a taxonomy from the available taxonomies with an option to import taxonomies available in HPE Systinet that are not in your extension.

## New Property Group

The following are the general parameters for the new property group:

Parameter	Definition
Display Name	The name of the property group as it appears in the HPE Systinet UI.
URI	The identifier for the property group descriptor in the configuration.
Properties	Use <b>Add</b> and <b>Remove</b> to select the properties in the group.

## New Task

The following are the general parameters for the new task:

Parameter	Definition
Uri	The identifier for the task.
Component	Use <b>Browse</b> to select the component to associate with the task.
Caption	A visible name for the task.
Task parameters	Use <b>Add</b> and <b>Remove</b> to select parameters to use with the task.

## New Taxonomy

The following are the general parameters for the new taxonomy:

Parameter	Definition
Taxonomy Name	The name of the new taxonomy as it will appear in the HPE Systinet UI.
Taxonomy ID	The system identifier of the new taxonomy.
Filename	The name of the taxonomy as it appears in the repository.



# Registry Mapping

- ["Registry Mapping: New Artifact Mapping "](#) below
- ["Registry Mapping: New Property Mapping "](#) below

## Registry Mapping: New Artifact Mapping

The following are the general parameters for the new registry mapping of the artifact:

Parameter	Definition
Registry	Use <b>Select</b> to select from the available registries.
T-ModelKey	Use <b>Add</b> to select from the available taxonomies or input one.
T-ModelKey version 2	If you are mapping to a UDDI version 2 registry, input a taxonomy key.
Key Name	The name used to categorize the UDDI entity in the registry.
Key Value	The value used to categorize the tModel in the registry.
Display Name	The name of the mapping as it appears in the HPE Systinet UI.
Cache content	If selected, the content of documents referenced from the UDDI entity are cached in HPE Systinet.
Direction	Select the direction of the mapping from the drop-down list.

## Registry Mapping: New Property Mapping

The following are the general parameters for the new registry mapping of the property:

Parameter	Definition
Property	Use <b>Select</b> to select from the available properties.
Direction	Select the direction of the mapping from the drop-down list.
Registry	Use <b>Select</b> to select from the available registries.
T-ModelKey	Use <b>Add</b> to select from the available taxonomies or input one.

Parameter	Definition
T-ModelKey version 2	If you are mapping to a UDDI version 2 registry, input a taxonomy key.
Key Name	The name used to categorize the UDDI entity in the registry.

## Search

The following are the general parameters to search extension project:

Parameter	Definition
Containing text	The parameter to search.
Case sensitive	Check the box to make the search case sensitive.
Regular expression	Enables more sophisticated search parameters (for example, to find everything with more than one word). This is an advanced topic beyond the scope of this guide.
Search For	Select the SDM entities to search.
Search In	Select the entity attributes to search.
Scope	Select the extension to search.

## Share Project

The following are the parameters to define for the CSV repository:

Parameter	Definition
Host	The name of the host server.
Repository path	The location of the host server.
User and Password	Your login credentials for the host server.
Connection type	The communication protocol for the CVS host server.

# Appendix B: Localname Naming Rules

The naming strategy used to generate the names of database tables and columns can only process a specific subset of XML identifiers (element and attribute names). This is because the existing database engines have limits for the length of tablename/columnname identifiers. XML identifiers are generated from the SDM configuration in a straightforward way. This section omits the XML layer for simplicity and summarizes the rules at the SDM level which is more user-friendly.

To create an artifact type or property for localname, ensure to follow these rules:

- Use Latin characters where possible (A-Z, a-z).
- Use short identifiers where possible to improve the readability of the DB schema. For example, `stSchtronXs1t` is a better localname than `searchAndTransformSchematronXs1tUr1Link`.
- You can safely use underscores in localnames.

Do not use numbers in localnames. Avoid using numbers on property localnames when the property is in multiple cardinality.

**Note:**

- The extension application may fail if you use numbers in localnames.
- There are no restrictions on property group localnames as they are not represented in the database schema.

It is difficult to generate separate validation rules for properties and artifacts because the generated DB identifiers are often influenced by a combination of both.