# Defects and Requirements Exchange with HP Service Manager and HP Application Lifecycle Management

Software Version: 1.03 for Process Designer (PD) Content Pack 9.30.2 and 9.30.3 For the supported Windows<sup>®</sup> operating system

Installation and Administration Guide



Document Release Date: April 2015 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**

© 2008-2015 Hewlett-Packard Development Company, L.P.

#### **Trademark Notices**

Adobe® is a trademark of Adobe Systems Incorporated. AMD and the AMD logo are trademarks of Advanced Micro Devices, Inc. Intel and Pentium are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both. Java and Oracle® are registered trademarks of Oracle and/or its affiliates. Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation. UNIX® is a registered trademark of The Open Group.

#### **Documentation Updates**

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

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

To check for recent updates or to verify that you are using the most recent edition of a document, go to: 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

# Contents

Chapter 1: Introduction	9
Document Purpose	
Synchronization Concepts	
References	11
Chapter 2: Planning the Deployment	12
Supported Products and Platforms	12
Deployment Scenarios	13
Data Types	13
Deployment Tasks	13
Release Package	14
Chapter 3: Installing and Configuring QC/ALM Synchronizer	15
Installing QC/ALM Synchronizer	15
Installation Requirements	
Deploying the Adapters	
Generating/Deploying the Stub	17
Copying SM Adapter Configuration Files	
Editing the Configuration Files	18
Module Types	
Field Types	
Requirements	
SM Field Type and Definition Rule SM Change Management Example	
SM Problem Management Example	
- · · · · · · · · · · · · · · · · · · ·	
Chapter 4: Configuring Links in QC/ALM Synchronizer	24
Create a Link	
Advanced Parameters	28
QC/ALM Field <-> SM Field Mappings	

Creating Mappings	29
General Mapping Requirements	
Matching Types	
List Value Mappings	
Constant -> SM Field Mappings	35
Chapter 5: Integration Account	
Creating an SM Integration Account	36
Create a Contact Record	
Create a Profile Record	
Create a SecRole Record	
Create an Operator Record	
Creating a QC/ALM Integration Account	42
Chapter 6: SM Change -> QC/ALM Defect	
Customizing Service Manager/ServiceCenter for Change Management	49
Add Fields	
Specify the External Access Definition on Service Manager	
Specify the External Access Definition on ServiceCenter	
Create a Subform	
Add the Subform to a Form	
Add Format Control Calculations/Validations	
Add Rule Set Calculations/Validations	
Copy an Existing Workflow	
Associate an Existing Change Category with the New Workflow	
Create New Rule Set for Initialization and Validation	
Associate the New Workflow with the New Rule Set	
Customizing the QC/ALM Defects Module	66
On QC 10 or Earlier	
Add Fields	66
Add Tabs	67
Add Fields to Tabs	69
Verify	70
On ALM 11	
Add fields	72

Verify	73
Configuring Links in QC/ALM Synchronizer	75
Specify Endpoints / Type of Link	75
Define Field Mappings	76
QC/ALM Field <-> SM Field	77
Define Events	
Test the Link	
Chapter 7: SM Change -> QC/ALM Requirement	83
Customizing Service Manager/ServiceCenter for Change Management	83
Customizing the QC/ALM Requirements Module	83
On QC 10 or Earlier	83
Add Fields	
Add Tabs	
Add Fields to Tabs	
Create the SM Incoming Changes Folder	87
On ALM 11	88
Add Fields	
Create the SM Incoming Changes Folder	
Configuring Links in QC/ALM Synchronizer	
Specify Endpoints / Type of Link	90
Define Field Mappings	92
QC/ALM Field <-> SM Field	93
Define Events	94
Test the Link	94
Chapter 8: SM Problem -> QC/ALM Defect	97
Customizing Service Manager/ServiceCenter for Problem Management	97
Add Fields	
Specify the External Access Definition on Service Manager	
Specify the External Access Definition on ServiceCenter	
Create a Subform	
Add the Subform to a Form	
Add Format Control Calculations/Validations	
Add Rule Set Calculations/Validations	110

Copy an Existing Workflow	111
Associate an Existing Problem Category with the New Workflow	
Create New Rule Set for Initialization and Validation	
Associate the New Workflow with the New Rule Set	
Customizing the QC/ALM Defects Module	116
On QC 10 and Earlier	116
Add Fields	
Add Tabs	
Add Fields to Tabs	
On ALM 11	120
Add Fields	
Configuring Links in QC/ALM Synchronizer	122
Specify Endpoints / Type of Link	122
Define Field Mappings	122
Define Events	123
Test the Link	124
Charter O. OC/ALM Defects CM Decklare	120
Chapter 9: QC/ALM Defect -> SM Problem	126
Customizing Service Manager/ServiceCenter for Problem Management	126
Add Fields	
Specify the External Access Definition on Service Manager	
Specify the External Access Definition on ServiceCenter	
Create a Subform	
Add the Subform to a Form	137
Customizing the QC/ALM Defects Module	139
On QC 10 and Earlier	139
Add Fields	
Add Fields	
Add Fields	
Add Fields Add Tabs Add Fields to Tabs	
Add Fields Add Tabs Add Fields to Tabs Create a View	
Add Fields Add Tabs Add Fields to Tabs Create a View Verify	
Add Fields Add Tabs Add Tabs Add Fields to Tabs Create a View Verify On ALM 11	

Configuring Links in QC/ALM Synchronizer	147
Specify Endpoints / Type of Link	
Define Filters	
Define Field Mappings	148
QC/ALM Field <-> SM Field	151
Constants -> SM Fields	152
Define Events	154
Test the Link	154
Chapter 10: SM Problem <-> QC/ALM Defect	157
Customizing Service Manager/ServiceCenter for Problem Management	
Add Fields	
Specify the External Access Definition on Service Manager	
Specify the External Access Definition on ServiceCenter	
Create a Subform	
Add the Subform to a Form	
Add Format Control Calculations/Validations	169
Add Rule Set Calculations/Validations	
Copy an Existing Workflow	171
Associate an Existing Problem Category with the New Workflow	
Create New Rule Set for Initialization and Validation	
Associate the New Workflow with the New Rule Set	
Customizing the QC/ALM Defects Module	
On QC 10 and Earlier	
Add Fields	
Add Tabs	
Add Fields to Tabs	
Create a View	
Verify	
On ALM 11	
Add Fields	
Create a View	
Verify	
Configuring Links in QC/ALM Synchronizer	
Specify Endpoints / Type of Link	
Define Filters	

Define Field Mappings	187
Define Events	190
Test the Link	191
Chapter 11: Upgrading the Integration	195
Upgrading to the Latest Release	195
Back up Jar Files and Links	195
Deploy the Latest Adapters	195
Upgrade for Various Synchronization Scenarios	196
Upgrade in Non-PD Environment	196
SM Change -> QC/ALM Defect	197
SM Change -> QC/ALM Requirement	198
SM Problem -> QC/ALM Defect	199
QC/ALM Defect -> SM Problem	199
SM Problem <-> QC/ALM Defect	202
Upgrade from Non-PD Environment to PD Content Pack 9.30.2 (for SM 9.3x only) $\ldots$	202
Upgrade from Non-PD Environment to PD Content Pack 9.30.3/SM 9.4x Codeless $\dots$	202
Upgrade from 9.3x with PD Content Pack 9.30.2 to 9.30.3/SM 9.4x Codeless	204
Post-Upgrade Tasks	205
Appendix A: Error Messages	206
Required Fields	207
Installation	208
Configuration	208
Runtime	211
XML Validation	215
Send Documentation Feedback	219

# **Chapter 1: Introduction**

This chapter describes:

- "Document Purpose" below
- "Synchronization Concepts" on the next page
- "References" on page 11

### **Document Purpose**

This document describes how to configure and deploy the integration components

- HP Service Manager / HP ServiceCenter (SM)
- HP Quality Center (QC)
- HP Quality Center Synchronizer (QCS)

**Note:** As of version 11.00, HP Quality Center is known as HP Application Lifecycle Management (ALM); as of version 1.40, HP Quality Center Synchronizer (QCS) is known as HP ALM Synchronizer.

This document also describes how to configure and test synchronization links between QC/ALM and SM.

**Note:** This document contains numerous examples that use the SM and QC/ALM default installation configuration and databases. Your particular configuration may differ significantly. The example synchronization configuration may also differ significantly from your requirements.

The target readers include HP Consultants and/or Application Administrators who must set up and maintain QC/ALM Synchronizer, ensuring that the synchronizer meets all user organization procedural requirements. This document assumes that the reader is an experienced user of either Service Manager or QC/ALM (but not necessarily of both), and therefore describes only the basics of both SM and QC/ALM.

# Synchronization Concepts

This section provides a detailed introduction to the following basic synchronization concepts.

#### • SM Change -> QC/ALM Defect

When a business owner enters a change request in SM and marks the change "Forward as defect", a defect is created in QC/ALM. This informs the QA personnel that they should begin the QA process.

During the QA process, key information is synchronized from QC/ALM to SM. The integration administrator has the responsibility of determining the key information and specifying the information in the field mapping (using the provided integration tool) in order for the business owner to view updated (scheduled) information in SM. The information includes the status of all changes in the testing cycle.

#### • SM Change -> QC/ALM Requirement

The requirement synchronization feature of this integration allows requirements found during the change management process to be systematically tracked by SM and QC.

When a business owner enters a change request in SM and marks it as "Forward as requirement", a requirement is created in QC/ALM. This informs the QA personnel that they should begin the QA process.

During the QA process, key information is synchronized from QC to SM. The integration administrator has the responsibility of determining the key information and specifying the information in the field mapping (using the provided integration tool). This allows the business owner to view updated (scheduled) information in SM.

#### SM Problem -> QC/ALM Defect

After a problem is created, if the CPE engineer determines that there is bug with the problem after analyzing it, and the bug fixing work needs to be tracked, the CPE engineer triggers/initiates the creation of the QC CR ticket. When this problem is marked as "Synchronize with QC Defect", a defect is created in QC/ALM.

#### • QC/ALM Defect -> SM Problem

The business process for defect management in QC/ALM supports creation of known errors in SM based on information in QC/ALM. However, in the current solution, the integration can only create a

problem in SM from a defect in QC/ALM. A user must create the known error in SM manually from the problem in SM. In non-PD environment, known errors are a source of information for informal knowledge articles in the Knowledge Base. In PD environment, known error is a problem record and its "isKnownError" flag is true.

• SM Problem <-> QC/ALM Defect

This user story is a combination of SM Problem -> QC/ALM Defect and QC/ALM Defect -> SM Problem.

### References

- HP Quality Center Synchronizer User's Guide / HP ALM Synchronizer User's Guide
- HP Quality Center Administrator's Guide / HP ALM Administrator's Guide
- HP Service Manager Installation Guide
- HP Service Manager Online Help
- Best Practices for Publishing and Consuming Web Services with ServiceCenter

# Chapter 2: Planning the Deployment

This chapter describes the following deployment plannings:

- "Supported Products and Platforms" below
- "Deployment Scenarios" on the next page
- "Data Types" on the next page
- "Deployment Tasks" on the next page
- "Release Package" on page 14

### Supported Products and Platforms

Supported products are shown in the following table.

Supported Product	Version
HP Quality Center Synchronizer / HP ALM Synchronizer	1.2, 1.3, 1.4, 1.5, and 1.6
HP Quality Center / HP Application Lifecycle Management	9.2 patch 4 and above, 10, 11, 11.5, and 12
HP Service Manager	7.01, 7.02, 7.10, 9.20 patch 1, 9.3x and 9.40
HP ServiceCenter	6.2.2 and above

- In this integration solution, supported platforms for Quality Center Synchronizer or ALM Synchronizer include:
- Microsoft Windows 2000 with Service Pack 4 (32bit)
- Microsoft Windows XP with Service Pack 2 (32bit)
- Microsoft Windows 2003 Server with Service Pack 2 (32bit)

**Note:** When used with ALM 11.00, ALM Synchronizer only supports Microsoft Windows XP with Service Pack 2 (32bit).

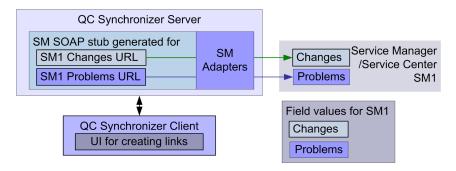
For information about supported platforms of Service Manager, and Quality Center/ALM, refer to their product support matrix.

### **Deployment Scenarios**

The deployment scenarios include:

- A single SM server has a dedicated synchronizer.
- A single SM server can connect to multiple QC/ALM systems.

The scenarios are shown in the following figure:



# Data Types

"Matching Types" on page 31 describes the data type requirements between QC, QCS, and SM.

# Deployment Tasks

Deployment tasks include:

- Customizing Service Manager/Service Center
- Customizing Quality Center/ALM
- Installing/Configuring QC/ALM Synchronizer
- Configuring Links in QC/ALM Synchronizer

# Release Package

The release package is delivered as an executable self-extracting installer. Run the installer by double clicking it. The major contents are shown in the following table.

Directory	Contents	
sm-adapter\adapter	Adapter and dependencies (except stub)	
sm-adapter\ant	Build lib	
sm-adapter\bin	Script to generate the stub	
sm-adapter\doc	Release documents (including this document)	
sm-adapter\jdk5	Sun JDK 1.5	
sm-adapter\lib	Binary libraries required to generate the stub	
sm-adapter\sample	Examples of WSDL and adapter configuration	
sm-adapter\out-of-box	Out-of-box demo package	

**Caution:** The out-of-box demo package does not support Service Manager 9.20 or later, ALM 11, or ALM Synchronizer 1.4.

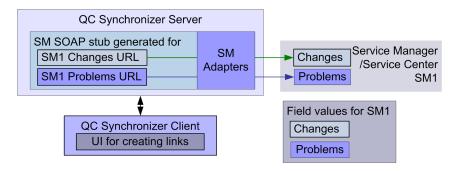
# Chapter 3: Installing and Configuring QC/ALM Synchronizer

The QC Synchronizer (QCS)/ALM Synchronizer allows centralized management of a set of tightly coupled one-to-one data synchronization links and provides an open and extensible platform for the development of new data synchronization adapters to entity repositories.

To install and configure the synchronizer, follow these steps:

- 1. "Installing QC/ALM Synchronizer" below
- 2. "Deploying the Adapters" on the next page
- 3. "Generating/Deploying the Stub" on page 17
- 4. "Copying SM Adapter Configuration Files" on page 17
- 5. "Editing the Configuration Files" on page 18

The following diagram provides an overview of the installation process.



# Installing QC/ALM Synchronizer

Download QC Synchronizer from:

http://updates.merc-int.com/qualitycenter/qc90/sync/qcsynchronizer/index.html

Download ALM Synchronizer 1.4 from:

http://update.external.hp.com/qualitycenter/qc110/sync/almsynchronizer/index.html

**Caution:** When installing the ALM Synchronizer, select the ALM 11 mode so that it supports synchronization with ALM 11.

### Installation Requirements

The following requirements apply for all supported versions of Service Manager / ServiceCenter, QC / ALM, and QCS / ALM Synchronizer:

- The QC/ALM client is installed when you are logging into QC/ALM. The correct QC/ALM client should be installed on the QC/ALM Synchronizer server.
- The QC/ALM Synchronizer machine should have the same time zone with the QC/ALM machine. For more information, see the integration solution release notes.
   For installation instructions, see the QC/ALM Synchronizer user guide. You can get the user guide from the installation package or from http://h20230.www2.hp.com/selfsolve/manuals.
- Make sure that the time difference in UTC between SM and QC/ALM Synchronizer is within 5 minutes, otherwise the data might be lost during synchronization. For example, the SM server time is 2008-1-1 21:00:00 in UTC, then the QCS server time must be between 2008-1-1 20:55:00 and 2008-1-1 21:05:00.

# Deploying the Adapters

Copy all files under the <release-package>\adapter directory to the <QCS\_Install\_ Dir>\adapters\lib directory. Adapters include:

sm-adapter-XX.XX.XXX.jar (XX.XX.XXX is the version number for the current release)

sm-adapter-axis-1.4.jar

```
sm-adapter-commons-discovery-0.2.jar
```

```
sm-adapter-commons-lang-2.3.jar
```

sm-adapter-jaxrpc-1.1.jar

sm-adapter-jdom-1.1.jar

sm-adapter-saaj-1.2.jar

sm-adapter-wsdl4j-1.5.1.jar

sm-adapter-commons-codec-1.3.jar

sm-adapter-commons-httpclient-3.1.jar

# Generating/Deploying the Stub

To generate and deploy the stub:

- 1. Start the SM service (stub generation requires access to SM).
- 2. Edit the following lines in <release-package>\bin\build.properties as required for access to SM:

#Set up WSDL URL, please change the URL to your actual SM server, eg, http://<your-server>:<port>/.../<service-name>.wsdl

#Comment this line by this sign "#" if you do not generate stub jar for change management module sm.change.wsdl=http://localhost:13080/sc62server/PWS/ QCIntChangeService.wsdl

#Comment this line by this sign "#" if you do not generate stub jar for problem management module sm.problem.wsdl=http://localhost:13080/sc62server/PWS/ QCIntProblemService.wsdl

- Run the script build.bat from the operating system's command prompt (check the console output for errors). The stub <Release\_Package>\build\sm-adapter-ws-client.jar is generated.
- 4. Copy the stub to the <QCS\_Install\_Dir>\adapters\lib directory.

# **Copying SM Adapter Configuration Files**

To copy SM adapter configuration files:

1. Start/restart QCS.

Click Start > All Programs > HP Quality Center Synchronizer > Start/Stop Synchronizer. The directories <QCS\_Install\_Dir>\adapters\dat\SM ChangeManagement and SM ProblemManagement appear after the synchronizer service is started (this can take up to one minute).

C:\Program Files\HP\Quality Center Synchronizer\adapters\dat			
Address 🛅 C:\Program Files\HP\Quality Center Synchronizer\adapters\dat			
ClearQuest			
C			
RequisitePro			
SM ChangeManagement			
🚞 SM ProblemManagement			

- 2. Copy the file <Release\_Package>\sample\configuration\_file\_default.xml to the following
  folders:
  - SM ChangeManagement
  - SM ProblemManagement

# Editing the Configuration Files

Edit the files as described below. The files will be specified later when you create links.

### Module Types

There are two module types for this configuration file: change or problem.

A module named change or problem means that this module is for Change Management or Problem Management, respectively.

For example:

<itg:module name="change">

You can define one module or two in this file, but duplicate definitions are not permitted.

### **Field Types**

#### **Field XML Element Specification**

Parameter	Description
name	Field name. This name should be the same as <b>Caption</b> enabled in the SM/SC WSDL. This field is required.

Parameter	Description
type	Field type. Its value can be String/Number/Date/Single_Value_List/Multi_Value_ List. This field is required.
readonly	Indicates whether the field is read-only. Its value can be true or false. This field is optional. Default is false.
required	Indicates whether the field is mandatory, recommended or optional. This field is optional. Default value for the field is optional.
length	The length of the field in the SM endpoint. This field is optional. The length is unlimited if not specified.

#### Field XML Element Specification, continued

The configuration file is an XML file that provides Change/Problem field values to the SM adapter. These values include:

- Field name (the caption of a field in the SM WSDL configuration form, such as Status, Priority)
- Field types
  - String
  - Number
  - Date
  - Single\_Value\_List
  - Multi\_Value\_List
- List types
  - Array (multi-value list)
  - Single-value list
- For a value list, the mapping of the value in the database and the exposed caption (for a type other than a value list type, the adapter automatically determines the desired data type).

### Requirements

- Default field configuration is readable and writable with unlimited length.
- Default field configuration for a Single\_Value\_List or a Multi\_Value\_List must be explicitly specified.

- Read-only fields must be explicitly specified.
- You must specify the type and read/write explicitly only for a Single\_Value\_List/Multi\_Value\_List.
- If a field is not configured, the field is read/write with unlimited length.
- A list or multi-list field may contain item elements. For each item specify the value and display text in the form <itg:item value="\$value">\$display text</itg:item>.
- If the field in WSDL is an Array, it must be mapped to Multi\_Value\_List.
- If the QC field is User\_List, you can only specify String or Single\_Value\_List for the corresponding SM field.
- If the field is read-only you must set the attribute readonly as true.
- If the field attribute required is mandatory, the field is mandatory for creation of a new entity.
- If the field has a length limitation (attribute length), values from other endpoints could be truncated to match this limitation.

### SM Field Type and Definition Rule

There are restrictions on data type and field type combination. Define field type in the configuration file according to such rules.

SM 7.0x/7.10 DB data type	SC 6.2 DB data type	Field type on Form	WSDL data type	Field Definition Rule
Date/time	Date/time	Date	DateTimeType	Optional. Permitted type is "Date".
Number	Decimal	Decimal Text	< <i>Empty</i> > DecimalType IntType	Optional. Permitted type is "Number".
Logical	Boolean	Check Box Radio Button	< <i>Empty</i> > BooleanType	Optional. Permitted type is "String".
Character	Text	Text TextArea Combo Box Comfill	< <i>Empty</i> > StringType	Optional. Permitted types include "String" and "Single_Value_List".

#### SM Field Type and Definition Rules

SM 7.0x/7.10 DB data type	SC 6.2 DB data type	Field type on Form	WSDL data type	Field Definition Rule
Array	Array	Text Area	<empty></empty>	Optional. Permitted types include "String" and "Single_Value_List".
Array	Array	Combo Box Comfill Text	<empty></empty>	Required. Permitted type is "Multi_ Value_List".

SM Field Type and Definition Rules, continued

### SM Change Management Example

The following is typical for SM change management.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<itg:mapping xmlns:itg="http://www.hp.com/smci/SMQCIntegration/config">
```

<itg:module name="change">

```
<itg:field name="Urgency" type="Single_Value_List" readonly="false"
required="mandatory" length="50">
```

```
<itg:items>
<itg:item value="1">1 - Critical</itg:item>
<itg:item value="2">2 - High</itg:item>
<itg:item value="3">3 - Average</itg:item>
</itg:items>
</itg:field>
</itg:module>
</itg:mapping>
```

### SM Problem Management Example

The following is the included configuration\_file\_default.xml for SM problem management.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<itg:mapping xmlns:itg="http://www.hp.com/smci/SMQCIntegration/config">
```

```
<itg:module name="problem">
```

```
<itg:field name="Status" type="Single_Value_List" required="mandatory">
```

<itg:items>

<itg:item value="Open">Open</itg:item>

<itg:item value="Accepted">Accepted</itg:item>

<itg:item value="Work In Progress">Work In Progress</itg:item>

<itg:item value="Pending Vendor">Pending Vendor</itg:item>

<itg:item value="Pending User">Pending User</itg:item>

<itg:item value="Rejected">Rejected</itg:item>

<itg:item value="Deferred">Deferred</itg:item>

```
</itg:items>
```

</itg:field>

```
<itg:field name="AssignmentGroup" type="Single_Value_List" required="mandatory">
```

<itg:items>

<itg:item value="Application">Application</itg:item>

<itg:item value="Network">Network</itg:item>

</itg:items>

```
</itg:field>
```

```
<itg:field name="Service" type="Single_Value_List" required="mandatory">
```

<itg:items>

<itg:item value="Applications">Applications</itg:item>

<itg:item value="Service Management">Service Management</itg:item>

</itg:items>

```
</itg:field>
```

```
<itg:field name="Title" type="String" required="mandatory" length="50"/>
```

<itg:field name="Description" type="String" required="mandatory"/>

```
<itg:field name="Area" type="Single_Value_List" required="mandatory">
```

<itg:items>

```
<itg:item value="data">data</itg:item>
```

</itg:items>

</itg:field>

```
<itg:field name="Subarea" type="Single_Value_List" required="mandatory">
```

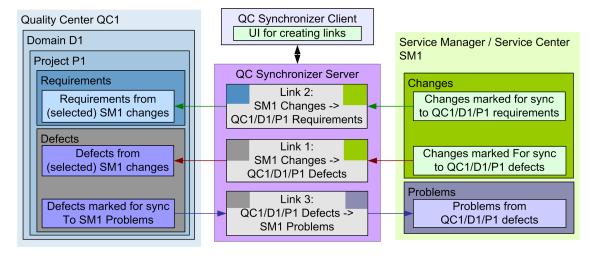
<itg:items>

```
<itg:item value="data or file corrupted">data or file
corrupted</itg:item>
   </itg:items>
</itg:field>
    <itg:field name="Impact" type="Single_Value_List" readonly="false"</pre>
                   required="mandatory">
  <itg:items>
    <itg:item value="1">1 - Enterprise</itg:item>
    <itg:item value="2">2 - Site/Dept</itg:item>
    <itg:item value="3">3 - Multiple Users</itg:item>
    <itg:item value="4">4 - User</itg:item>
  </itg:items>
</itg:field>
<itg:field name="Urgency" type="Single Value List" readonly="false"
                   required="mandatory">
  <itg:items>
    <itg:item value="1">1 - Critical</itg:item>
    <itg:item value="2">2 - High</itg:item>
    <itg:item value="3">3 - Average</itg:item>
    <itg:item value="4">4 - Low</itg:item>
  </itg:items>
</itg:field>
</itg:module>
</itg:mapping>
```

# Chapter 4: Configuring Links in QC/ALM Synchronizer

This chapter describes how to configure and test links. Although instructions in this chapter are provided as examples using QC Synchronizer and Quality Center, they still apply for ALM Synchronizer and ALM.

The following diagram summarizes link configuration:



You need to create synchronization links in QC/ALM Synchronizer between two endpoints. Each endpoint is an application or system containing data that is synchronized by the synchronizer. A link defines which entities are included in the synchronization, and how the synchronization is performed.

This chapter describes aspects of link creation that are common to all three types of links.

- "Create a Link" on the next page
- "QC/ALM Field <-> SM Field Mappings" on page 29
- "List Value Mappings" on page 35
- "Constant -> SM Field Mappings" on page 35

Filters are only required for QC/ALM Defect -> SM Problem (see "Define Filters" on page 148). The events settings determine what the synchronizer does in response to specified events. Events must be specified for all three link types.

# Create a Link

The following table summarizes the properties required in the wizard. Have this data available before starting the wizard.

**Note:** A link cannot be duplicated. For example, if a link already exists for SMServer1/Changes -> QCServer1/Domain1/Project1/Defects, a second link between these two entities cannot be created.

End Point	Parameter	Requirements
QC	Username	
QC	Password	
QC	Server URL	
QC	Domain	
QC	Project	
SM	User name	
SM	Password	
SM	Service URL	<pre>http://<service_manager_host>: &lt; port &gt;/sc62server/PWS/QCIntChangeService.wsdl or http://<service_manager_host>: <port>/sc62server/PWS/QCIntProblemService.wsdl</port></service_manager_host></service_manager_host></pre>
SM	Adapter Configuration (SM field values) filename	Empty or the adapter data folder file (see "Copying SM Adapter Configuration Files" on page 17).
SM	QCProject	Required (because of an adapter limitation). The format is <qc_host>/<qc_domain>/<qc_project></qc_project></qc_domain></qc_host>

#### To create a link:

- 1. Click **Link/Create**. The "Step 1: Assign general properties" dialog appears.
- 2. Enter the required information (the following example is for SM Change -> QC Defect).

🧝 Create Link - Step 1 of 4 - General Properties						
	Assign gener	ral properties:				
	Link name:	change_to_defect_1				
(IP)	Description:	change to defect 1				
Quality Center Synchronizer	Endpoint 1 type:	Quality Center				
Synchronizer	Endpoint 2 type:	SM ChangeManagement 💌				

- 3. Click Next. The "Step 2: Assign QC endpoint connection properties" dialog appears.
- 4. Enter the required information.

雅 Create Link - Step 2	of 4 - Quality Center Endpoi	nt 💌
STATISTICS STATISTICS		
Ø		nter endpoint connection properties:
Quality Center	User name: SMQCIntUse	er
Synchronizer	Password:	
	Parameter	Value
	ServerURL	http://localhost:8080/gcbin
	Domain	DEFAULT
	Project	Demo
		Chec <u>k</u> Connectivity

- 5. Click **Next**. One of the following appears:
  - "Step 3: Assign SM ChangeManagement endpoint connection properties"
  - "Step 3: Assign SM ProblemManagement endpoint connection properties"
- 6. Enter the required information (the following example is for SM Change -> QC Defect).

🧝 Create Link - Step 2	of 4 - Quality Center Endpoint	×
STATISTICS STATISTICS		
Quality Center Synchronizer	Assign SM ChangeMa User name: SMQCIntUser Password: ••••••	nagement endpoint connection prop
	Parameter	Value
	Service URL	http://localhost:13080/sc62server/PWS/QCIntChangeService.ws
	Configuration File Name	configuration_file_default.xml
	QC Project	localhost/DEFAULT/Demo
		Chec <u>k</u> Connectivity

Note: QC Project has the same value as specified on SM customization.

- 7. Click **Next**. If this is a change management link, "Step 4: Select entity types" dialog appears.
- 8. Select one of the following:
  - Change as Defect
  - Change as Requirement



9. Click **Save**. The link is created.

10. Modify required settings on the **Connectivity** tab.

:			SM ChangeMar	nagement		
Jser name:	SMQCIntUser		User name:	SMQCIntUser		
assword:			Password:	•••••		
Parameter		Value	Parameter		Value	
Domain		DEFAULT	Configuration	File Name	configuration_file_default.xml	
Project		Demo	QC Project		localhost/DEFAULT/Demo	
ServerUBL http://		http://localhost:8080/gcbin	Service URL	September 1	http://localhost:13080/sc62serv	

### **Advanced Parameters**

Advanced parameters are shown on the Advanced tab.

Parameter	Value
Socket Timeout (Minutes)	2
Retries On Locked Record	0
Retry Interval (Seconds)	10

• Retries On Locked Record When a record in the SM endpoint is locked, it will cause synchronization failure. The integration will retry the synchronization according to the value of this parameter. Ø means the retry feature is

disabled. For details, see *HP Defects and Requirements Exchange with HP Service Manager and HP Quality Center Release Notes*.

• Retry Interval (Seconds)

When the retry feature is enabled, this parameter defines the retry interval. The retry interval must be an integer between 1 and 10. For details, see *HP Defects and Requirements Exchange with HP Service Manager and HP Quality Center Release Notes*.

• Socket Timeout (Minutes)

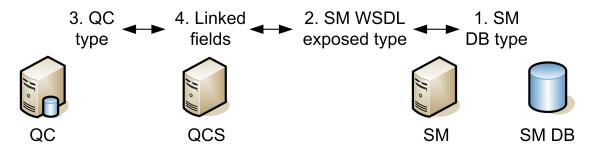
Socket connection will be established during synchronization. If there are many records matching the filter in the SM endpoint, retrieving list operation will cost some time, which might cause timeout of socket connection. This parameter is used to define the socket timeout. Its range is: 0-120.

# QC/ALM Field <-> SM Field Mappings

This section describes how to map QC/ALM fields and SM fields.

- "Creating Mappings" below
- "General Mapping Requirements" on the next page
- "Matching Types" on page 31

The following diagram shows the field mapping chain.



For examples of field mappings, see:

- SM Change to QC/ALM Defect, " QC/ALM Field <-> SM Field" on page 77.
- SM Change to QC/ALM Requirement, "QC/ALM Field <-> SM Field" on page 93.
- QC/ALM Defect to SM Problem, "QC/ALM Field <-> SM Field" on page 151.

### **Creating Mappings**

To map fields:

- 1. Select a field on each side.
- 2. Select a direction in the Map Selected Fields tab.

P Map Selected Fields	-								
Create bidirection			M Hai R2 (25) 4 Add Constant Value SM ChangeManagement Change as Defect Schema						
Map QC field to S		_							
+ Map SM Changel	anagement field to	OC field	apped		Name		Туре	Attributes	Mapped
a r nony	origio raiso				Chang	eNumber	String	RW/	No
Problem ID	String	RW	No		Pa Descri	ption	String	RW/	No
Project	Single value	list RW	No		Nodifie	ed	Date	RW	No
Reproducible	Single value	list RW	No		CEnt	iyiD	Number	RW/	No
Severity	Single value	list RW	No		CPro	ject	String	RW/	No
Status	Single value	list RW	No	1	Urgena	cy.	Single value	list RW	No
Summary	String	RW/	No	•					
× 🎸 Check Field Map	oping 🕘 Import (	∋ Export				Mapping Properties	Value Mapp	ing   Field Prop	verties
Aapped Fields						Misc			
Type QC Field Direction		SM ChangeMa Field	nageme	int	Direction Dominant side		<> Endpoint 1		
Summary		>	Description			Synchronize bac	k on create 1	No	

### **General Mapping Requirements**

When creating field links, keep the following limitations in mind:

- If you change the mappings you must do a full synchronization to ensure synchronization of historical data. Otherwise, your historical data cannot be synchronized correctly and you may get errors in the next incremental synchronization.
- A field in one endpoint can be mapped to only one field in the other endpoint.
- Mandatory fields must be mapped. If a null value is written to a mandatory field, an error will occur at runtime.
- If you map string fields with different maximum lengths, during synchronization a string value in the source endpoint will be truncated as necessary if it exceeds the maximum length of the other field.

### Matching Types

The following table lists allowed data type combinations. Highlighted entries are demonstrated in examples in this document.

#### Data Type Combinations

QC Data type	QCS QC Type	Dir	QCS SM type	WSDL data type	Field type on form in SM/SC	SM DB data type for SM	SM DB data type for SC
Number	Number	<->	Number	DecimalType or IntType <sup>1</sup>	Decimal or Text	Number	Decimal
String	String	<->	String	BooleanType	Check Box or Radio Button	Logical	Boolean
Date <sup>2</sup>	Date	<->	Date	DateTimeType (required)	Date	Date/time	Date/time
String	String	<->	String <sup>3</sup>	StringType	Text, TextArea, Combo Box or Comfill	Character	Text
Memo	Memo/ String	<->	String <sup>4</sup>	StringType	Text, TextArea, Combo Box or Comfill	Character	Text
User List <sup>5</sup>	User List	->	String <sup>6</sup>	StringType	Text, TextArea, Combo Box or Comfill	Character	Text

#### Data Type Combinations, continued

QC Data type	QCS QC Type	Dir	QCS SM type	WSDL data type	Field type on form in SM/SC	SM DB data type for SM	SM DB data type for SC
Lookup List	Single- value list	<->	Single- value list/ String <sup>7</sup>	StringType	Text, TextArea, Combo Box or Comfill	Character	Text
String	String	<->	String	StringType	TextArea	Array <sup>8</sup>	Array <sup>9</sup>
Memo	Memo String	<->	String	StringType	TextArea	Array <sup>10</sup>	Array <sup>11</sup>
Lookup List	Single- value List	<->	String	StringType	TextArea	Array <sup>12</sup>	Array <sup>13</sup>
User List <sup>14</sup>	User List	<->	String	StringType	TextArea	Array <sup>15</sup>	Array <sup>16</sup>
Lookup List	Multi- value List	<->	Multi- value List	StringType	Text, Comfill or Combo Box	Array <sup>17</sup>	Array <sup>18</sup>
Attachment <sup>19</sup>	Attachment	<->	Attachment	Attachment	Attachment	Image	lmage

**Note:** There is no need to explicitly specify WSDL data type on WSDL configuration for all types except for the Date type. For details, see *Best Practices for Publishing and Consuming Web Services with ServiceCenter*.

<sup>1</sup>IntType supports a data range from -2,147,483,648 to 2,147,483,647.

<sup>2</sup>QC data only supports Yr/Mo/Dt.

<sup>3</sup>It is recommended to leave this field blank. Otherwise "Invalid byte 2 of 3-byte UTF-8 sequence" might occur if certain I18N characters are synchronized.

<sup>4</sup>See footnote 3.

Installation and Administration Guide Chapter 4: Configuring Links in QC/ALM Synchronizer

<sup>5</sup>Write to the QC field User\_List only if SM has exactly the same users (including logins, names, etc.) as QC. An incorrect entry can cause serious problems in QC. You can read from QC User\_List field and write to SM String type field only if the field in SM is NOT a field with SM logins.

<sup>6</sup>See footnote 3.

<sup>7</sup>See footnote 3.

<sup>8</sup>Only an array of characters is supported.

<sup>9</sup>See footnote 8.

<sup>10</sup>See footnote 8.

<sup>11</sup>See footnote 8.

<sup>12</sup>See footnote 8.

<sup>13</sup>See footnote 8.

<sup>14</sup>See footnote 5.

<sup>15</sup>See footnote 8.

<sup>16</sup>See footnote 8.

<sup>17</sup>See footnote 8.

<sup>18</sup>See footnote 8.

<sup>19</sup> The SM-QC/ALM integration does not support synchronization of an attachment with OKB size. Consider the following guidelines when mapping attachment fields:

- You can create only one mapping between attachment fields per link.
- Synchronizer identifies attachments by their file name, and not by their content. Therefore:
  - If you change the file name of an attachment, even if you do not change its content, Synchronizer determines that the original attachment has been deleted and a new attachment added, and synchronizes the attachment fields accordingly.
  - If you have different attachments in each of the endpoints, but they have the same file name, Synchronizer is not able to distinguish between them and considers them as the same attachment.
- For a bidirectional attachment field mapping, if an attachment was updated in both endpoints since the last synchronization, Synchronizer copies the attachment in the non-dominant endpoint to the conflict\_backup directory, located under the main HP ALM Synchronizer directory. It then overwrites the attachment in the non-dominant endpoint with the attachment in the dominant endpoint.

# List Value Mappings

This section describes how to map values for multi-valued lists. QCS does not have access to the values of SM multi-values lists, and therefore the values must be specified in an XML file.

Some list fields also require mapping of available values (as shown in the following figure).

Map Selected Fields					Ha Ha Ra Co SM ChangeManagen			ema	
Name	Туре	Attributes	Mapped	-	Name	Ty	pe	Attributes	Mapped
Priority	Single value list	RW/	No		ChangeNumber	St	ing	RW	No
Problem ID	String	RW/	No		Description	St	ing	RW	No
Project	Single value list	RW	No		Modified	D-	xe.	RW	No
Reproducible	Single value list	RW/	No		CEntityID	No	mber	RW	No
Seventy	Single value list	RW	No		CProject	St	ing	RW	No
Status Status	Single value list	RW/	No		Urgency	Si	ngle value list	RW/	No
Summary	String	RW/	No	-					
apped Fields				м	apping properties	Value ma	pping Fie	ld propert	ies
🗇 Sevenity	<>	Urgeno	y .	E	Endpoint 1 value	Direction	Endpoint	2 value	
				5	Urgent	<>	1 - Critical		
				4	Very High	<>	2 · High		
				2	LEals	<>	3 - Averag	10	
				1 3	High	()	3 - WALLO	<b>,</b> c	

# Constant -> SM Field Mappings

For examples of constant -> SM field mappings, see "Constants -> SM Fields" on page 152.

The following figure shows an example of constant -> SM field mapping.

eneral	Connectivity Scheduling	Filters   Events	Field Mapping
appe	d Fields		
Туре	Endpoint 1 Field	Direction	Endpoint 2 Field
8	Value: AUTO	>	AssignmentGroup
8	Value: client system	>	Category
8	Value: BOB.HELPDESK	>	ProblemOwner

# **Chapter 5: Integration Account**

This chapter includes:

- "Creating an SM Integration Account" below
- "Creating a QC/ALM Integration Account" on page 42

### **Creating an SM Integration Account**

The integration account is equivalent to an operator in Service Manager for exclusive use with this solution.

This section includes:

- "Create a Contact Record" below
- "Create a Profile Record" below
- "Create an Operator Record" on page 40

### Create a Contact Record

Create a contact for the integration administrator by clicking **System Administration**> **Base System Configuration** > **Contacts** in Service Manager; or clicking **Support** > **Contacts** in ServiceCenter.

Page	Field	Value
Contact Information	Contact Name	<administrator's name=""></administrator's>
Contact Information	Full Name	<administrator's full="" name=""></administrator's>

### Create a Profile Record

To create a profile record for the integration account:

• Create profile for Change Management to synchronize SM Changes with QC/ALM Requirements and Defects ("Change -> Requirement" and "Change -> Defect").

Profile records grant specific rights and privileges to the integration account to enable Change Management.

#### **On Service Manager:**

Click **System Administration > Ongoing Maintenance > Profiles**and create a Change management profile record by with the parameters shown in the following table.

No	Tab Page	Field	Value	Comment
1		Profile Name	CMProfile_QCInt	
2		Profile Area	Changes	
3	Security/Rights	Update	Always	
4	Security/Rights	View	Yes	Check Box
5	Security/Rights	Reopen	Yes	Check Box
6	Query	Query Options	Yes	Check Box

#### **On ServiceCenter:**

Click **Services** > **Change Management** > **Maintenance** > **Profiles** and create a change management profile record with the parameters shown in the following table.

No	Tab Page	Field	Value	Comment
1		Profile Name	CMProfile_QCInt	
2		Profile Area	Changes	
3	Basic/Basic Options	Open	Yes	Check Box
4	Basic/Basic Options	Reopen	Yes	Check Box
5	Basic/Basic Options	Save	Yes	Check Box
6	Query/Query Options	All	Yes	Check Box

Create profile for Problem Management to synchronize SM Problem with ALM/QC Defect ("Problem <-> Defect", "Problem -> Defect" and "Problem <- Defect").</li>

Profile records grant specific rights and privileges to the integration account to enable Problem

### Management.

### **On Service Manager:**

Click **System Administration > Ongoing Maintenance > Profiles**and create a change management profile record with the parameters shown in the following table.

No	Tab Page	Field	Value	Memo	Remarks
1		Profile Name	PMProfile_ QCInt		
2	Problems/Security/Rights	New	Yes	Check Box	This parameter is not required when synchronizing SM Problems with QC/ALM Defects ("Problem - > Defect").
3	Problems/Security/Rights	Close	Yes	Check Box	
4	Problems/Security/Rights	Update	Always		
5	Problems/Security/Rights	Reopen	Yes	Check Box	

### **On ServiceCenter:**

Click **Services** > **Problem Management** > **Administration** > **User Profiles**and create a problem management profile record with the parameters shown in the following table.

No	Tab Page	Field	Value	Memo	Remarks
1		Profile Name	PMProfile_ QCInt		
2	Problem Details	Browse	Yes	Check Box	
3	Problem Details	Open	Yes	Check Box	This parameter is not required when synchronizing SM Problems with QC/ALM Defects ("Problem -> Defect").

No	Tab Page	Field	Value	Memo	Remarks
4	Problem Details	Update	Yes	Check Box	
5	Problem Details	Reopen	Yes	Check Box	

# Create a SecRole Record

**Note:** Steps in this topic are applicable for the Service Manager Process Designer (PD) Content Pack 9.30.x only. Skip this topic if the PD Content Pack is not installed.

You can create a SecRole Record for Change Management to synchronize SM Changes with QC/ALM Requirements and Defects ("Change -> Requirement" and "Change -> Defect"). The SecRole records grant specific rights and privileges to the integration account to enable Change Management.

To create a SecRole record for the integration account:

- 1. Click System Administration > Security > Roles.
- 2. Create a role named SMQCIntSecRole with the parameters shown in the following table. Remove all rights from all areas with the exception of the Change area and the Problem area.

No	Tab Page	Field	Value	Comment
1		Security Role Name	SMQCIntSecRole	
2		Security Area	Change	
3	Rights	View	Yes	Check Box
4		Update	Always	
		Security Area	Problem	
5	Rights	View	Yes	Check Box
6		New	Yes	Check Box
7		Update	Always	

# Create an Operator Record

The operator record identifies the logon name, password, and other settings for each SM operator. Create the required operator records by clicking **System Administration** > **Ongoing Maintenance** > **Operators** on Service Manager; or clicking **Utilities** > **Administration** > **Security** > **User Administration** > **Search for Operators** on ServiceCenter with the parameters shown in the following table.

No	Page	Field	Value	Remarks
1	General	Logon Name	SMQCIntUser	
2	General	Full Name	QC Integration Default Account	
3	General	Contact ID	<integration administrator's account in SM&gt;</integration 	The contact created in the previous section.
4	Security	Unlimited Sessions	Yes	Check Box
5	Security	Password	<your password=""></your>	
6	Startup	Execute Capabilities	SOAP API	
7	Login Profile	Time Zone	Greenwich/Universal (or create a time zone with no time difference or DST switch in Database Manager)	
8	Login Profile	Date Format	yy/mm/dd	The date format cannot be changed (changing it will cause loss of all data during synchronization).
9	Startup	Execute Capabilities	ChMAdmin	Set the two parameters to synchronize SM Changes with QC/ALM Requirements and
10	General/Application Profiles	Change Profiles	CMProfile_QCInt	Defects ("Change -> Requirement" and "Change -> Defect").

#### **Operator Record Parameters**

No	Page	Field	Value	Remarks
11	Startup	Execute Capabilities	ProbAdmin	Set the two parameters to synchronize SM Problem with ALM/QC Defect ("Problem <->
12	General/Application Profiles	Problem Profile	PMProfile_QCInt	Defect", "Problem -> Defect" and "Problem <- Defect").

### **Operator Record Parameters, continued**

**Note:** If the Service Manager Process Designer (PD) Content Pack 9.30.x is installed, refer to the parameters shown in the following table for the General/Application Profiles configuration.

### Operator Record Parameters

No	Page	Field	Value	Remarks
10	General/Application Profiles	Security Role	SMQCIntSecRole	Set the this parameter to synchronize SM Changes with ALM/QC Requirements and Defects ("Change -> Defect" and "Change -> Requirement").
12	General/Application Profiles	Security Role	SMQCIntSecRole	Set this parameter to synchronize SM Problems with ALM/QC Defects ( "Problem <-> Defect", "Problem -> Defect"and "Defect -> Problem").

# Creating a QC/ALM Integration Account

To create an integration account:

#### 1. Create a User.

Perform the following steps:

- a. Log in to the "Quality Center Site Administration" or "Application Lifecycle Management Site Administration" using the QC/ALM site administrator account.
- b. On the Site Users tab, create and configure integration account SMQCIntUser (including the User Name and password).
- c. In the Site Projects tab, choose the project from the list.
- d. Click the Project Users tab in the right panel, and click Add From The Users List.
- e. Add the configured user SMQCIntUser to the project.
- f. Log off.

#### 2. Create a Group.

Perform the following steps:

### On Quality Center 10 and Earlier:

- a. Log on to the QC project using a project administrator account.
- b. Click **TOOLS > Customize...**.
- c. Select Groups.
- d. Click New.
- e. Enter name SMIntegration.
- f. For Create As: select Viewer.
- g. Click **OK**.
- h. Select **Yes** to create the user group.

### On ALM 11:

- a. Log on to the ALM project using a project administrator account.
- b. Click **TOOLS** > **Customize...**.
- c. Select Groups and Permissions.
- d. Click New Group.
- e. Click **Yes** to create the new group.
- f. Enter name SMIntegration.

- g. For Set As: select Viewer.
- h. Click **OK**.
- 3. Assign Permissions.

Perform the following steps:

a. Click the SMIntegration group, go to the Permissions tab, and assign permissions for the user group on the Requirements, Defects, and Administration subtabs as shown in the following tables.

• Change -> Requirement

Subtab	Permission	QC Setting	ALM Setting
	Add Requirement	Add Requirement	Select the following options:
	Modify Requirement	Modify Requirement	Coverage Level
	Moully Requirement	Delete Requirement	Create
		Add Tests To Coverage	• Create
		Remove Tests From Coverage	• Update
		Add Requirement Traceability	
		Modify Requirement Traceability	Requirement
		Remove Requirement Traceability	Create
		🛨 🔲 Risk-Based Quality Management	
			• Update
			Risk-Based Quality Management
			Assess Business Criticality
			Assess Failure Probability
			Assess Functional Complexity

• Change -> Defect

Subtab	Permission	QC Setting	ALM Setting
--------	------------	------------	-------------

Defects	Add Defects/ Modify Defects	✓       Add Defect         ●       ✓       Modify Defect         □       Delete Defect         □       Add Defect Link         ●       □       Modify Defect Link         ●       □       Remove Defect Link	<ul> <li>Select the following options:</li> <li>Defect <ul> <li>Create</li> <li>Update</li> </ul> </li> </ul>
---------	--------------------------------	--	---

• Problem -> Defect

Subtab	Permission	QC Setting	ALM Setting
Defects	Add Defects/ Modify Defects	<ul> <li>Add Defect</li> <li>Modify Defect</li> <li>Delete Defect</li> <li>Add Defect Link</li> <li>Modify Defect Link</li> <li>Remove Defect Link</li> </ul>	Select the following options: <ul> <li>Defect</li> <li>Create</li> <li>Update</li> </ul>

• Problem <- Defect

Subtab	Permission	QC Setting	ALM Setting
Defects	Add Defects/ Modify Defects	Add Defect         Modify Defect         Delete Defect         Add Defect Link         Modify Defect Link         Remove Defect Link	Select the following options: • Defect • Create • Update

Administration	Add Public	Add Public Favorite Views	Select the following options:
	Favorite Views	Modify Public Favorite Views	
		Delete Public Favorite Views	Add Public Favorite View Folders
	Modify Public Favorite Views	Add Private Favorite Views	Add Public Favorite Views
	Favorite views	Modify Private Favorite Views	
	Delete Public	🔽 Delete Private Favorite Views	• Delete Public Favorite View Folders
	Favorite Views		Delete Public Favorite Views
	Add Private		Manage Private Favorite Views
	Favorite Views		_
	Modify Private		Manage Project Planning and Tracking
	Favorite Views		Modify Public Favorite View Folders
	Delete Private		Modify Public Enverite Views
	Favorite Views		Modify Public Favorite Views

• Problem <- Defect

		QC setting	ALM setting
Defects	Add Defects/ Modify Defects	<ul> <li>Add Defect</li> <li>Modify Defect</li> <li>Delete Defect</li> <li>Add Defect Link</li> <li>Modify Defect Link</li> <li>Remove Defect Link</li> </ul>	Select the following options: <ul> <li>Defect</li> <li>Create</li> <li>Update</li> </ul>

Administration	Add Public	Add Public Favorite Views	Select the following options:
	Favorite View	Modify Public Favorite Views	
	Madifu Dublia	Delete Public Favorite Views	Add Public Favorite View Folders
	Modify Public Favorite Views	Add Private Favorite Views	Add Public Favorite Views
	ravorite views	Modify Private Favorite Views	
	Delete Public	🔽 Delete Private Favorite Views	Delete Public Favorite View Folders
	Favorite Views		Delete Public Favorite Views
	Add Private		Manage Private Favorite Views
	Favorite Views		• Hundge Hivate Fuvorite Views
	Modify Private		Manage Project Planning and Tracking
	Favorite Views		Modify Public Favorite View Folders
	Delete Private		Modify Public Favorite Views
	Favorite Views		

- b. Add the integration user SMQCIntUser to group SMIntegration.
- c. Save and close. The integration account is created.

# Chapter 6: SM Change -> QC/ALM Defect

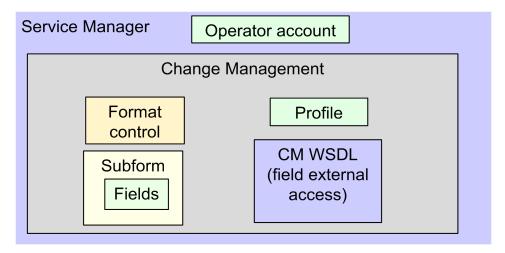
This chapter describes how to synchronize SM Changes with QC/ALM Defects ("Change -> Defect").

This chapter includes:

- "Customizing Service Manager/ServiceCenter for Change Management" below
- "Customizing the QC/ALM Defects Module" on page 66
- "Configuring Links in QC/ALM Synchronizer" on page 75

# Customizing Service Manager/ServiceCenter for Change Management

The following diagram summarizes the components which require tailoring in Service Manager/ServiceCenter.



To customize Service Manager/ServiceCenter for Change Management, perform the following tasks:

- 1. "Add Fields" on the next page
- 2. "Specify the External Access Definition on Service Manager" on the next page or "Specify the External Access Definition on ServiceCenter" on page 52
- 3. "Create a Subform" on page 53

- 4. "Add the Subform to a Form" on page 55
- 5. "Add Format Control Calculations/Validations" on page 57

# Add Fields

To add the required fields:

- 1. Click System Definition > Tables > cm3r.
- 2. Add the following required fields to the cm3r table. Do not change them.

	Туре			
Field	Service Manager	ServiceCenter		
qcintegration.type	Character	Text		
qcintegration.id	Number	Decimal		
qcintegration.project	Character	Text		

Note: The data type requirements for SM fields are described in "Matching Types" on page 31.

## Specify the External Access Definition on Service Manager

To specify the External Access Definition on Service Manager:

- Create a custom External Access Definition QCIntChangeService by clicking Tailoring > WSDL configuration on Service Manager 7.0x; or clicking Tailoring > Web Services > WSDL Configuration on Service Manager 7.1x or later with the following values:
  - Service Name: QCIntChangeService
  - Name: cm3r
  - Object Name: QCIntChange
  - Allowed Actions: save / Action Names: Update

object.name Change QCIntChange								
External Ad	External Access Definition							
Service Name	QCIntC	:han	igeService					
Name:	Name: cm3r			•	Obje	ct Name:	QCIntChange	
🔷 Allowed Actions 🔷			Expressions	4	Fields			
Allowed Actions			Action Nam	es		Action T	уре	
save		Update						

**Note:** The above values are required (Do NOT change them).

2. Enable required fields in the web service.

Field	Caption	Туре
header,number	ChangeNumber	StringType
qcintegration.id	QCEntityID	IntType
sysmodtime	Modified	DateTimeType

object.name						
Change	Change					
QCIntChange						
External Access Definition						
Service Name: QCIntChangeSe	rvice					
Name: cm3r	👻 Object Na	me: QCIntChange				
Allowed Actions	essions 🔷 Fields					
Field	Caption	Туре				
header,number	Change Number	StringType				
qcintegration.id	QCEntityID	IntType				
severity	Urgency	StringType				
sysmodtime	Modified	DateTimeType				
header,reason	Reason					
description.structure,desc	Description					
middle,logical.name	ConfigurationItem					
header,risk.assessment	RiskAssessment					
header,coordinator	Coordinator					
header,requested.by	RequestedBy					
header, priority.code	Priority					

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption and so on). The above values are required (Do NOT change them).

## Specify the External Access Definition on ServiceCenter

All fields of the ServiceCenter Change entity or Problem entity can be exposed in Web services by modifying their WSDL configuration. In ServiceCenter, you can modify the WSDL configuration by changing the Web Services API properties of the fields in table definition.

Note: Restart the ServiceCenter server whenever you make changes to a WSDL configuration.

To specify the External Access Definition on ServiceCenter:

 Click System Definition > Tables > cm3r > Fields and keys definitions for cm3r table and modify the settings of the following fields:

No.	Field	Include in API	Field name in API	Field data type in API
1	header,number	Y	ChangeNumber	StringType
2	qcintegration.id	Y	QCEntityID	IntType
3	sysmodtime	Y	Modified	DateTimeType

#### Web Services API properties

Field rendering in service oriented interface.

Include in API	
Field name in API:	QCEntityID
Field data type in API:	IntType

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on).

Click Toolkit > WSDL Configuration and search for the cm3r table. Update the External Access
Definition as follows based on the cm3r table.

No.	Field	Value
1	Service Name	QCIntChangeService
2	Object Name	QCIntChange
3	Allowed Actions	save
4	Action Names	Update

#### External Access Definition

Service Name:		QCIntCha	QCIntChangeService			
Name:		cm3r		🛃 🔍	Object Name:	QCIntChange
🔷 Allo	wed Actions	Expressions	🔷 Data Policy			
	Allowed Actio	ons		A	ction Names	
	save				pdate	
1 1	Save			9	paato	

## Create a Subform

To create a subform:

1. Create a global list by clicking **Tailoring > Tailoring Tools > Global Lists** on Service Manager; or clicking **Utilities > Tools > Global Lists** on ServiceCenter) with the following parameters:

No.	Parameter	Value	Remarks
1	List Name	SMQC Integration CM Project List	
2	Regen Every	1 00:00:00	
3	Build List on Startup?	Yes	Check box
4	List Variable	\$G.qcintegration.change.project	
5	User Defined List?	Yes	Check box
6	Value List	{"server1/domain1/project1", "server2/domain2/project2"}	Change to the values for your system.
			<b>Note:</b> No spaces between slashes.

Save this global list and click **Rebuild Global List** in the Options menu.

2. Click **Tailoring** > **Forms Designer** on Service Manager; or click **Toolkit** > **Forms Designer** on ServiceCenter to create the cm3r.qcint.subform subform with the following components:

Note: Click No when the system message "Do you want to use Form Wizard?" appears.

Component	Properties
Label	Caption: Forward to QC:

Component	Properties
Combo Box	Input: qcintegration.type
	Value List: 0;1;2
	<ul> <li>Display List:</li> <li>0 - Not Forward;1 - Forward as Requirement;2 - Forward as Defect</li> </ul>
	Select Only: Yes
	Read-Only Condition: [\$qcint.type.readonly]
Label	Caption: Def/Req ID:
Text	Input: qcintegration.id
	Read-Only: Yes
Label	Caption: Server/Domain/Project:
Combo Box	Input: qcintegration.project
	Value List: \$G.qcintegration.change.project
	Read-Only Condition: [\$qcint.project.readonly]
	Mandatory Condition: [qcintegration.type]>0

😽 Forms Designer: cm3r.qcint.subform 🗙									
Forward to QC:		*							
Def/Req ID:									
Server/Domain/Pro	ject:	~							
🔲 Properties 🗙	🔲 Properties 🗙								
Property	Value	Current Entries							
Input Read-Only Condition	qcintegration.type [\$qcint.readonly]	0 - Not Forward 1 - Forward as Requirement 2 - Forward as Defect							
Display List									

## Add the Subform to a Form

If the Service Manager Process Designer (PD) Content Pack 9.30.2 or 9.30.3 is not installed, follow the steps below to add the subform you created to a form:

**Note:** The following steps also apply to SM 9.4x Classic.

**Note:** In Service Manager 9.20 or later, a form may contain only collapsible sections (groups) instead of notebook tabs. If this is the case, replace the terms "notebook tab" and "tab" with "section" in the following steps.

- 1. Open the form of a phase of a category by using the Forms Designer (cm3r.rfc.build.g is used as an example).
- 2. Add a notebook tab with the QC Integration caption.
- 3. Add a subform to the new tab with format cm3r.qcint.subform, which is displayed in the following screenshot:

😽 Forms Designer:	cm 🗙							🔲 Properties 🛛	
			Reques	t For Chan	je	<b>_</b>		Si	ubformat
Fieldset117587883	35347							Property	Value
RFC No.				Planned 9	Start:			Name	subview1208421811523
Phase				Planned B	ind:			X	14
Status	_			Risk Asse	ssment			Y	2
				[				Width	90
Approval Status				Initial Imp	act Assessme	nt:		Height	16
Alert Stage:				Urgency:				Visible	$\checkmark$
Reason for Chang	e:			Priority:				Visible Condition	
	·			· ·				Format	cm3r.qcint.subform
				Folder:				Virtual Join	
	0.5	0. D	<u> </u>		A	A 155-1-		Display Blank	$\checkmark$
🔷 qc integration	🔶 General	Description	I ASSO	ciated CIs	🔶 CI Info	Affected		Display Using Table	
<b>P</b>							111	Input	
L									
1					-				
		-							
		-							

4. Save the changes.

**Note:** If the error message "Format 'cm3r.qcint.subform' not found (display, show.rio)" appears, log out and then log in again to enable the subform.

If the Service Manager Process Designer (PD) Content Pack 9.30.2 or 9.30.3 is installed, follow the steps below to add a subform:

**Note:** The following steps also apply to SM 9.4x Codeless.

- 1. Open the form of a phase of a category by using the Forms Designer (chm.normal.registration is used as an example).
- 2. Add a notebook tab with the QC Integration caption.
- 3. Add a subform to the new tab with format cm3r.qcint.subform, which is displayed in the following screenshot:

📓 To Do Queue: My To Do List 🛛 📓 Fo	rms Designer	🛃 Forms Designer: chm	normal.registration	n 🛛				Properties 🛛		
🖽 OK 🗯 Cancel							S. 🗸		Subformat	
III 🔁 🗂 🖿 🗆 🗮 🖿 👘	a 🔏 🗭 🛄 o		10 🗹 - 1	* 🗆 🖬 🖪 🚥	💿 🔜 🛲 🗮 I	) 🖇 🖪 🐻 🚺 🗖	l 🖧 🔝 🐂 🛛	Property	Value	
							*	Name	Value	
								X	16	
								Y	10	
								Width	64	
								Height	22	
								Visible	222 22	
♦ Workflow ♦ Affected Services ♦ A	ssociated CIs 🗇	Tasks 🗼 🔶 SLA 🍕	Attachments 4	OC Integration	Closure 🗼 OC I	ntegration »	ר ו	Visible Condition	•	
								Tab Stop	0	
								Format	cm3r.qcint.subform	
								Virtual Join		
								Display Blank	×	
								Display Using Table		
								Input		
							E			
							E			

3. Save the changes.

**Note:** If the error message "Format 'cm3r.qcint.subform' not found (display, show.rio)" appears, log out and then log in again to enable the subform.

## Add Format Control Calculations/Validations

**Note:** Steps in this topic are applicable if the Service Manager Process Designer (PD) Content Pack is not installed, or if you are working with SM 9.4x Classic. Otherwise, skip this topic and refer to "Add Rule Set Calculations/Validations" on page 59 for configurations in the PD environment and the SM 9.4x Codeless environment.

To add format control calculations and validations, follow the steps below:

 Open the format control record of the previous change form by clicking Tools > Format Control on ServiceCenter 6.2, or clicking Tailoring > Format Control on Service Manager (cm3r.rfc.build is used as an example in Service Manager 7.0x)

### 2. Click Calculations.

3. Add two rows with the following values.

display	initial	calculation
true	true	<pre>\$qcint.type.readonly=2;if (qcintegration.type in \$file~=0) then (\$qcint.type.readonly=1)</pre>
true	true	\$qcint.project.readonly=2;if (qcintegration.type in \$file~=0 and not null (qcintegration.project in \$file)) then (\$qcint.project.readonly=1)

**Note:** When you copy the calculations into the rows, make sure that each calculation is in one line; also note that there is a space between lines in the table above. For example, the calculation in the first row is: \$qcint.type.readonly=2;if (qcintegration.type in \$file~=0) then (\$qcint.type.readonly=1)

The Change calculations are shown in the following figure:

Form	is	Querie	es	Calc	lations JavaScript Validations Subroutines Addl Options	Privileges				
	Format Control Maintenance - Calculations									
Nam	ie:		cm3r.rf	c.build	View:	short				
add	update	delete	disp	initial	calculation					
true	true				risk.assessment in \$file=nullsub(risk.assessment in \$file, "1")					
true	true	true			misc3 in \$file=nullsub(misc3 in \$file, "no")					
		priorit			if (misc3 in \$file="yes") then (\$phasepntr=3;current.phase in \$file="RFC Testing")					
		true			billtype in \$file=nullsub(billtype in \$file, "dept")					
			true	true	<pre>\$qcint.readonly=2;if (qcintegration.type in \$file="1" or qcintegration.type in \$file="2") then (\$qcint.readonly=1)</pre>					

### 4. Click Validations.

5. Add a row with the following values.

No	Parameter	Value
1	Validation	not null(qcintegration.project in \$file)
2	Message	The Server/Domain/Project is required.

No	Parameter	Value
3	Add	qcintegration.type in \$file~=0
4	Update	qcintegration.type in \$file~=0
5	Set Focus to	qcintegration.project

The Change validation values are shown in the following figure:

Validations			
Validation	not null(qcintegration.project in \$file)	Delete	
Message	The Server/Domain/Project is required.	Display	
Comments		Initial	
Add	qcintegration.type in \$file~=0	Set Focus to	qcintegration.project
Update	qcintegration.type in \$file~=0	Message ID	

6. Save the changes.

## Add Rule Set Calculations/Validations

**Note:** Steps in this topic are applicable for Service Manager 9.3x with Process Designer (PD) Content Pack 9.30.2 or 9.30.3 and Service Manager 9.4x Codeless. Otherwise, refer to "Add Format Control Calculations/Validations" for configurations in SM 9.3x non-PD environment and SM 9.4x Classic.

To add rule set calculations and validations, follow the steps below:

- 1. "Copy an Existing Workflow" below
- 2. "Associate an Existing Change Category with the New Workflow" on the next page
- 3. "Create New Rule Set for Initialization and Validation" on page 61
- 4. "Associate the New Workflow with the New Rule Set" on page 64

### Copy an Existing Workflow

You can use copies of the existing workflows in another business process, or make changes to the HP proprietary workflow copies.

To copy an existing workflow, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Copy Existing Workflow**.
- 2. On the Clone a Workflow page, select the workflow you want to copy. For example, Normal.
- 3. Type SMQCIntChM in the New workflow name field.

To Do Q	ueue: My To Do List	Wizard: Clone a Workflow 🗵	
Clo	one a Workflov	/	
		Please specify the new workflow name, as well as the prefix for new rule sets if they are to be copied as w	ell.
		New workflow name: * SMQCIntChM	
		Copy rule sets?	

- 4. Select the Copy rule sets check box if you want to copy rule sets, and then type a rule set prefix.
- 5. Click **OK**.

The newly copied workflow appears in the list on the Clone a Workflow page.

### Associate an Existing Change Category with the New Workflow

You can update existing change categories, subcategories and areas and associate the updated categories with the new workflow so that they can be used in another business process. For more information, refer to *HP Service Manager – Process Designer Content Pack Administrator's Guide*.

To associate an existing change category with the new workflow, follow the steps below:

- 1. From the System Navigator, click **Change Management > Configuration > Change Categories**.
- 2. Click Search.
- 3. Select the change category for which you want to add a workflow. For example, Normal Change.
- 4. In the Change Category Definition page, remove the currently assigned workflow from the Workflow field.
- 5. Type SMQCIntChM in the workflow field.

6. Click **Save** to associate the change category with the workflow.

Change Category Definition	
Name * Normal Change	
Description Normal Change	
Availability ≠ true	
Workflow * SMOCINICIAM	
Default Template	
Assign number before commit?	
Workflow SubCategories	4
Registration and Categorization 💊 Valdation 🛶 Risk and Impact Analysis 🛶 TCAB Approval 🛧 Build and Test 🛶 DCAB Approval 💠 DCAB Approval	Backout
Abandoned CMDB Update Post imp	elementation Review
	+
	Closure

### Create New Rule Set for Initialization and Validation

To create a new rule set for initialization, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Rule Sets**.
- 2. Type the values as follow:

Field	Value
ID	chm.alm.int.init
Available as action	
Name	Initialize for ALM integration in the Change Record
Table name	cm3r
HP Proprietary	

- 3. Click New and Save.
- 4. Click Add Rule.
- 5. In the Select Rule Type page, click Run JavaScript.

Installation and Administration Guide Chapter 6: SM Change -> QC/ALM Defect

6. In the Run JavaScript page, type the values as follow:

Field	Value		
Rule Description	Run Javascript for initializing Integration type and project in the Change Record		
Statement	<pre>vars['\$qcint.type.readonly'] = 2; vars['\$qcint.project.readonly'] = 2; var _null=system.functionsnull; var file = vars.\$L_file; if(file["qcintegration.type"] !=0 &amp;&amp; !_null(file["qcintegration.type"])) { vars['\$qcint.type.readonly'] = 1 } if(file["qcintegration.type"] !=0 &amp;&amp; !_null(file["qcintegration.project"])) { vars['\$qcint.project.readonly'] = 1 }</pre>		

### 7. Click **OK**.

8. Click **Save** and **Exit**.

To create a new rule set for validation, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Rule Sets**.
- 2. Type the values as follow:

Field	Value
ID	chm.alm.int.validation
Available as action	
Name	Validation for ALM integration in the Change Record

Field	Value
Table name	cm3r
HP Proprietary	

- 3. Click New and Save.
- 4. Click Add Rule.
- 5. In the Select Rule Type page, click **Set Mandatory Fields**.
- 6. Click Edit. The Condition Editor opens.
- 7. Add an expression as illustrated in the following screenshot:

CurrentRecord	=	Value
Qcintegration Type		0
	Apply	

- 8. Click Apply.
- 9. Add condition, Select the AND operator, and then add another expression as illustrated in the following screenshot:

CurrentRecord.Qcintegration Type != 0 🥒
AND
CurrentRecord != Blank/NULL Qcintegration Type Apply
Add Condition -

### 10. Click Apply.

Condition Editor		
1		
	CurrentRecord.Qcintegration Type != 0 🧷	
	AND	
	CurrentRecord.Qcintegration Type != NULL 🧷	
	Add Condition -	
Clear	****	Ok Cancel

- 11. Click **OK** to exit the Condition Editor.
- 12. Click **OK** to exit the Set Mandatory Fields page.
- 13. Click Save and Exit.

### Associate the New Workflow with the New Rule Set

To associate the new workflow with the new rule set, follow the steps below:

- 1. From the System Navigator, click **Change Management > Configuration > Change Workflows**.
- 2. Select SMQCIntChM in the workflows list.
- 3. Select the first phase in the workflow graph.
- 4. Click **Rule Sets** tab > **Initialization** tab.
- 5. Click **Add** and select the chm.alm.int.init rule set you just created.

MP Service Manager			User: falcon	Logo
	To Do Queue: My To Do List Workflows Workflow: SMQC	IntChM ®		
🛛 🗟 😒 🔍	💾 Save   🍳 Zoom in 🤤 Zoom out   🗖 Add phase 🛞 Delete	Workflow properties	🗏 🗖 I	
Request Management Service Catalog Service Desk Service Desk Service Level Management System Administration Taloring > Audt > Differential Upgrade > Differential Upgrade > Differential Upgrade > Detrent Services > Kenvices > Kenvices	Abandoned Add Rule Sets	Impadi Analysis TCAB Approval	CCAB Approval CAB Approval CMDB Update CMDB Update Course	
Process Designer     Configuration     Copy Existing Workflow     Export Workflow     Export Workflow     Exub Sets     Workflows     SoL Utitles     Tailoring Tools     Web Services     Codes     Database Manager     Database Manager     Database Manager     Sature	CoseChain CoseC	ate Apply Template all Chm 00 Initial dation chm 00 Validation init Initiatize for ALUI Integrati validation Validation Validation for ALUI Integrat validation Chm Apply Change Mode losetime Chm Cleae Related Chm Cleae Related	ation in the Change Record	

- 6. Click **OK**.
- 7. Click Rule Sets tab > On display tab.
- 8. Repeat step 5 and 6.
- 9. Click **Rule Sets** tab > **On update** tab.
- 10. Click **Add** and select the chm.alm.int.validation rule set you just created.

HP Service Manager					User. falcon
		To Do Queue: My To Do List Workflows	Workflow: SMQCIntChM		
🔁 i 🛃 🥵	**	💾 Save   🔍 Zoom in 🔍 Zoom out   🔲 Add	phase 💮 Delete   🚰 Workflow properties		8 🖬 (
Request Management	^				
Service Catalog					
Service Desk					
Service Level Management				<del>*</del>	
System Administration		Registration and Categorization Validation	n 💛 Risk and Impact Analysis 💛 TCAB Approval 🕁	Build and Test	Backout
Tailoring		dd			,
> Audit					
Differential Upgrade				CMDB Upd	ate Post Implementation Review
Document Engine		Abandoned		СМОВ Ора	ate Post Implementation Review
Event Services			Add Rule Sets - On enter		
Knowledge Engineering			E Id	Name	
Notifications			CloseChange	Close Change	Closure
Process Designer		1	apply.template	Apply Template	
Configuration		•	chm.00.initial	Chm DO Initial	
Copy Existing Workflow Export Workflow		Phase - Registration and Categorization	Chm.00valdation	chm.00Validation	
Rule Sets		Details Forms Rule Sets Actions	Chm.alm.int.int	Initialize for ALM integration in the Change Record	-
Workflows	=				
SQL Utilities		On enter On exit Initialization On	Chm.alm.int.validation	Validation for ALM integration in the Change Record	•
Tailoring Tools		🖨 Add 💮 Delete 😚 View 🏠 Up 🕹 Dow	chm.apply.change.model	Chm Apply Change Model	
Web Services		Rule Sets	chm.clean.closetime	Chm Clean Closetime	
Codes		Rule Sets	chm.close.related	Chm Close Related	
Database Dictionary			chm.closure.wizard	Chm Closure Wizard	-
Database Manager					
Data Policy				OK Cancel	
Format Control			L		

- 11. Click **OK**.
- 12. Click **Rule Sets** tab > **On enter** tab.
- 13. Repeat step 10 and 11.

14. Click Save.

# Customizing the QC/ALM Defects Module

The steps for customizing the Defects module vary with different QC versions.

- "On QC 10 or Earlier" below
- "On ALM 11" on page 72

# On QC 10 or Earlier

To customize the Defects module on Quality Center 10 or earlier, perform the following tasks:

- 1. "Add Fields" below
- 2. "Add Tabs" on the next page
- 3. "Add Fields to Tabs" on page 69
- 4. "Verify" on page 70

### Add Fields

To add the required fields for Defect module customization:

- 1. Log on to QC as a project administrator.
- 2. Click Tools / Customize. The "QC Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (*XX* and *XY* are sequential numbers auto-generated by QC).

Field Name	Field Label	Field Type
BG_USER_XX	Change ID	String
BG_USER_ <i>XY</i>	Created from	String

The following figure shows an example project entity.

P	roject Entities		
Pro	ject Entities	- Field Settings	
	∰ Cycle ∰ Defect	Field Name:	BG_USER_02
<u>y Management</u>	⊕ <u>)</u> System Fields ⊟ <u>)</u> User Fields	Field Label:	Change ID
		Field Type:	String
	🏭 Release 🏭 Release Folder 🊟 Requirement	Field Length:	40
		🗌 History	Required
		☐ History ☐ Masked	Required     Searchable

**Note:** The data type requirements for QC fields are described in "Matching Types" on page 31.

### Add Tabs

To add tabs to the Defect form and show fields on these tabs:

1. In "QC - Project Customization", click **Workflow** > **Script Editor**.

#### 2. Select Defects module script.

🖲 📓 Manual Runner script

E S Defects module script

Quality Center - Project Customization

User Properties Project Users	Workflow				
Groups Module Access Project Entities Requirement Types Risk-Based Quality Man Project Lists Automal	icript Generator - Add Defect Field Customization inables you to customize the fields displayed for each user group in the Add Defects dialog box ou can also specify field order and whether a field is required. <u>cript Generator - Defect Details Field Customization</u> nables you to customize the fields displayed for each user group in the Defect Details dialog ox. You can also specify field order and whether a field is required.				
Alert Rules Workflow	<u>Script Editor</u> Enables you to write VBScript code for all Quality Center modules. You can also use the Script Editor to modify the scripts generated by the above tools.				
Script Editor Script Editor Toolbar Bu	ton Editor GetNewBugPageName				
Workflow Scripts     Ormmon script     Sequirements m     Energy Test Plan modul     Energy Test Lab modul	e script				

Add the following code to the GetNewBugPageName event procedure (which is triggered before QC opens the Add Defect dialog box).

Bug\_CanDelete Bug\_AfterPost SetFieldApp

```
select case PageNum
case "2"
GetNewBugPageName = "SM Integration (New)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For a new bug, the tab name is SM Integration (New).

4. Add the following code to the **GetDetailsPagename** event procedure (which is triggered before QC displays the Defect Details dialog box).

```
select case PageNum
case "2"
GetDetailsPageName = "SM Integration (Details)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For an existing defect, the tab name is SM Integration (Details).

### Add Fields to Tabs

To add fields to tabs:

- 1. In "QC Project Customization", click **Workflow** > **Script Editor**.
- 2. Select **Defects module script**.

User Properties Project Users	Workf	w				
Groups         Script G           Module Access         Enables           Project Entities         You car           Requirement Types         Script G           Risk-Based Quality Man         Enables           Project Lists         Enables		enerator - Add Defect Field Customization : you to customize the fields displayed for each user group in the Add Defects dialog bo also specify field order and whether a field is required. enerator - Defect Details Field Customization : you to customize the fields displayed for each user group in the Defect Details dialog can also specify field order and whether a field is required.				
Alert Rules Workflow	Script Ed	litor you to write VBScript code for all Quality Center modules.				
Script Editor	You can	also use the Script Editor to modify the scripts generated by the above tools.				
Script Editor 1	You can Youbar Button Editor	also use the Script Editor to modify the scripts generated by the above tools.  GetNewBugPageName GetDetailsPageName Bug_New				
Script Editor 1	You can foolbar Button Editor r Scripts ton script	also use the Script Editor to modify the scripts generated by the above tools.   GetNewBugPageName  GetDetailsPageName  Bug_New Bug_MoveTo				
Script Editor 1	You can Youbar Button Editor	also use the Script Editor to modify the scripts generated by the above tools.				
Script Editor 1	You can Youbar Button Editor Scripts ion script rements module script Plan module script Lab module script	also use the Script Editor to modify the scripts generated by the above tools.   GetNewBugPageName  GetDetailsPageName  Bug_New Bug_New Bug_FieldCanChange				
Script Editor 1 	You can Youbar Button Editor Scripts non script rements module script Plan module script	also use the Script Editor to modify the scripts generated by the above tools.				

- 3. If **WizardFieldCust\_Details** and **WizardFieldCust\_Add** are not found in the list, do the following to generate these two methods.
  - a. Script Generator Add Defect Field Customization
  - b. Script Generator Defect Details Field Customization



4. Add the following code to the **WizardFieldCust\_Details** event procedure.

SetFieldApp "BG\_USER\_XX", True, False, 1, 0

SetFieldApp "BG\_USER\_XY", True, False, 1, 1

The parameter values are:

- Field name (BG\_USER\_XX, where XX consists of two digits)
- Visible (True)
- Required (False)
- Page number (start from Ø)
- View order (start from Ø)
- 5. Add the following code to the WizardFieldCust\_Add event procedure.

SetFieldApp "BG\_USER\_XX", True, False, 1, 0

```
SetFieldApp "BG_USER_XY", True, False, 1, 1
```

 Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

Bug\_Fields.Field("BG\_USER\_XX").IsReadOnly=True

Bug\_Fields.Field("BG\_USER\_XY").IsReadOnly=True

7. Save your changes.

### Verify

To verify whether the Defects module on Quality Center 10 or earlier is customized successfully:

New Def	fect	
🗙 Clear	Attach: 🥒 🥜 📸 🚹 🚔 🔊 🕶 💱 💷 🖤	0
* Summary:		
Details	SM Integration (New)	
	Change ID: Created from:	
	Submit Close	

1. Create a new defect. The dialog box has a new tab titled "SM Integration (New)" with two fields.

2. Open an existing defect. The second tab is titled "SM Integration (Details)", and both the "Change ID" field and the "Created from" field are read-only.

📴 Defect Deta	ils									_ 0	X
	►I   🕵	► !	<u>=</u> ≋ ▼	1							0
Defect:	13	a new de	efect								
	Details	SM Inte	gration (	Details)							
Q Details		Ch	ange ID:	C18			C	Created from	Created f	rom SM/SC	
17											
Attachments											
a											
Linked Entities											
<b>\$</b>											
History											
Execution Report					ок	Cancel					

# On ALM 11

On ALM 11, you only need to add new fields directly to the Details tab of the Defect form.

To customize the ALM Defects module, perform the following tasks:

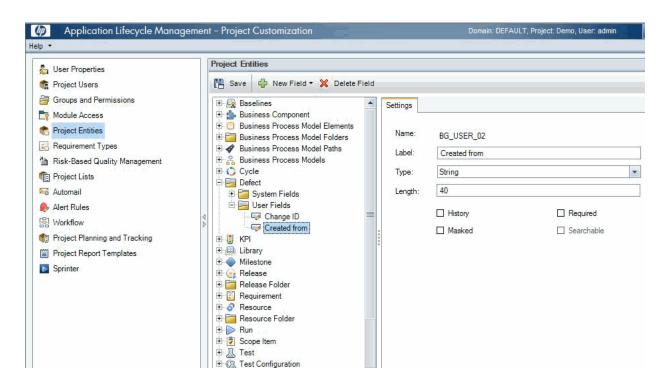
- 1. "Add fields" below
- 2. "Verify" on the next page

### Add fields

To add the required fields for Defect module customization:

- 1. Log on to ALM as a project administrator.
- Click Tools / Customize. The "Application Lifecycle Management Project Customization" module opens.
- 3. Add the following fields for the defect entity in project entities (*XX* and *XY* are sequential numbers auto-generated by ALM).

Field Name	Field Label	Field Type
BG_USER_XX	Change ID	String
BG_USER_ <i>XY</i>	Created from	String



- 4. Click Workflow > Script Editor.
- 5. Select Defects module script.
- Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

Bug\_Fields.Field("BG\_USER\_XX").IsReadOnly=True

Bug\_Fields.Field("BG\_USER\_XY").IsReadOnly=True

7. Save your changes.

#### Verify

To verify the Defects module on ALM 11 is customized successfully:

New Defect		
🗙 💩 - 🍫 🛃 📇		-
* Summary:		
Details Attachments		
	Submit Close Help	

1. Create a new defect. Both the "Change ID" field and the "Created from" field are read-only.

2. Open an existing defect. Both the "Change ID" field and the "Created from" field are always readonly.

🖳 🗖 🔀							
Defect Details   Image: Subscript of the state of the	Details Reproducible: Severity: City City City City City City City City	Medium	Assigned To: Status: Target Cycle: Change ID: Problem ID: ments:	New           C10003	dd Comment		
OK Cancel Help							

## Configuring Links in QC/ALM Synchronizer

To configure and test a link in the QC/ALM synchronizer, perform the following tasks:

- 1. "Specify Endpoints / Type of Link" below
- 2. "Define Field Mappings" on the next page
- 3. "Define Events" on page 80
- 4. "Test the Link" on page 80

## Specify Endpoints / Type of Link

**Caution:** If you are using ALM Synchronizer 1.4 with ALM 11.0, install the HP Quality Center Connectivity Add-in on the ALM server before you proceed; otherwise ALM Synchronizer will not be able to connect to ALM. To install this add-in, click the **Add-Ins Page** link on your ALM options window: http:// <ALM Platform server name> <: port number >/ qcbin.

Specify the connection properties as described in "Create a Link" with the following settings specific for this type of link:

- 1. Step 1: "Endpoint 2 type" = **SM ChangeManagement**.
- 2. Step 2: "Service URL" = http://service\_manager\_host>:<port>/sc62server/PWS/QCIntChangeService.wsdl
- 3. Step 3: "Select entity types" = Change as Defect.

## Define Field Mappings

Basic field mappings are summarized below:

QC	Direction	SM	Constant value	Remarks
Change ID	<-	ChangeNumber		
Defect ID	->	QCEntityID		Synchronize back on create: Yes
Created from			Created from SM/SC	

Example field mappings are shown in the following screenshot:

	Mapped Fields					
	Туре	QC Field	Direction	SM ChangeManagement Field		
	<u> </u>	Severity	<>	Urgency		
		Change ID	<	ChangeNumber		
I		Defect ID	>	QCEntityID		
	<b>D</b>	Summary	<>	Description		
	</th <th>Created from</th> <th>&lt;</th> <th>Value: Created from SM/SC</th>	Created from	<	Value: Created from SM/SC		

### QC/ALM Field <-> SM Field

The following table summarizes the field mappings between QC/ALM and SM. The first two rows are required mappings.

SINCI	Sm Change - QC/ALM Delect mappings								
QC Len	QC DB Name	QC Type	QC/ QCS Label	QC/ QCS Type	Dir	QCS SM Type	QCS Name/ SM WSDL Caption	SM WSDL Type	SM DB Name/ SM WSDL field
40	BG_USER _02 <sup>2</sup>	String	Change ID	String	<-	String	ChangeNumber	StringType	header,number
10	BG_BUG _ID	Number	Defect ID <sup>3</sup>	Number	->	Number	QCEntityID <sup>4</sup>	IntType	qcintegration.id
255	BG_ SUMMARY	String	Summary	String	<->	String	Description	StringType	description.struc description
70	BG_ SEVERITY	Lookup List	Severity	Single value list	<->	Single value list	Urgency	Character	severity

#### SM Change -> QC/ALM Defect Mappings

If you specify a value mapping, for example, Severity <-> Urgency, you can specify as follows:

QC De	fect Schem	a	SM ChangeManagement			
Name		ype	Name Type			
Pu Se	wanity S	ingle value list	Urgency Single valu	ie list		
lappe	d Fields			Mapping properties	Value ma	pping Field properties
Туре	QC Field	Direction	SM ChangeManagement	Endpoint 1 value	Direction	Endpoint 2 value
Type	dic rielu	Discussi	Field	5-Urgent	<>	1 - Critical
C	Change ID	<	ChangeNumber	4-Very High	<>	2 · High
6	Defect ID	>	QCEntityID			-
	•	1.		3-High	<>	3 - Average
6	Summary	<>	Description	2-Medium	<>	4 - Low
6	Severity	<>	Urgency	1		

The following tables summarize the mappings you just created between the single value lists (which have their own directions).

SM Change	-> QC	<b>Defect Lis</b>	t Value	Mappings
-----------	-------	-------------------	---------	----------

QC Len	QC DB Name	QC Type/ Lookup list Values	QC/QCS Label	QC/QCS Type	Dir	QCS SM Type	QCS Name/ SM WSDL Caption	SM WSDL Type	SM DB Name/ SM WSDL Field	SM DB Type	SM Len
70	BG_ SEVERITY	Lookup List	Severity	Single value list	<->	Single value list	Urgency	String Type	severity	Char (SM7) or Text (SC6)	40

Detailed value mappings between the QC/QCS Severity field and the QCS/SM WSDL Urgency field are listed in the table below:

QC Value (from Lookup List) <sup>1</sup>	Value Map Dir	SM Field Value (from SM Adapter Config File for Change Management) <sup>2</sup>
5-Urgent	<->	1-Critical
4-Very High	<->	2-High

QC Value (from Lookup List) <sup>1</sup>	Value Map Dir	SM Field Value (from SM Adapter Config File for Change Management) <sup>2</sup>
3-High	<->	3-Average
2-Medium	<->	4-Low

1

#### Lookup list is created in QC.

Field Settings		
Field Name:	BG_SEVERITY	Project Lists
Field Label:	Severity	Lists: Severity 💌
Field Type:	Lookup List 🔹	List Items
Lookup List		1-Low
		📄 2-Medium
Severity 💌	New List Goto List	📄 3-High
		📄 📄 4-Very High
🔽 Verify Value		📄 5-Urgent

<sup>2</sup>The XML file is in <*QCS\_Install\_Dir*>\adapters\dat\SM ChangeManagement\configuration\_file\_default.xml (see "SM Change Management Example" on page 21).

## **Define Events**

The following table lists the event settings for the two endpoints.

Operation	QC Action (Event)	SM Action (Event)
Creation	Do nothing.	Create a corresponding record in the other endpoint.
Update	Update its corresponding record in the other endpoint.	Update its corresponding record in the other endpoint.
Deletion	Do nothing.	Do nothing.

#### The following screenshot shows the settings:

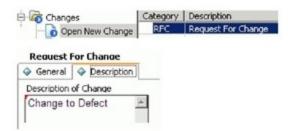
C Links → ⊅ change_to_defect_1	
General Connectivity Scheduling Filters Events Field Mapping Adv	vanced
QC	SM ChangeManagement
Creation	Creation
When a record is created in this endpoint	When a record is created in this endpoint
C Create a corresponding record in the other endpoint	Create a corresponding record in the other endpoint
C Do nothing	Do nothing
Update	Update
When a record is updated in this endpoint	When a record is updated in this endpoint
Update its corresponding record in the other endpoint	C Update its corresponding record in the other endpoint
Do nothing	Do nothing
Deletion (Full Synchronization Only)	Deletion (Full Synchronization Only)
When a record is deleted from this endpoint	When a record is deleted from this endpoint
© Do nothing	© Do nothing
© Delete its corresponding record in the other endpoint	© Delete its corresponding record in the other endpoint
© Recreate based on its corresponding record in the other endpoint	© Recreate based on its corresponding record in the other endpoint

## Test the Link

To test the link:

**Note:** The following sample steps are for your reference only. The exact steps required on your system may differ significantly. The phase in which the QC Integration tab appears may be different on your system.

- 1. Save the configuration (an integrity check is automatically run).
- 2. Click Enable Link.
- 3. Create a Service Manager Change (the category of the Change depends on each Service Manager customization; RFC is used in ServiceCenter 6.2/Service Manager 7.0x as an example).



4. Change the phase to **Building**. The QC Integration tab appears.

Phase Name	Description
Assessment	Assessment
Building	RFC Building
<b>RFC Implementation</b>	Place Changes Into Service
RFC Testing	RFC Testing

5. Select a value in the Server/Domain/Project field and select **Forward as Defect** in the Forward to QC field.

Backout Method	History	Approvals	Attachments	Related Records	Workflow	QC Integration	27
	Forward to	QC:		2 - Forward as Defe	t		
	Def/Req ID	:					

#### 6. Synchronize.

🛞 Cancel Current Task 📄 View Report 🔣 Refresh Progress 🔽 Auto Refresh
Running: Connecting to endpoint 1 Running: Connecting to endpoint 2 Running: Querying non filtered set Running: Handling endpoint 1 - Processing entity #1 of #1 in the Create list, (Total: passed = 0, failed = 0) Passed: Disconnecting Completed : Passed

7. View the Defect in QC.

📴 Defect Deta	ils 📃 🖂 🕹
Defect:	9 Change to defect
	Details SM Integration (Details)
Details	Change ID: C19 Created from: Created from SM/SC
- Oy	
Attachments	
â	
Linked Entities	
<b>\$</b>	
History	
Execution Report	OK Cancel

**Note:** In ALM 11, the Change ID and Created from fields reside on the Details tab of the Defect form.

# Chapter 7: SM Change -> QC/ALM Requirement

This chapter describes how to synchronize SM Changes with QC/ALM Requirements ("Change -> Requirement").

This chapter includes:

- "Customizing Service Manager/ServiceCenter for Change Management" below
- "Customizing the QC/ALM Requirements Module" below
- "Configuring Links in QC/ALM Synchronizer" on page 90

# Customizing Service Manager/ServiceCenter for Change Management

For details, see the "Customizing Service Manager/ServiceCenter for Change Management" section in "SM Change -> QC/ALM Defect".

## Customizing the QC/ALM Requirements Module

The steps for customizing the Requirements module vary with different Quality Center/ALM versions.

- "On QC 10 or Earlier" below
- "On ALM 11" on page 88

## On QC 10 or Earlier

To customize the Requirements module on Quality Center 10 or earlier, perform the following tasks:

- 1. "Add Fields" on the next page
- 2. "Add Tabs" on page 85

- 3. "Add Fields to Tabs" on page 87
- 4. "Create the SM Incoming Changes Folder" on page 89

### Add Fields

To add required fields for requirement customization, follow these steps.

- 1. Log on to QC as a project administrator.
- 2. Click **Tools / Customize**. The "QC Project Customization" module displays.
- 3. Add the following fields for the requirement entity in project entities (*XX* and *XY* are sequential numbers auto-generated by QC).

Field Name	Field Label	Field Type
RQ_USER_ <i>XX</i>	Change ID	String
RQ_USER_ <i>XY</i>	Created from	String

**Note:** The data type requirements for QC fields are described in Matching Types.

This is shown in the following figure.

Project Entities			Project Entities
Project Entities	Field Settings     Field Name:     Field Label:     Field Type:     Field Length:     History     Masked	RQ_USER_01 Change ID in SM String 40 Searchable	Project Entities

 In Requirement Types add fields "Change ID"/"Created from" to the Business type requirement. Business type is the default requirement type for incoming requirements (other types can be used).

User Properties Project Users Requirement Types	
Groups Module Access Types-Properties-	
Project Entities  Requirement Types  Assigned Icon:  Replace	
Risk-Based Quality Mana Functional Test Coverage: None None	•
Automail         Group         Risk-Based Quality         Perform Assessment           Alert Rules         Undefined         Undefined         Undefined	•
Workflow User Fields	
Not in Type     In Type       Old Type (obsolete)     >       Name     Rec       Change ID in SM     X	uired

### Add Tabs

To add tabs to the Requirement form and display the fields on these tabs, click **Workflow** > **Script Editor**. Add the following code to the requirement module.

**Note:** For a new Requirement, the tab label is "SM Integration (New)". For an existing Requirement, the tab label is "SM Integration (Details)". The parameter 2 specifies tab 2 (the second tab). If N tabs exist, then the number of a new tab should be N+1. This function is called when an existing requirement is shown in the dialog.

```
Sub SetReqField( FieldName, Vis, Req, PNo, VOrder )
With Req_Fields(FieldName)
.IsVisible = Vis
.IsRequired = Req
.PageNo = PNo
.ViewOrder = VOrder
End With
End Sub
Function GetNewReqPageName(PageName,PageNum)
On Error Resume Next
select case PageNum
case "2"
```

Installation and Administration Guide Chapter 7: SM Change -> QC/ALM Requirement

```
GetNewReqPageName = "SM Integration (New)"
```

end select

On Error GoTo 0

End Function

Function GetReqDetailsPageName(PageName,PageNum)

On Error Resume Next

select case PageNum

case "2"

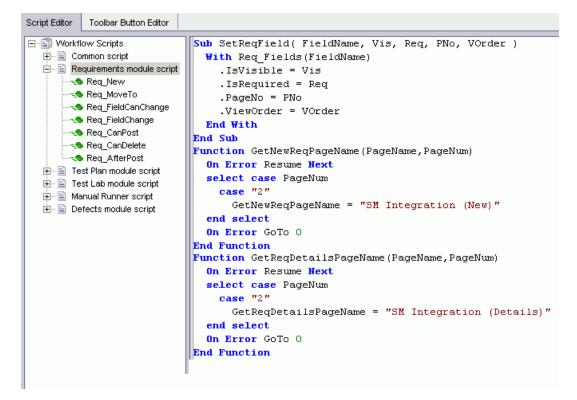
```
GetReqDetailsPageName = "SM Integration (Details)"
```

end select

On Error GoTo 0

End Function

The resulting script is shown in the following screenshot:

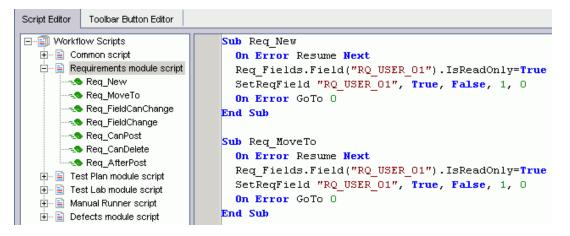


### Add Fields to Tabs

To set the fields as read-only and place the fields on the tabs, in the Script Editor for the Requirements module script, add the following code to **Req\_New** and **Req\_Moveto** (**Req\_New** is called when a new Requirement is created; **Req\_Moveto** is called when an existing Requirement is opened).

```
Req_Fields.Field("RQ_USER_XX").IsReadOnly=True
Req_Fields.Field("RQ_USER_XY").IsReadOnly=True
SetReqField "RQ_USER_XX", True, False, 1, 0
SetReqField "RQ_USER_XY", True, False, 1, 1
```

The resulting script is shown in the following screenshot:



### Create the SM Incoming Changes Folder

To create the folder for Requirements originated from SM changes:

- 1. From the menu, select **Requirements / New Folder**.
- 2. Set the folder name to SM Incoming Changes.

Requirements Edit View Favorites A	Analysis	
📖 🕼 × 💁 🏹 • 🔟 🛤 🖉 🔍	R   E -   Q;	
🕕 🕃 🗜 🛌 Name	<ul> <li>Direct Cover Status</li> </ul>	Req ID
E C Requirements	Direct Cover Status	Req ID 0

## On ALM 11

In ALM 11, you only need to add new fields directly to the Details tab of the Requirement form.

To customize the ALM Requirements module, perform the following tasks:

- 1. "Add Fields" below
- 2. "Create the SM Incoming Changes Folder" on the next page

### Add Fields

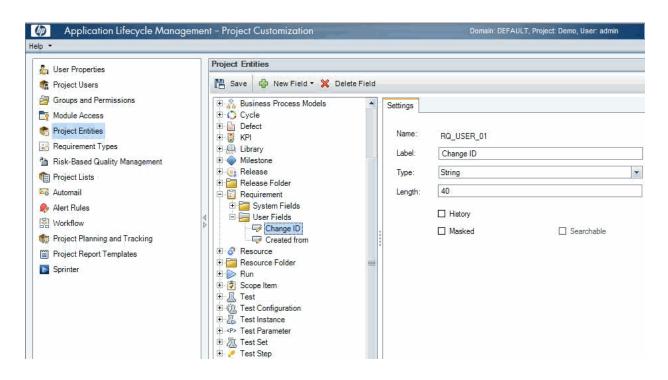
To add required fields for requirement customization, follow these steps.

- 1. Log on to ALM as a project administrator.
- Click Tools / Customize. The "Application Lifecycle Management Project Customization" module opens.
- 3. Add the following fields for the requirement entity in project entities (*XX* and *XY* are sequential numbers auto-generated by ALM).

Field Name	Field Label	Field Type
RQ_USER_ <i>XX</i>	Change ID	String
RQ_USER_ <i>XY</i>	Created from	String

Note: The data type requirements for QC fields are described in "Matching Types" on page 31.

This is shown in the following figure.



- In Requirement Types add fields "Change ID"/"Created from" to the Business type requirement. Business type is the default requirement type for incoming requirements (other types can be used).
- 5. Click Workflow > Script Editor.
- 6. Select Requirements module script.
- 7. Add the following code to Req\_New and Req\_Moveto (Req\_New is called when a new Requirement is created; Req\_Moveto is called when an existing Requirement is opened.)

Req\_Fields.Field("RQ\_USER\_XX").IsReadOnly=True

Req\_Fields.Field("RQ\_USER\_XY").IsReadOnly=True

8. Save your changes.

### Create the SM Incoming Changes Folder

To create the folder for Requirements originated from SM changes:

- 1. From the menu, select Requirements / New Folder.
- 2. Set the folder name to SM Incoming Changes.

# Configuring Links in QC/ALM Synchronizer

To configure and test a link in QC/ALM synchronizer, perform the following tasks:

- "Specify Endpoints / Type of Link" below
- "Define Field Mappings" on page 92
- "Define Events" on page 94
- "Test the Link" on page 94

## Specify Endpoints / Type of Link

Specify the connection properties as described in "Create a Link" with the following settings specific for this type of link:

- 1. Step 1: "Endpoint 2 type" = SM ChangeManagement.
- 2. Step 2: "Service URL" =
  http://service\_manager\_host>:<port>/sc62server/PWS/QCIntChangeService.wsdl
- 3. Step 3: "Select entity types" = Change as Requirement.

4. Specify the incoming requirement folder as shown in the following screenshot:

General	Connectivity	Scheduling	Filters	Sut	otype Mapping	Advanced
	<u> </u>	QCIntUser				
	arameter omain		Valu DEF4	-		
Pr	oject		Demo	D		
Se	erverURL		http:/	//loca	alhost:8080/qct	pin
					Check Conne	ectivity
	Use alternate equirements\		:d_incon	ning_	requirement_fol	der>

Requirements will be created in the specified folder in QC.

🕕 🖁 🚦 🛌 Name		
🖃 🚞 Requirements		
🖂 😑 SM Incoming Changes		
🔳 💼 change	to problem	

5. In the "Sub types mapping" tab, specify the type of requirements created from changes. General Connectivity Scheduling Filters Sub Types Mapping

Endpoint 1 Available Sub Ty	pes -	Mapped Sub Types			
Functional Testing Undefined	Add Mapping >> Available Sub Types	Business - S/I Change			
		Req ID: 6 * Name: change to problem	* Requirement Type:	Business	ž
				1 Group	
			ć	Undefined	-

## Define Field Mappings

Basic field mappings are summarized below:

QC	Direction	SM	Constant value	Remarks
Change ID	<-	ChangeNumber		
Req ID	->	QCEntityID		Synchronize back on create: Yes
Created from			Created from SM/SC	

Example field mappings are shown in the following screenshot:

Mapped Fields						
Type QC Field		Direction	SM ChangeManagement Field			
- E	Name	<>	Description			
Ð	Change ID	<	ChangeNumber			
- E	Reg ID	>	QCEntityID			
$\langle \rangle$	Created from	<	Value: Created from SM/SC			

### QC/ALM Field <-> SM Field

The following table summarizes the field mappings between QC/ALM and SM. The first two rows are required mappings.

QC Len	QC DB Name	QC Type	QC/QCS Label	QC/ QCS Type	Dir	QC S SM Type	QCS Name/ SM WSDL Caption	SM WSDL Type	SM DB Name	SM7 DB type	SC6 DB type	SM Len
40	RQ_USER_01	String	Change ID	String	<-	String	ChangeNumber	StringType	header,number	Char	Text	100
10	RQ_REQ_ID	Number	ReqID	Number	->	Number	QCEntityID	IntType	qcintegration.id	Num	Decimal	xx
255	RQ_REQ_ COMMENT	Memo	Description	String	<->	String	Description		description.structure, description	Char	Text	xx

SM Change -> QC/ALM Requirement Mappings

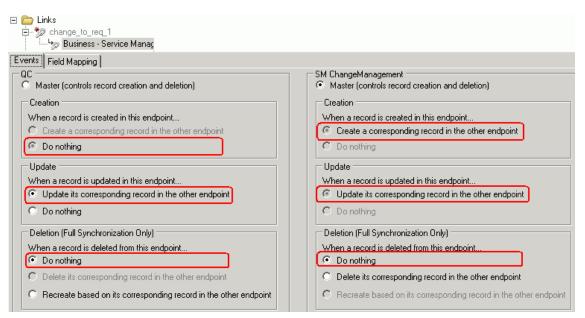
#### **Note:** Check for QCEntityID mapping property **Synchronize back on create**.

## **Define Events**

The following table lists the event settings for the two endpoints.

Events Tab Settings	QC Action (Event)	SM Action (Event)
Creation	Do nothing.	Create a corresponding record in the other endpoint.
Update	Update its corresponding record in the other endpoint.	Update its corresponding record in the other endpoint.
Deletion	Do nothing.	Do nothing.

The following screenshot shows the settings:

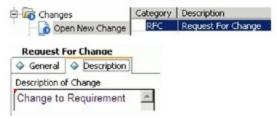


## Test the Link

To test the link:

**Note:** The following is only an example. The exact steps required on your system may differ significantly. The phase in which the tab for QC Integration appears may be different on your system.

- 1. Save the configuration (an integrity check is automatically run).
- 2. Click Enable Link.
- 3. Create a Service Manager change (the category of the change depends on each Service Manager customization; RFC is used in this example).



4. Change the phase to **Building**. The "QC Integration" tab appears.

Phase Name	Description
Assessment	Assessment
Building	RFC Building
<b>RFC</b> Implementation	Place Changes Into Service
RFC Testing	RFC Testing

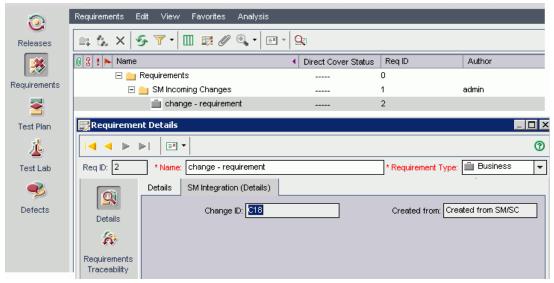
5. Select Forward as Requirement.

Backout Method	History	Approvals	Attachments	Related Records	♦ Workflow	QC Integration	27
	Forward to	QC:		1 - Forward as Requ	irement	Ŧ	
	Def/Req II	):					
			Server/Domain/Project:				

6. Synchronize.

🛞 Cancel Current Task 📄 View Report 🔸 Refresh Progress 🔽 Auto Refresh	
Running: Connecting to endpoint 1 Running: Connecting to endpoint 2 Running: Querying non filtered set Running: Handling endpoint 1 - Processing entity #1 of #1 in the Create list, (Total: passed = 0, failed = 0) Passed: Disconnecting Completed : Passed	

7. View the requirement in QC.



**Note:** In ALM 11, the **Change ID** and **Created from** fields reside on the Details tab of the requirement form.

# Chapter 8: SM Problem -> QC/ALM Defect

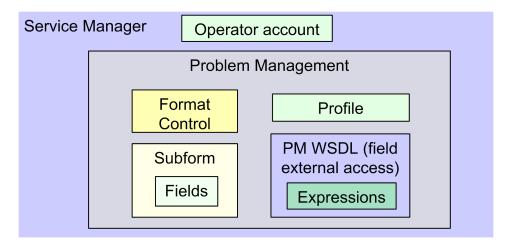
This chapter describes how to synchronize SM Problems with QC/ALM Defects ("Problem -> Defect").

This chapter includes:

- "Customizing Service Manager/ServiceCenter for Problem Management" below
- "Customizing the QC/ALM Defects Module" on page 116
- "Configuring Links in QC/ALM Synchronizer" on page 122

# Customizing Service Manager/ServiceCenter for Problem Management

The following diagram summarizes the components which require tailoring in Service Manager/ServiceCenter.



To customize Service Manager/ServiceCenter for Problem Management, perform the following tasks:

- 1. "Add Fields" on the next page
- "Specify the External Access Definition on Service Manager" on the next page or "Specify the External Access Definition on ServiceCenter" on page 103
- 3. "Create a Subform" on page 105

- 4. "Add the Subform to a Form" on page 108
- 5. "Add Format Control Calculations/Validations" on page 109

## Add Fields

Add the following required fields to the rootcause table. Do not change them.

	Туре				
Field	Service Manager	ServiceCenter			
qcintegration.type	Character	Text			
qcintegration.id	Number	Decimal			
qcintegration.project	Character	Text			

**Note:** The data type requirements for SM fields are described in "Matching Types" on page 31.

## Specify the External Access Definition on Service Manager

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have PD Content Pack 9.30.2 installed, follow the steps below to specify the External Access Definition on Service Manager:

Note: The following steps also apply to SM 9.4x Classic.

- Create a custom External Access Definition QCIntProblemService by clicking Tailoring > WSDL configuration on Service Manager 7.0x; or clicking Tailoring > Web Services > WSDL configuration on Service Manager 7.1x or later with the following values:
  - Service Name: QCIntProblemService
  - Name: rootcause
  - Object Name: QCIntProblem
  - Allowed Actions / Action Names:

- add / Create
- save / Update

External Access Definition								
Service Name: QCIntProblemService								
Name:	Name: rootcause			•	Obje	ect Name:	QCIntProblem	
Allowed Actions		Expressions	٠	Fields				
Allowed Actio	Allowed Actions Acti			Action Names		Action Ty	ре	
add	add Create							
save			Update					

**Note:** The above values are required (Do NOT change them).

2. Enable the required fields in the web service.

Field	Caption	Туре
id	ProblemID	StringType
sysmodtime	Modified	DateTimeType
qcintegration.id	QCEntityID	IntType
qcintegration.project	QCProject	StringType
qcintegration.type	QCIntegrationType	StringType
qcintegration.created.from	CreatedFrom	StringType
current.phase	CurrentPhase	StringType
category	WorkFlowType	StringType

External Acce	ess De	finition			
Service Name:	QCIntPr	oblemService		_	
Name:	rootcau	se		=	
Allowed Actio	ons 🗇	Expressions	🔶 F	ields	
Field		Caption		Туре	
qcintegration.	id	QCEntityID		IntType	
id		ProblemID		StringType	
sysmodtime	sysmodtime			DateTimeType	
qcintegration.	qcintegration.project			StringType	
incident.categ	incident.category			StringType	
subcategory		SubCategory		StringType	
product.type		ProductType	StringType		
problem.type		ProblemType		StringType	
initial.impact		Impact		StringType	
severity	severity			StringType	
description	description			StringType	
assignment	assignment			StringType	
ticket.owner	ticket.owner			StringType	
category		WorkFlowType	е	StringType	

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on). The above values are required (Do NOT change them).

3. Define expressions for the web service.

#### Note:

- a. Problem Management requires an activity update provided with each save and for better flow, and this activity update will be hard-coded with the following expressions.
- b. Expressions 1 to 4 are for fixing an update issue with the Problem Management web service. For more information, see *SCR 41399*.

N	0.	Expression
1		cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)

No.	Expression
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save

				unununununun arteratunun
Service Name:	QCIntProblemService			
Name:	rootcause	•	Object Name:	QCIntProblem
♦ Allowed Actions ♦ Expressions	♦ Fields			
Expressions				
	nup(\$rc.update);if same(update in \$L.file, u	ipdate in \$L.file.save) then (\$	need.to.update=true)	
cleanup(\$pm.activity);clea	nup(\$rc.update);if same(update in \$L.file, u file;if (denull(\$rc.update)={}) then (\$rc.upda		need.to.update=true)	
cleanup(\$pm.activity);clea \$rc.update=update in \$L.f			need.to.update=true)	
cleanup(\$pm.activity);clea \$rc.update=update in \$L.f	file;if (denull(\$rc.update)={}) then (\$rc.upda ie) then (\$rc.update={"QC update sent"})		need.to.update=true)	
cleanup(\$pm.activity);clea \$rc.update=update in \$L.f if (\$L.need.to.update=tru update in \$L.file=update in	file;if (denull(\$rc.update)={}) then (\$rc.upda ie) then (\$rc.update={"QC update sent"})		need.to.update=true)	

If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, follow the steps below to specify the External Access Definition on Service Manager:

Note: The following steps also apply to SM 9.4x Codeless.

- 1. Refer to step 1.
- 2. Enable the required fields in the web service.

Field	Caption	Туре
id	ProblemID	StringType
sysmodtime	Modified	DateTimeType
qcintegration.id	QCEntityID	IntType

#### **External Access Definition**

Service Name: Name:		* QCIntProblemService		Released:
Object Name:		QCIntProblem	QCIntProblem	
Allowed A	ctions Expressions Fields			
	Field	Caption	Турс	
	id	ProblemID	StringType	
	sysmodtime	Modified	DateTimeType	
	qcintegration.id	QCEntityID	IntType	
	qcintegration.project	QCProject	StringType	
qcintegration.type		QCIntegrationType	StringType	
	qcintegration.created.from CreatedFrom		StringType	

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on). The above values are required (Do NOT change them).

3. Define expressions for the web service.

**Note:** Problem Management requires an activity update provided with each save and for better flow, and this activity update will be hard-coded with the following expressions.

No.	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save
5	isKnownError in \$L.file=nullsub(isKnownError in \$L.file,true)

🔝 🐴 OK 💢 Cancel 💩 Previous 👘	Next 🜵 Add 💾 Save 🐨 Delete 🔍 Find 🗊	Fill	
① External Access Definition record upon	ated.		
External Access Definition			
Service Name:	QCIntProblemService	Released	
Name:	rootcause 👻	Deprecated	
Object Name:	QCIntProblem		
♦ Allowed Actions ♦ Expressions	♦ Fields		
Src.update=update in SL.file; if (SL.need.to.update=true) th update in SL.file=update in S	p(\$rc.update);if same(update in \$L.file, update in \$L.fil f (denull(\$rc.update)={}} then (\$rc.update={"QC updat en (\$rc.update={"QC update sent"}) .file.save ıb(isKnownError in \$L.file,true)		

## Specify the External Access Definition on ServiceCenter

To specify the External Access Definition on ServiceCenter:

 Click System Definition > Tables > rootcause > Fields and keys definitions for the rootcause table and change the settings of these fields.

No.	Field	Include in API	Field name in API	Field data type in API
1	id	Y	ProblemID	StringType
2	sysmodtime	Y	Modified	DateTimeType
3	qcintegration.id	Y	QCEntityID	IntType

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on).

- Click Menu navigation > Utilities > Tools > Document Engine > Process. Type rca.save in the Process Name field, and then click Search.
- 3. Change the name to rca.qcupdate and click Add.

#### 4. Append the following lines to the Initial Expressions tab.

No.	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save

#### 5. Click Save.

- 6. Update the rca.view state record.
  - a. Click Utilities > Tools > Document Engine > States. Type the state name rca.view and click
     Search.
  - b. Add the following row to the Non-base methods table, and then click Save.

<b>Display Action</b>	Process Name	Condition	Save First
qcupdate	rca.qcupdate	\$L.mode~="close" and \$L.mode~#"add"	false

- 7. Update the Problem Management integration web service.
  - a. Click Toolkit > WSDL Configuration. In the Name field type rootcause, and then click Search.
  - b. Update the External Access Definition as follows based on table rootcause.

No.	Field	Value
1	Service Name	QCIntProblemService
2	Object Name	QCIntProblem
3	Allowed Actions/Action Names	add/Create
4	Allowed Actions/Action Names	qcupdate/Update

#### **External Access Definition**

Service Name:		CIntProblemService			]	
Name: r		iuse	1	Q	Object Name:	QCIntProblem
owed Actions	Expressions	🔷 Data Policy				
Allowed Actions			Ac	tion Names		
add			Create			
qcupdate				Up	date	
	owed Actions Allowed Act add	owed Actions Allowed Actions add	Inclusion Toblember Vice       rootcause       owed Actions       Allowed Actions       add	Image: Second Following Web       Image:	Image: Allowed Actions     Image: Allowed Actions       Allowed Actions     Actions	Allowed Actions     Action Names       add     Create

Note: Delete all Allowed Actions without an Action Name.

## Create a Subform

To create a subform:

- 1. Create a global list.
  - a. Create a global list by clicking **Tailoring > Tailoring Tools > Global Lists** on Service Manager 7.0x or later; or clicking **Utilities > Tools > Global Lists** on ServiceCenter with the following

#### parameters:

No.	Parameter	Value	Remarks	
1	List Name	SMQC Integration PM Project List		
2	Regen Every	1 00:00:00		
3	Build List on Startup?	Yes	Check box	
4	List Variable	\$G.qcintegration.problem.project		
5	User Defined List?	Yes	Check box	
6	Value List	{"server1/domain1/project1", "server2/domain2/project2"}	Change to the values for your system	
			<b>Note:</b> No spaces between slashes.	

- b. Save this global list and click **Rebuild Global List** in the options menu.
- 2. Create a subform.

Click **Tailoring** > **Forms Designer** on Service Manager; or click **Toolkit** > **Forms Designer** on ServiceCenter to create the pm.qcint.subform subform with the following components:

Note: Click No when the system message "Do you want to use Form Wizard?" appears.

No.	Component	Properties
1	Label	Caption: Synchronize with QC:

No.	Component	Properties		
2	Combo Box	Input: qcintegration.type		
		Value List: 0;1;		
		Display List: 0 - Not Synchronize;1 - Synchronize with QC Defect		
		Select Only: Yes		
		Read-Only Condition: [\$qcint.type.readonly]		
3	Label	Caption: Defect ID:		
4	Text	Input: qcintegration.id		
		Read-Only: Yes		
5	Label	Caption: Server/Domain/Project:		
6	Combo Box	Input: qcintegration.project		
		Value List: \$G.qcintegration.problem.project		
		Read-Only Condition: [\$qcint.project.readonly]		
		Mandatory Condition: [qcintegration.type]>0		

V OK 🗱 Cancel	
	<b>\$</b>
Synchronize with QC:	
Defect ID:	
Server/Domain/Project	
Created from:	

### Add the Subform to a Form

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have installed PD Content Pack 9.30.2, follow the steps below to add the subform you created to a form:

Note: The following steps also apply to SM 9.4x Classic.

- From Forms Designer, open the default form of a Problem Management phase (PM.pc.ident.and.class is used as an example on ServiceCenter 6.2/Service Manager 7.0x).
- 2. Add a notebook Tab with the QC Integration caption.
- Add a subform to the new tab with format pm.qcint.subform, which is displayed in the following screenshot:

😫 *Forms Designer: 🗙 👺 Problem Managemen 🛛 👺 rootcausephase: 🎽 🔭 🗆	Properties 🛛	- 8
	Subformat	
	Property	Value
🔟 🔟 📑 🛥 🔜 🧮 🔰 🗇 🍢 🛗 🖏 👗 🚺	Display Blank	
	Display Using Table	
Problem Control - Problem Investigation and Diagnosis	Format	pm.qcint.subform
Record Number:	Height	40
	Input	To and the first lists
Expected Resolution E	Name	
Brief Description:	Virtual Join	
	Visible	
SLA & Related Records & Tasks & History & Workflow & QC Integration *s	Visible Condition	
	Width	143
	X	3
	Y	1

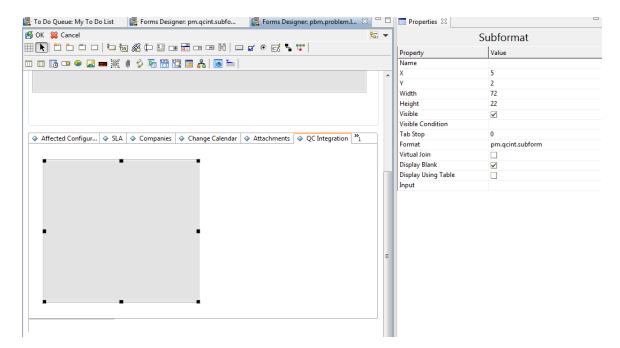
4. Save the changes.

**Note:** If the error message "Format 'pm.qcint.subform' not found (display, show.rio)" appears, restart the SM server to enable the subform.

If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, follow the steps below to add the subform you created to a form:

Note: The following steps also apply to SM 9.4x Codeless.

- From Forms Designer, open the default form of a Problem Management phase (pbm.problem.logging is used as an example on Service Manager 9.3x).
- 2. Add a notebook Tab with the QC Integration caption.
- 3. Add a subform to the new tab with format pbm.problem.logging, which is displayed in the following screenshot:



4. Save the changes.

## Add Format Control Calculations/Validations

**Note:** Steps in this topic are applicable if the Service Manager Process Designer (PD) Content Pack is not installed, or if you have SM 9.3x with PD Content Pack 9.30.2 installed, or if you are using SM 9.4x Classic. Otherwise, skip this topic and refer to "Add Rule Set Calculations/Validations" on the next page for configurations in the SM 9.3x with PD Content Pack 9.30.3 and SM 9.4x Codeless environment.

To add format control calculations and validations, follow the steps below:

 Open the format control record of the previous Problem Management form PM.pc.ident.and.class. Installation and Administration Guide Chapter 8: SM Problem -> QC/ALM Defect

#### 2. Click Calculations.

#### 3. Add two rows with the following values:

Record	Parameter	Value
1	display	true
	initial	true
	calculation	<pre>\$qcint.type.readonly=2;if (qcintegration.type in \$file~=0) then (\$qcint.type.readonly=1)</pre>
2	display	true
	initial	true
	calculation	<pre>\$qcint.project.readonly=2;if (qcintegration.type in \$file~=0 and not null (qcintegration.project in \$file)) then (\$qcint.project.readonly=1)</pre>

#### 4. Click Validations.

5. Add one row with the following values:

No.	Parameter	Value
1	Validation	not null(qcintegration.project in \$file)
2	Message	The Server/Domain/Project is required.
3	Add	qcintegration.type in \$file~=0
4	Update	qcintegration.type in \$file~=0
5	Set Focus to	qcintegration.project

6. Save your changes.

## Add Rule Set Calculations/Validations

Note: Steps in this topic are applicable for the Service Manager 9.3x with Process Designer (PD) Content Pack 9.30.3 and SM 9.4x Codeless only. Otherwise, refer to "Add Format Control Calculations/Validations" on the previous page for configurations in Service Manager 9.3x non-PD, 9.3x with PD 9.30.2, and SM 9.4x Classic environment.

#### To add rule set calculations and validations, follow the steps below:

- 1. "Copy an Existing Workflow" below
- 2. "Associate an Existing Problem Category with the New Workflow" below
- 3. "Create New Rule Set for Initialization and Validation" on the next page
- 4. "Associate the New Workflow with the New Rule Set" on page 114

### Copy an Existing Workflow

You can use copies of the existing workflows in another business process, or make changes to the HP proprietary workflow copies.

To copy an existing workflow, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Copy Existing Workflow**.
- 2. On the Clone a Workflow page, select the workflow you want to copy. For example, Problem.
- 3. Type SMQCIntPbM in the New workflow name field.

Т	o Do Qu	ieue: My To Do List	Wizard: Clone a Workflow 🗷		
	Clo	ne a Workflov	N		
			Please specify the new worki	flow name, as well	as the prefix for new rule sets if they are to be copied as well
			New	v workflow name:	* SMQCIntPbM
				Copy rule sets?	

- 4. Select the Copy rule sets check box if you want to copy rule sets, and then type a rule set prefix.
- 5. Click **OK**.

The newly copied workflow appears in the list on the Clone a Workflow page.

### Associate an Existing Problem Category with the New Workflow

You can update existing problem categories, subcategories and areas and associate the updated categories with the new workflow so that they can be used in another business process. For more information, refer to *HP Service Manager – Process Designer Content Pack Administrator's Guide*.

To associate an existing change category with the new workflow, follow the steps below:

- 1. From the System Navigator, click **Problem Management > Configuration > Problem Categories**.
- 2. Click Search.
- 3. Select the problem category for which you want to add a workflow. For example, problem.
- 4. In the Problem Category page, remove the currently assigned workflow from the Workflow field.
- 5. Type SMQCIntPbM in the workflow field.
- 6. Click **Save** to associate the problem category with the workflow.

Name:	problem	Apply To:	Problem	
Active:				
Description:	incident			
Workflow:	* SMQCIntPbM	₫°Q.		
Subcategories Workflow				(
		\$		
	<b>*</b>			
Logging	Categorization	Resolution Review	Closure	
Logging –			Closure	
Logging			Closure	

### Create New Rule Set for Initialization and Validation

To create a new rule set for initialization, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Rule Sets**.
- 2. Type the values as follow:

Field	Value
ID	pbm.alm.int.init
Available as action	False

Field	Value
Name	Initialize for ALM integration in the Problem Record
Table name	rootcause
HP Proprietary	

#### 3. Click New and Save.

#### 4. Click Add Rule.

- 5. In the Select Rule Type page, click Run JavaScript.
- 6. In the Run JavaScript page, type the values as follow:

Field	Value
Rule Description	Run Javascript for initializing Integration type and project in the Problem Record
Statement	<pre>vars['\$qcint.type.readonly'] = 2; vars['\$qcint.project.readonly'] = 2; var _null=system.functionsnull; var file = vars.\$L_file; if(file["qcintegration.type"] != 0 &amp;&amp; !_null(file["qcintegration.type"])) { vars['\$qcint.type.readonly'] = 1 } if(file["qcintegration.type"] != 0 &amp;&amp; !_null(file["qcintegration.project"])) {</pre>
	vars['\$qcint.project.readonly'] = 1 }

#### 7. Click **OK**.

#### 8. Click **Save** and **Exit**.

To create a new rule set for validation, follow the steps below:

#### 1. From the System Navigator, click **Tailoring > Process Designer > Rule Sets**.

2. Type the values as follow:

Field	Value
ID	pbm.alm.int.validation
Available as action	False
Name	Validation for ALM integration in the Problem Record
Table name	rootcause
HP Proprietary	

- 3. Click New and Save.
- 4. Click Add Rule.
- 5. In the Select Rule Type page, click **Set Mandatory Fields**.
- 6. Refer to step 6 to step 13 as described in "Create New Rule Set for Initialization and Validation" on page 61.

### Associate the New Workflow with the New Rule Set

To associate the new workflow with the new rule set, follow the steps below:

- 1. From the System Navigator, click **Problem Management > Configuration > Workflows**.
- 2. Select SMQCIntPbM in the workflows list.
- 3. Select the first phase in the workflow graph.
- 4. Click **Rule Sets** tab > **Initialization** tab.
- 5. Click **Add** and select the pbm.alm.int.init rule set you just created.

MP Service Manager								User: falcon
		To Do Queue: My To Do List Workflows	Work	flow: SMQCIntPbM 🛞				
🖬 🔂 🚭 👘 🖓	~	💾 Save 🔍 Zoom in 🔍 Zoom out 🗌 Add j	phase	Delete Workflow properties				🖴 🗖 I
Request Management Service Catalog	^							
Service Desk Service Level Management				<b> </b>	<b>\$</b>			
System Administration Tailoring		Logging	, ]-t	x→→ Investigation ☆→ Resolution ☆	Review Closure			
awormg ▷ Audit ▷ Differential Upgrade ▷ Document Engine		èti						
Event Services		Abandonme	Add	Rule Sets - Initialization				
Knowledge Engineering				Id	Name			
Notifications			m	apply.template	Apply Template		^	
Process Designer				create.template	Create Template from Record		Ξ	
<ul> <li>Configuration</li> <li>Copy Existing Workflow</li> </ul>				pbm.abandon.wizard	Run Abandon Wizard		-	
Export Workflow	Ш.	Phase - Logging		pbm.abandonment	PBM abandonment			
Rule Sets		Details Forms Rule Sets Actions		pbm.abandonment.field.validation	PBM field validation at abandonment phase			
Workflows	=			obm alm init init	Initialize for ALM integration in the Problem Record	٦		
SQL Utilities		On enter On exit [Initialization] O		pbm.alm.int.validation	Validation for ALM integration in the Problem Reco			
Tailoring Tools		🖨 Add 🗑 Delete 🙃 View 🏫 Up 😺 Down		pbm.bac.pl.assciate	Associate with BAC PI			
Web Services		Rule Sets		pbm.calculate.rc.calendar	Calculate whether RC calendar needs to be displa	ved		
Codes Database Dictionary				pbm.categorization.init.status	Initialize om status at categorization phase	100		
Database Dictionary Database Manager				poinceregonization.inceration	minanzo prinaratoa ar categorization priase		٣	
Data Policy					ок	Cancel		
Format Control								

- 6. Click **OK**.
- 7. Click Rule Sets tab > On display tab.
- 8. Repeat step 5 and 6.
- 9. Click **Rule Sets** tab > **On enter** tab.
- 10. Click **Add** and select the pbm.alm.int.validation rule set you just created.

MP Service Manager					
		To Do Queue: My To Do List Workflows	Workflow: SMQCIntPbM 🗵		
12 12 23	~	💾 Save   🍕 Zoom in 🤤 Zoom out   🗖 A	Add phase 🐨 Delete   🚰 Workflow properties		
- Request Management	^				
Service Catalog					
Service Desk					
Service Level Management			<b>\$</b>	¢ \$	
System Administration					
		Logging Categoriza	ation 🗘 Nrvestigation ⊅	Resolution 🗘 Review Closure	
Tailoring	_	a			
Audit					
<ul> <li>Differential Upgrade</li> <li>Document Engine</li> </ul>					
<ul> <li>Document Engine</li> <li>Event Services</li> </ul>		Abando	Add Rule Sets - On enter		
<ul> <li>Event Services</li> <li>Knowledge Engineering</li> </ul>					
<ul> <li>Notifications</li> </ul>			ld Id	Name	
Process Designer			apply.template	Apply Template	-
<ul> <li>Process Designer</li> <li>Configuration</li> </ul>			create.template	Create Template from Record	=
Copy Existing Workflow			pbm.abandon.wizard	Run Abandon Wizard	
Export Workflow		Phase - Logging	pbm.abandonment	PBM abandonment	
Rule Sets		Details Forms Rule Sets Actions		PBM field validation at abandonment phase	
Workflows	E		bin abandoninen: neid vandation		
SQL Utilities		On enter On exit Initialization	On ( Contraction of the second	Initialize for ALM integration in the Problem Record	_
Tailoring Tools		🖨 Add 💮 Delete 😚 View 🏫 Up 🐺 D	pbm.alm.int.validation	Validation for ALM integration in the Problem Record	
> Web Services			pbm.bac.pi.assciate	Associate with BAC PI	
Codes		Rule Sets	pbm.calculate.rc.calendar	Calculate whether RC calendar needs to be displayed	
Database Dictionary			pbm.categorization.init.status	Initialize pm status at categorization phase	
Database Manager					
Data Policy				OK Can	cel
Format Control					
Forms Designer					
Integration Manager					
Localization Utility					

- 11. Click **OK**.
- 12. Click Save.

# Customizing the QC/ALM Defects Module

The steps for customizing the Defects module vary with different QC versions.

- "On QC 10 and Earlier" below
- "On ALM 11" on page 120

## On QC 10 and Earlier

To customize the Defects module on Quality Center 10 or earlier, perform the following tasks:

- 1. "Add Fields" below
- 2. "Add Tabs" on the next page
- 3. "Add Fields to Tabs" on page 118

### Add Fields

To add the required fields for Defect module customization:

- 1. Log on to QC as a project administrator.
- 2. Click **Tools / Customize**. The "QC Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (*XX* and *XY* are sequential numbers auto-generated by QC).

Field Name	Field Label	Field Type	Remarks
BG_USER_XX	Problem ID	String	
BG_USER_XY	Created from	String	

The following figure shows an example project entity.

Quality Center - Project Customization									
User Properties Project Users Groups Module Access Project Entities Requirement Types Risk-Based Quality Project Lists Automail Alert Rules	Project Customization Project Customization Project Entities Project Entities Defect User Fields User Fields User Fields Forward as problem Rec Problem ID	Field Settings Field Name: Field Label: Field Type: Field Length:	BG_USER_01) Forward as problem Lookup List						
Workflow	1 :	History	🗌 Required						
		Masked	Searchable						

**Note:** The data type requirements for QC fields are described in "Matching Types" on page 31.

### Add Tabs

To add tabs to the Defect form and show fields on these tabs:

1. In "QC - Project Customization", click **Workflow** > **Script Editor**.

#### 2. Select Defects module script.

Quality Center - Project Customization

User Propertie Project Users		Norkflow
Groups Module Acces Project Entitie Resuirement Risk-Based G Project Lists Automail	s Types	Script Generator - Add Defect Field Customization Enables you to customize the fields displayed for each user group in the Add Defects dialog bo You can also specify field order and whether a field is required. Script Generator - Defect Details Field Customization Enables you to customize the fields displayed for each user group in the Defect Details dialog box. You can also specify field order and whether a field is required.
Alert Rules Workflow		<u>Script Editor</u> Enables you to write VBScript code for all Quality Center modules. You can also use the Script Editor to modify the scripts generated by the above tools.
Script Editor	Toolbar Butto	Editor GetNewBugPageName

Add the following code to the GetNewBugPageName event procedure (which is triggered before QC opens the Add Defect dialog box).

```
select case PageNum
case "2"
GetNewBugPageName = "SM Integration (New)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For a new bug, the tab name is SM Integration (New).

4. Add the following code to the **GetDetailsPagename** event procedure (which is triggered before QC displays the Defect Details dialog box).

```
select case PageNum
case "2"
GetDetailsPageName = "SM Integration (Details)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For an existing defect, the tab name is SM Integration (Details).

### Add Fields to Tabs

To add fields to tabs:

1. In "QC - Project Customization", click **Workflow** > **Script Editor**.

#### 2. Select Defects module script.

E E Defects module script

Quality Center - Project Customization

User Properties Project Users	Workflow
Groups Module Access Project Entities Requirement Types Risk-Based Quality Man Project Lists Automail	Script Generator - Add Defect Field Customization Enables you to customize the fields displayed for each user group in the Add Defects dialog bo You can also specify field order and whether a field is required. Script Generator - Defect Details Field Customization Enables you to customize the fields displayed for each user group in the Defect Details dialog box. You can also specify field order and whether a field is required.
Alert Rules Workflow	<u>Script Editor</u> Enables you to write VBScript code for all Quality Center modules. You can also use the Script Editor to modify the scripts generated by the above tools.
Script Editor Script Editor Toolbar Bu	ton Editor GetNewBugPageName
Workflow Scripts     Ormon script     Sequirements n     E    Est Plan modu     E    Est Lab modul     E    Manual Runner	script - Sug_FieldChange script - Sug_CanPost

3. If **WizardFieldCust\_Details** and **WizardFieldCust\_Add** are not found in the list, do the following to generate these two methods.

Bug\_AfterPost SetFieldApp

- a. Script Generator Add Defect Field Customization
- b. Script Generator Defect Details Field Customization



 Add the following code to the WizardFieldCust\_Details event procedure. SetFieldApp "BG\_USER\_XX", True, False, 1, 0

SetFieldApp "BG\_USER\_XY", True, False, 1, 1

The parameter values are:

Installation and Administration Guide Chapter 8: SM Problem -> QC/ALM Defect

- Field name (BG\_USER\_XX, where XX consists of two digits)
- Visible (True)
- Required (False)
- Page number (start from Ø)
- View order (start from Ø)
- Add the following code to the WizardFieldCust\_Add event procedure. SetFieldApp "BG\_USER\_XX", True, False, 1, 0

```
SetFieldApp "BG_USER_XY", True, False, 1, 1
```

6. Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

Bug\_Fields.Field("BG\_USER\_XX").IsReadOnly=True

Bug\_Fields.Field("BG\_USER\_XY").IsReadOnly=True

7. Save your changes.

## On ALM 11

On ALM 11, you only need to add new fields directly to the Details tab of the Defect form.

To customize the ALM Defects module, perform the following task:

• "Add Fields" below

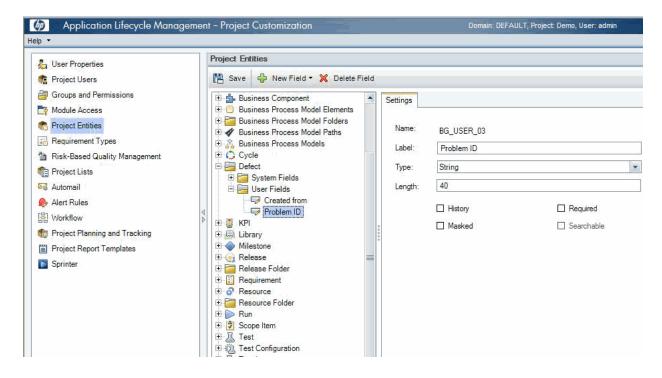
### Add Fields

To add the required fields for Defect module customization:

- 1. Log on to ALM as a project administrator.
- Click Tools / Customize. The "Application Lifecycle Management Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (XX and XY are sequential numbers

#### auto-generated by ALM).

Field Name	Field Label	Field Type
BG_USER_XX	Problem ID	String
BG_USER_XY	Created from	String



**Note:** The data type requirements for QC/ALM fields are described in "Matching Types" on page 31.

- 4. Click Workflow > Script Editor.
- 5. Select Defects module script.
- Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

```
Bug_Fields.Field("BG_USER_XY").IsReadOnly=True
```

Bug\_Fields.Field("BG\_USER\_XZ").IsReadOnly=True

7. Save your changes.

# Configuring Links in QC/ALM Synchronizer

To configure and test a link in the QC/ALM synchronizer, perform the following tasks:

- 1. "Specify Endpoints / Type of Link" below
- 2. "Define Field Mappings" below
- 3. "Define Events" on the next page
- 4. "Test the Link" on page 124

## Specify Endpoints / Type of Link

Specify the connection properties as described in "Create a Link" with the following settings specific for this type of link:

- 1. Step 1: "Endpoint 2 type" = SM ProblemManagement.
- Step 3: "Service URL" = http://service\_manager\_ host>:<port>/sc62server/PWS/QCIntProblemService.wsdl
- 3. Step 4: "Select entity types" = Problem by Defect (this is the only available selection).

## **Define Field Mappings**

If the Service Manager Process Designer (PD) Content Pack is not installed, or if you are working with SM 9.4x Classic, see the following summary for basic field mappings:

QC	Direction	SM	Constant value	Remarks
Problem ID	<-	ProblemNumber		
Defect ID	->	QCEntityID		Synchronize back on create: Yes
Created from			Created from SM/SC	

Example field mappings are shown in the following screenshot:

Mappe	Mapped Fields					
Туре	QC Field	Direction	SM ChangeManagement Field			
	Severity	<>	Urgency			
	Problem ID	<	ProblemI D			
	Defect ID	>	QCEntityID			
	Summary	<>	Description			
$\langle \rangle$	Created from	<	Value: Created from SM/SC			

If the Service Manager Process Designer (PD) Content Pack is installed, or if you are working with SM 9.4x Codeless, status drive the business process of Help Desk in Process Designer Content Pack 9.30.3 and SM 9.4x Codeless. You need to pay attention to the value mapping about the **Status** field between Service Manager and ALM.

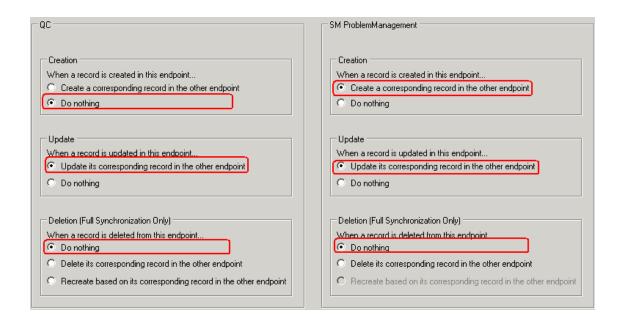
## **Define Events**

The following table lists the event settings for the two endpoints:

Events Tab Settings	QC Action (Event)	SM Action (Event)
Creation	Do nothing.	Create a corresponding record in the other endpoint.
Update	Update its corresponding record in the other endpoint.	Update its corresponding record in the other endpoint.
Deletion	Do nothing.	Do nothing.

The following screenshot displays the settings:

#### Installation and Administration Guide Chapter 8: SM Problem -> QC/ALM Defect



## Test the Link

To test the link:

#### Note:

- 1. A mandatory field (in either SM and QC) does not accept a null value. Synchronization may fail if a mandatory field is mapped to a field that can be null.
- 2. The following sample steps are for your reference only. The exact steps required on your system may differ significantly. The phase in which the QC Integration tab appears may be different on your system.
- 1. Save the configuration (an integrity check is automatically run).
- 2. Click Enable Link.

3. Create a Problem and select **Synchronize with QC Defect**.

9	🕖 Problem PM0016 has bee	n opened.		$\langle p \rangle$
	Problem Control - Proble	em Identification and Classification		
	Record Number:	PM0016 Status: Ope Expected Resolution Date:	en 🔽	
	Brief Description:	problem to defect		
	Classification	♦ Attachments         ♦ Related Records         ♦ History         ♦ Workflow         ♦ QC Integration		
	Synchronize with Q	C: T - Synchronize with QC Defect		
	Defect ID:			
	Server/Domain/Pro	ject localhost/DEFAULT/Demo		

4. Synchronize.

🛞 Cancel Current Task 📄 View Report	😯 Refresh Progress 🔽 Auto Refresh
Running: Querying non-filtered set Running: Handling endpoint 1 - Processing Passed: Disconnecting Completed : Passed	entity #1 of #1 in the Create list, (Total: passed = 0, failed = 0)

5. View the Problem in SM.

Record Number:	Record Number:			Status:		l	Updated	•
				Exp	ected Resolutic	in Date:		-
Brief Description	:	problem to	defect					
Classification	Activities	Attachments	Related Records	🔷 History	🔷 Workflow	🔷 QC Integratio	on	
Sync	hronize with QC	:	1 - S	ynchronize wi	th QC Defect			
	ect ID:		21					

6. View the defect in QC.

**Note:** In ALM 11, the Problem ID field and the Created from field reside on the Details tab of the Defect form.

# Chapter 9: QC/ALM Defect -> SM Problem

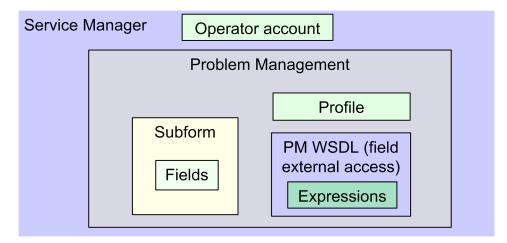
This chapter describes how to synchronize QC/ALM Defects with SM Problems ("Defect -> Problem ").

This chapter includes:

- "Customizing Service Manager/ServiceCenter for Problem Management" below
- "Customizing the QC/ALM Defects Module" on page 139
- "Configuring Links in QC/ALM Synchronizer" on page 147

# Customizing Service Manager/ServiceCenter for Problem Management

The following diagram summarizes the components which require tailoring in Service Manager/ServiceCenter.



To customize Service Manager/ServiceCenter for Problem Management, perform the following tasks:

- 1. "Add Fields" on the next page
- "Specify the External Access Definition on Service Manager" on the next page or "Specify the External Access Definition on ServiceCenter" on page 131

- 3. "Create a Subform" on page 136
- 4. "Add the Subform to a Form" on page 137

## Add Fields

Add the following required fields to the rootcause table. Do not change them.

	Туре			
Field	Service Manager 7.0x or later	ServiceCenter		
qcintegration.type	Character	Text		
qcintegration.id	Number	Decimal		
qcintegration.project	Character	Text		
qcintegration.created.from	Character	Text		

Note: The data type requirements for SM fields are described in "Matching Types" on page 31.

## Specify the External Access Definition on Service Manager

To specify the External Access Definition on Service Manager:

- Create a custom External Access Definition QCIntProblemService by clicking Tailoring > WSDL configuration on Service Manager 7.0x; or clicking Tailoring > Web Services > WSDL configuration on Service Manager 7.1x or later with the following values:
  - Service Name: QCIntProblemService
  - Name: rootcause
  - Object Name: QCIntProblem
  - Allowed Actions / Action Names:
    - $\circ~$  add / Create
    - save / Update

External Access Definition					
Service Name:	QCIntPro	blemService			
Name: rootcause		•	<ul> <li>Obje</li> </ul>	ect Name:	QCIntProblem
Allowed Act	tions 🔷	Expressions	🔷 Fields		
Allowed Actions		Action Names		Action Typ	ре
add		Create			
save		Update			

**Note:** The above values are required (Do NOT change them).

2. Enable the required fields in the web service.

Field	Caption	Туре
id	ProblemID	StringType
sysmodtime	Modified	DateTimeType
qcintegration.id	QCEntityID	IntType
qcintegration.project	QCProject	StringType
qcintegration.type	QCIntegrationType	StringType
qcintegration.created.from	CreatedFrom	StringType
current.phase	CurrentPhase	StringType
category	WorkFlowType	StringType

External Access Definition				
Service Name: QCIntF	ProblemService	_		
Name: rootca	use	=		
Allowed Actions	🕨 Expressions 🛛 🧇	Fields		
Field	Caption	Туре		
qcintegration.id	QCEntityID	IntType		
id	ProblemID	StringType		
sysmodtime	Modified	DateTimeType		
qcintegration.project	QCProject	StringType		
incident.category	Category	StringType		
subcategory	SubCategory	StringType		
product.type	ProductType	StringType		
problem.type	ProblemType	StringType		
initial.impact	Impact	StringType		
severity	Severity	StringType		
description	Description	StringType		
assignment	AssignmentGroup	StringType		
ticket.owner	ProblemOwner	StringType		
category	WorkFlowType	StringType		

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on). The above values are required (Do NOT change them).

3. Define expressions for the web service .

#### Note:

- a. Problem Management requires an activity update provided with each save and for better flow, and this activity update will be hard-coded with the following expressions.
- b. Expressions 1 to 4 are for fixing an update issue with the Problem Management web service. For more information, see *SCR 41399*.

No	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)

#### Installation and Administration Guide Chapter 9: QC/ALM Defect -> SM Problem

No	Expression
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save

iervice Name:		QCIntProblemService			
Jame:		rootcause	-	Object Name:	QCIntProblem
Allowed Act	tions 🗇 Expressions	♦ Fields			
Evi	proceiope				
1 6 4	Expressions cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)				
cle		in(\$rc undate);if same(undate in \$1_file_ur	date in \$1. file save) then (	\$1. need to undate=true)	
	eanup(\$pm.activity);cleanu		. , ,	\$L.need.to.update=true)	
\$rc	eanup(\$pm.activity);cleanu c.update=update in \$L.file	;if (denull(\$rc.update)={}) then (\$rc.upda	. , ,	\$L.need.to.update=true)	
\$rc if (	eanup(\$pm.activity);cleanu c.update=update in \$L.file (\$L.need.to.update=true)	;if (denull(\$rc.update)={}) then (\$rc.upda then (\$rc.update={"QC update sent"})	. , ,	\$L.need.to.update=true)	
\$rc if () up(	c.update=update in \$L.file (\$L.need.to.update=true) date in \$L.file=update in \$	;if (denull(\$rc.update)={}) then (\$rc.upda then (\$rc.update={"QC update sent"})	. , ,	\$L.need.to.update=true)	

## Specify the External Access Definition on ServiceCenter

To specify the External Access Definition on ServiceCenter:

1. Click System Defintion > Tables > rootcaus	e > Fields and keys definitions for the rootcause	<b>table</b> and change the settings of these fields.
---	---	---

No.	Field	Include in API	Field name in API	Field data type in API	Remarks
1	id	Υ	ProblemID	StringType	
2	sysmodtime	Υ	Modified	DateTimeType	
3	qcintegration.id	Y	QCEntityID	IntType	
4	qcintegration.project	Y	QCProject	StringType	
5	qcintegration.type	Y	QCIntegrationType	StringType	
6	qcintegration.created.from	Y	CreatedFrom	StringType	
7	current.phase	Y	CurrentPhase	StringType	
8	category	γ	WorkflowType	StringType	

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on).

- 2. Create a new Process record named rca.qcupdate.
  - a. Click Utilities > Tools > Document Engine > Process. Type Process Name rca.save and then click Search.
  - b. Change the name to rca.qcupdate and click Add.

#### c. Append the following lines to the Initial Expressions tab, and click **Save**.

No.	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save

	-		
Process Name: rca.qcupdate			
Save Cursor Position?	1		
Run in Window? Window Title:	]		
♦ Initial Expressions ♦ Initial Javascript ♦ RAD ♦ Final Expressions ♦ Final Javascript ♦ Next Process			
if same(nullsub(full.name in \$G.rc.environment, full.name in \$G.rc.global.environment), true) then (\$L.operator=nullsub(\$lo.ufname, nullsu			
\$L.stamp=str(tod())+" ("+\$L.operator+"):"			
if (\$rc.update={} or \$rc.update={""}) then (\$rc.update=NULL); if (\$kne.update={} or \$kne.update={""}) then (\$kne.update=NULL); if (\$p.			
if (update in \$L.file=NULL) then (update in \$L.file={""})			
if (\$G.bg and not null(\$G.bg.activity.type)) then (\$rc.update=nullsub(\$rc.update, \$G.bg.activity.text);\$rc.update=nullsub(\$rc.update			
if (filename(\$L.file)~="rootcausetask") then if (status in \$L.file="Past Due" and expected.resolution.time in \$L.file>tod()) then (status in \$L			
\$L.save.status=status in \$L.file	_		
if (status in \$L.file="Open" or status in \$L.file="Reopened" and \$reopen.flag=false) then (status in \$L.file="Updated")	_		
\$reopen.flag=false	_		
cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)			
<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>			
if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})			
update in \$L.file=update in \$L.file.save	update in \$L.file=update in \$L.file.save		

#### 3. Update the rca.view state record.

a. Add the following row to the Non-base methods table, and then click Save.

Display Action	Process Name	Condition	Save First
qcupdate	rca.qcupdate	\$L.mode~="close" and \$L.mode~#"add"	false

- 4. Update the Problem Management integration web service.
  - a. Click **Toolkit** > WSDL Configuration. In the Name field type <code>rootcause</code>, and then click Search.

b. Update the External Access Definition as follows based on table rootcause.

No.	Field	Value
1	Service Name	QCIntProblemService
2	Object Name	QCIntProblem
3	Allowed Actions/ Action Names	add/Create
		qcupdate/Update

External Access Definition					
Service Name:	CIntProblemService				
Name:	rootcause	🛃 🔾 Object Name:	QCIntProblem		
Allowed Actions     A Expressions     A Data Policy					
Allowed Actions Action Names					
add		Create			
qcupdate		Update			

**Note:** Delete all Allowed Actions without an Action Name.

## Create a Subform

To create a subform:

Click **Tailoring > Forms Designer** on Service Manager7.0x or later; or click **Toolkit > Forms Designer** on ServiceCenter to create the pm.qcint.subform subform with the following components:

Note: Click No when the system message "Do you want to use Form Wizard?" appears.

No.	Component	Properties
1	Label	Caption: Defect ID:
2	Text	• Input: qcintegration.id
		• Read-Only: Yes
3	Label	Caption: Server/Domain/Project:
4	Text	Input: qcintegration.project
		• Read-Only: Yes
5	Label	Caption: Created from:
6	Text	Input: qcintegration.created.from
		• Read-Only: Yes

Main Menu: falcon Forms Designer: pm.qcint.subform 🗙	
🔝 🗸 OK 🗱 Cancel 🖳 Delete 😽 Design	8
Defect ID:	I.
Server/Domain/Project:	
Created from:	

### Add the Subform to a Form

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have PD Content Pack 9.30.2 installed, follow the steps below to add the subform you created to a form:

Note: The following steps also apply to SM 9.4x Classic.

- From Forms Designer, open the default form of a Problem Management phase (PM.pc.ident.and.class is used as an example in ServiceCenter 6.2/Service Manager 7.0x).
- 2. Add a notebook tab with the QC Integration caption.
- Add a subform to the new tab with format pm.qcint.subform, which is displayed in the following screenshot:

🔓 *Forms Designer: 🗙 👺 Problem Managemen 🛛 👺 rootcausephase: 🎽 🔭 🗆 🖓	Properties 🛛	- 8
	Subfo	ormat
	Property	Value
III III 🕞 🚥 🗶 🛲 🧮 🔰 🖇 🌄 🖧 🚺	Display Blank	
	Display Using Table	
Problem Control - Problem Investigation and Diagnosis	Format	pm.qcint.subform
Record Number:	Height	40
	Input	To and the first lists
Expected Resolution E	Name	
Brief Description:	Virtual Join	
	Visible	
♦ SLA ♦ Related Records ♦ Tasks ♦ History ♦ Workflow ♦ QC Integration *s	Visible Condition	
	Width	143
	X	3
	Y	1

4. Save the changes.

**Note:** If the error message "Format 'pm.qcint.subform' not found (display, show.rio)" appears, restart the SM server to enable the subform.

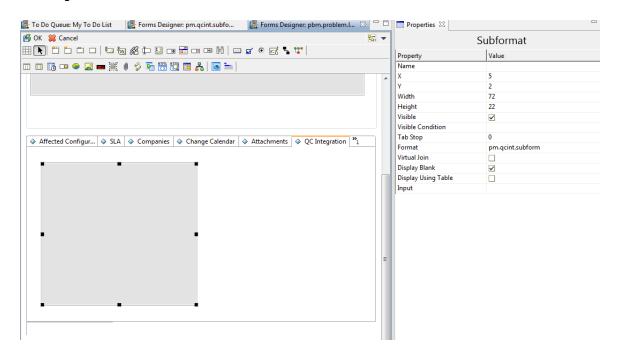
If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, follow the steps below to add the subform you created to a form:

**Note:** The following steps also apply to SM 9.4x Codeless.

 From Forms Designer, open the default form of a Problem Management phase (pbm.problem.logging is used as an example on Service Manager 9.3x).  Add a Notebook tab with the QC Integration caption and then set the value of Visible Condition to [isKnownError]<>true.

🚆 To Do Queue: My To Do List 🛛 📓 Forms Designer: pm.qcint.subfo 🛛 📓 Forms Designer: pbm.problem.l 🕮 🗖		■ Properties XX	
·····································	•	Ν	lotebook Tab
III 🔪 🗂 🗂 🗆 🛛 🏷 🏭 🏀 III 🚥 🚟 🖙 🖼 🕅 🗖 🖬 😴 🔍 💽 🧏 🚏		Property	Value
□□□ 🕞 □■ 🗶 🛲 🧮 ≬ 🗇 🖓 🔚 📆 🎇 🖬 🐥 🚺 🖬 늘		Name	
		Visible	$\checkmark$
		Visible Condition	[isKnownError]<>true
		Caption	QC Integration
		Caption Condition	
		Accessible Name	
		Accessible Description	
		Default to Expanded	
♦ Affected Configur ♦ SLA ♦ Companies ♦ Change Calendar ♦ Attachments ♦ QC Integration <sup>≫</sup> 1	Е		

3. Add a subform to the new tab with format pbm.problem.logging, which is displayed in the following screenshot:



4. Save the changes.

# Customizing the QC/ALM Defects Module

The steps for customizing the Defects module vary with different QC versions.

- "On QC 10 and Earlier" below
- "On ALM 11" on page 144

## On QC 10 and Earlier

To customize the Defects module on Quality Center 10 or earlier, perform the following tasks:

- 1. "Add Fields" below
- 2. "Add Tabs" on the next page
- 3. "Add Fields to Tabs" on page 141
- 4. "Create a View" on page 143
- 5. "Verify" on page 144

### Add Fields

To add the required fields for Defect module customization:

- 1. Log on to QC as a project administrator.
- 2. Click Tools / Customize. The "QC Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (*XX* and *XY* are sequential numbers auto-generated by QC).

Field Name	Field Label	Field Type	Remarks
BG_USER_ <i>XX</i>	Synchronize with SM Problem	Lookup List/YesNo	Select the "Verify Value" check box
BG_USER_ <i>XY</i>	Problem ID	String	

Quality Center - Project Customization						
<u>User Properties</u> Project Users <u>Groups</u>	Project Entities	Field Settings				
Module Access Project Entities Requirement Types Risk-Based Quality Project Lists Automail Alert Rules Workflow	Defect     Defect     System Fields     User Fields     One Change id     One C	Field Name: Field Label: Field Type: Field Length:	BG_USER_01 Forward as problem Lookup List 40	*		
		Masked	Searchable			

**Note:** The data type requirements for QC fields are described in "Matching Types" on page 31.

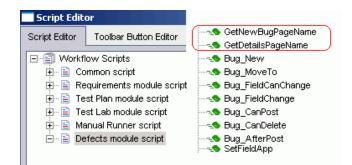
### Add Tabs

To add tabs to the Defect form and show fields on these tabs:

- 1. In "QC Project Customization", click **Workflow** > **Script Editor**.
  - Quality Center Project Customization

User Properties Project Users	Workflow
Groups Module Access Project Entities Requirement Types Risk-Based Quality Man Project Lists Automail	Script Generator - Add Defect Field Customization           Enables you to customize the fields displayed for each user group in the Add Defects dialog box.           You can also specify field order and whether a field is required.           Script Generator - Defect Details Field Customization           Enables you to customize the fields displayed for each user group in the Defect Details Generator - Defect Details Field Customization           Enables you to customize the fields displayed for each user group in the Defect Details dialog box. You can also specify field order and whether a field is required.
<u>Alert Rules</u> Workflow	Script Editor Enables you to write VBScript code for all Quality Center modules. You can also use the Script Editor to modify the scripts generated by the above tools.

2. Select Defects module script.



Add the following code to the GetNewBugPageName event procedure (which is triggered before QC opens the Add Defect dialog box).

```
select case PageNum
case "2"
GetNewBugPageName = "SM Integration (New)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For a new bug, the tab name is SM Integration (New).

4. Add the following code to the **GetDetailsPagename** event procedure (which is triggered before QC displays the Defect Details dialog box).

```
select case PageNum
case "2"
GetDetailsPageName = "SM Integration (Details)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For an existing defect, the tab name is SM Integration (Details).

### Add Fields to Tabs

To add fields to tabs:

1. In "QC - Project Customization", click **Workflow** > **Script Editor**.

#### 2. Select Defects module script.

Quality Center - Project Customization

User Properties Project Users	Workflow
Groups Module Access Project Entities Requirement Types Risk-Based Quality Ma Project Lists Automail	Script Generator - Add Defect Field Customization Enables you to customize the fields displayed for each user group in the Add Defects dialog bo You can also specify field order and whether a field is required. Script Generator - Defect Details Field Customization Enables you to customize the fields displayed for each user group in the Defect Details dialog box. You can also specify field order and whether a field is required.
Alert Rules Workflow Script Editor	Script Editor Enables you to write VBScript code for all Quality Center modules. You can also use the Script Editor to modify the scripts generated by the above tools.
Script Editor Toolba	atton Editor GetNewBugPageName
Workflow Scrip     Ommon sc     Requiremen     Fail Test Plan m     Fail Test Lab m	nodule script Bug_FieldCanChange

- 3. If **WizardFieldCust\_Details** and **WizardFieldCust\_Add** are not found in the list, do the following to generate these two methods.
  - a. Script Generator Add Defect Field Customization
  - b. Script Generator Defect Details FieldCustomization



4. Add the following code to the **WizardFieldCust\_Details** event procedure.

SetFieldApp "BG\_USER\_XX", True, False, 1, 0
SetFieldApp "BG\_USER\_XY", True, False, 1, 1
The parameters are:

Installation and Administration Guide Chapter 9: QC/ALM Defect -> SM Problem

- Field name (BG\_USER\_XX, where XX consists of two digits)
- Visible (True)
- Required (False)
- Page number (start from Ø)
- View order (start from Ø)
- 5. Add the following code to the WizardFieldCust\_Add event procedure.

```
SetFieldApp "BG_USER_XX", True, False, 1, 0
SetFieldApp "BG_USER_XY", True, False, 1, 1
```

 Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

```
if (Bug_Fields("BG_USER_XX").Value="Y") then
```

```
Bug_Fields("BG_USER_XX").IsReadOnly=True
```

end if

```
Bug_Fields.Field("BG_USER_XY").IsReadOnly=True
```

The if loop above marks the field "Synchronize with SM Problem" as read-only after selected and saved.

7. Save your changes.

### Create a View

To create a view:

- 1. Log on to QC with the integration account SMQCIntUser.
- 2. In the Defects module, click **View / Filter/Sort / Set Filters/Sort**. The purpose of this view is to make the QC Synchronizer correctly filter those defects to be synchronized to SM as Problems.
- 3. Set Synchronize with SM Problem to Y.
- 4. Add a view to Favorites:

- Name: SMIntegrationView
- Location: Private

Add Favori	te	×
Add Favori	te	
Name:	SMIntegration∀iew	
Location:	Private     O     Puk	lic

In QC Synchronizer this view will be selected as the QC data filter. Without this filter, QC defects cannot be forwarded to SM as Problems.

### Verify

Refer to the following screenshot to verify whether the Defects module on Quality Center 10 or earlier is customized successfully:

Details	SM Integration (Details)			
Syncl	hronize with SM Problem:		Problem ID:	PM0017
	Created from:	Created from SM/SC		

## On ALM 11

On ALM 11, you only need to add new fields directly to the Details tab of the Defect form.

To customize the ALM Defects module, perform the following task:

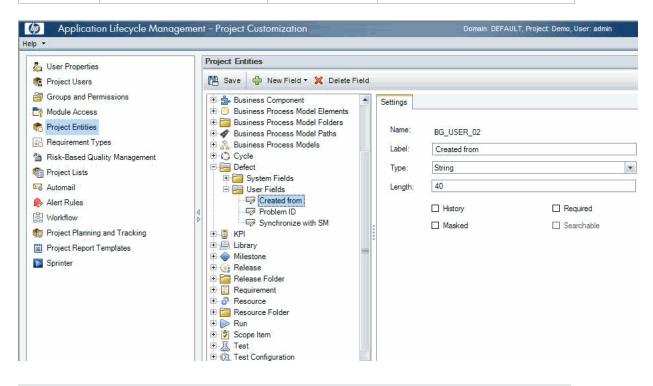
- 1. "Add Fields" below
- 2. "Create a View" on page 146
- 3. "Verify" above

### Add Fields

To add the required fields for Defect customization:

- 1. Log on to ALM as a project administrator.
- Click Tools / Customize. The "Application Lifecycle Management Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (*XX* and *XY* are sequential numbers auto-generated by ALM).

Field Name	Field Label	Field Type	Remarks
BG_USER_ <i>XX</i>	Synchronize with SM Problem	Lookup List/YesNo	Select the "Verify Value" check box
BG_USER_ <i>XY</i>	Problem ID	String	



**Note:** The data type requirements for QC/ALM fields are described in "Matching Types" on page 31.

- 4. Click Workflow > Script Editor.
- 5. Choose **Defects module script**.

 Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

```
if (Bug_Fields("BG_USER_XX").Value="Y") then
```

```
Bug_Fields("BG_USER_XX").IsReadOnly=True
```

end if

Bug\_Fields.Field("BG\_USER\_XY").IsReadOnly=True

The if loop above marks the field "Synchronize with SM Problem" as read-only after selected and saved.

7. Save your changes.

### Create a View

To create a view:

- 1. Log on to ALM with the integration account SMQCIntUser.
- 2. In the Defects module, click **View / Filter/Sort / Set Filters/Sort**. The purpose of this view is to make ALM Synchronizer correctly filter those defects to be synchronized to SM as problems.
- 3. Set Synchronize with SM Problem to Y.
- 4. Add a view to Favorites:
  - Name: SMIntegrationView
  - Location: Private

Add Favorit	e	×
Add Favorit	e	
Name:	SMIntegration∀iew	
Location:	Private     O     Public	

In ALM Synchronizer this view will be selected as the ALM data filter. Without this filter, ALM defects cannot be forwarded to SM as Problems.

### Verify

Open a new defect in ALM, select **Y** in the **Synchronize with SM** field, and click **Save**. If the Defects module on ALM 11 is customized successfully, the Defect form is displayed as follow:

Details					
Actual Fix Time:			Synchronize with SM:	Y	
Closing Date:		•	Closed in Version:		•
* Detected By:	SMQCIntUser 🖂	•	Detected in Cycle:		•
Detected in Release:		•	* Detected on Date:	2010-11-12	•
Detected in Version:		•	Estimated Fix Time:		
Planned Closing Ver		•	Priority:		•
Project:		•	Reproducible:	Y	•
Assigned To:		•	* Severity:	2-Medium	•
Status:	New	•	Subject:		•
Target Cycle:		•	Target Release:		•
Change ID:			Created from:		
Problem ID:			Modified:		

# Configuring Links in QC/ALM Synchronizer

To configure and test a link in the QC/ALM synchronizer, perform the following tasks:

- "Specify Endpoints / Type of Link" below
- "Define Filters" on the next page
- "Define Field Mappings" on the next page
- "Define Events" on page 154
- "Test the Link" on page 154

# Specify Endpoints / Type of Link

Specify the connection properties as described in "Create a Link" with the following settings specific for this type of link:

- 1. Step 1: "Endpoint 2 type" = **SM ProblemManagement**.
- 2. Step 2 "Configuration File Name" = configuration\_file\_default.xml
- 3. Step 3: "Service URL" = http://service\_manager\_ host>:<port>/sc62server/PWS/QCIntProblemService.wsdl
- 4. Step 4: "Select entity types" = Problem by Defect (this is the only available selection).

# **Define Filters**

On the Filters tab, select filter **SMIntegrationView** for the QC endpoint. If the filter is not available, see "Create a View" on page 143.

	SM ProblemManagement
C No Filter	• No Filter
<ul> <li>Use filter (for creation events):</li> </ul>	C Use filter (for creation events):
Private: SMIntegrationView	

# Define Field Mappings

Basic field mappings are summarized below:

QC	Direction	SM	Constant Value	Remarks
Problem ID	<-	ProblemID		
Defect ID	->	QCEntityID		Synchronize back on create: Yes
		QCIntegrationType	1	
		CreatedFrom	Created from QC	
QCProject	<-		(your setup)	This constant value should be the same as that for the "QC Project" parameter in the Connectivity tab.

QC	Direction	SM	Constant Value	Remarks
	->	CurrentPhase	XXX	Replace XXX with a valid phase name, such as "Problem Investigation and Diagnosis". This field mapping is optional for Service Manager 7.10.
	->	WorkFlowType	үүү	Replace YYY with a valid category name, such as <i>ITIL</i> for demo data of SM 7.0x/SC 6.2; <i>BPPM</i> is for demo data of Service Manager 7.10. This field mapping is optional for Service Manager 7.10.

Sample field mappings between Service Manager 7.10 and QC 10 are shown in the following screenshot:

Mappe	apped Fields								
Туре	QC Field	Direction	SM ProblemManagement Field						
<u> </u>	Summary	<>	Description						
Ē	Defect ID	>	QCEntityID						
<u> </u>	Severity	<>	Severity						
- P	Problem ID	<	ProblemID						
8	Value: Created from Quality C	>	CreatedFrom						
8	Value: 1	>	QCIntegrationType						
8	Value: AUTO	>	AssignmentGroup						
<u>()</u>	Value: BOB.HELPDESK	>	ProblemOwner						
8	Value: client system	>	Category						
<	Value: software	>	SubCategory						
(2)	Value: email client	>	ProductType						
()	Value: outlook	>	ProblemType						
<u>()</u>	Value: 4 - User	>	Impact						
8	Value: Problem Identification	>	CurrentPhase						
_ (?)	Value: localhost/DEFAULT/	>	QCProject						
<	Value: ITIL	>	WorkFlowType						

Sample field mappings between Service Manager 9.20 or later and ALM 11 are shown in the following screenshot:

#### Installation and Administration Guide Chapter 9: QC/ALM Defect -> SM Problem

Марре	Mapped Fields								
Туре	HP-ALM Field	Direction	SM-ProblemManagement Field						
Ē	ProblemID	<	ProblemID						
Ē	Summary	<>	Description						
Ē	Severity	<>	Severity						
Ē	Defect ID	>	QCEntityID						
<	CreatedFrom	<	Value: SM						
8	Detected on Date	<	Value: 11/2/2010						
<	Value: QC	>	CreatedFrom						
8	Value: 1	>	QCIntegrationType						
<	Value: 4 - User	>	Impact						
8	Value: DummyQCServer/DEF	>	QCProject						
8	Value: Application	>	AssignmentGroup						
8	Value: Software	>	AffectedItem						
8	Value: Problem Detection, Lo	>	CurrentPhase						
8	Value: problem	>	Category						
8	Value: hardware	>	SubCategory						
8	Value: hardware failure	>	ProductType						

### QC/ALM Field <-> SM Field

If your Service Manager 9.3x has no Process Designer (PD) Content Pack installed, or if your are working with SM 9.4x Classic, see the following summary for basic field mappings between QC/ALM and SM. The first two rows are required mappings.

QC Len	QC DB Name	QC Type	QC/ QCS Label	QC/ QCS Type	Dir	QCS SM Type	QCS Name/ SM WSDL Caption	SM WSDL Type	SM DB Name/ SM WSDL field	SM7 DB type	SC6 DB type	SM Len
40	BG_USER _03	String	Problem ID	String	<-	String	Problem ID	StringType	id	Char	Text	100
10	BG_BUG _ID	Number	Defect ID	Number	->	Number	QCEntityID	IntType	qcintegration.id	Num	Decimal	хх
255	BG_SUMMARY	String	Summary	String	<->	String	Description	StringType	description	Char	Text	хх
70	BG_SEVERITY	Lookup List	Severity	Single value list	<->	Single value list	Severity		severity	Char	Text	40

QC Defect -> SM Problem Mappings

If your Service Manager has Process Designer (PD) Content Pack is installed, or if your are working with SM 9.4x Codeless, status drive the business process of Help Desk in Process Designer Content Pack 9.30.3 and in SM 9.4x Codeless. You need to pay attention to the value mapping about the **Status** field between Service Manager and ALM.

### Constants -> SM Fields

In order to create a Problem in SM, you need to specify constant values in field mapping. The constant values vary with different SM versions and SM customizations.

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have installed PD Content Pack 9.30.2, see the following summary for sample constant field mappings:

**Note:** The following sample mappings also apply to SM 9.4x Classic.

QCS SM constant value	Dir	QCS Name/SM WSDL Caption	SM WSDL Type	SM7 DB Type	SC6 DB Type	SM DB Name / SM WSDL Field	SM Len
1	->	QCIntegrationType	StringType	Char	Text	qcintegration.type	60
Created from Quality Center	->	CreatedFrom	StringType	Char	Text	qcintegration.created.from	60
(your setup)	->	QCProject	StringType	Char	Text	qcintegration.project	60
	->	AssignmentGroup	StringType	Date/time	Date/time	assignment	50
AUTO	->	AssignmentGroup	StringType	Char	Text	assignment	50
client system	->	Category (1)	StringType	Char	Text	incident.category	40
BOB.HELPDESK	->	ProblemOwner	StringType	Char	Text	ticket.owner	40
exchange	->	ProblemType (1)	StringType	Char	Text	problem.type	40
email client	->	ProductType (1)	StringType	Char	Text	product.type	40

#### Constant -> SM Problem Mappings

#### Constant -> SM Problem Mappings, continued

QCS SM constant value	Dir	QCS Name/SM WSDL Caption	SM WSDL Type	SM7 DB Type	SC6 DB Type	SM DB Name / SM WSDL Field	SM Len
4 - User	->	Impact	StringType	Char	Text	initial.impact	50
software	->	SubCategory (1)	StringType	Char	Text	subcategory	40
ITIL	->	WorkFlowType	StringType	Char	Text	category	40
Problem Identification and Classification	->	CurrentPhase	StringType	Char	Text	current.phase	40

If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, see the table above plus the following row for sample constant field mappings:

QCS SM constant value	Dir	QCS Name/SM WSDL Caption	SM WSDL Type	SM 9.30 DB type	SM DB Name / SM WSDL Field	SM Len
true	->	IsKnownError	BooleanType	char	isKnownError	1

### **Define Events**

The following table lists the event settings for the two endpoints:

Events Tab Settings	QC Action (Event)	SM Action (Event)
Creation	Create a corresponding record in the other endpoint.	Do nothing.
Update	Update its corresponding record in the other endpoint.	Update its corresponding record in the other endpoint.
Deletion	Do nothing.	Do nothing.

The following screenshot displays the settings:

General Connectivity Scheduling Filters Events Field Mapping Advanced

IC	SM ProblemManagement
Creation When a record is created in this endpoint	Creation
Create a corresponding record in the other endpoint     Do nothing	Create a corresponding record in the other endpoint     Do nothing
Update	Update
When a record is updated in this endpoint	When a record is updated in this endpoint
Update its corresponding record in the other endpoint	Update its corresponding record in the other endpoint
C Do nothing	C Do nothing
Deletion (Full Synchronization Only)	Deletion (Full Synchronization Only)
When a record is deleted from this endpoint	When a record is deleted from this endpoint
🖲 Do nothing	© Do nothing
C Delete its corresponding record in the other endpoint	C Delete its corresponding record in the other endpoint
C Recreate based on its corresponding record in the other endpoint	C Recreate based on its corresponding record in the other endpoint

### Test the Link

To test the link:

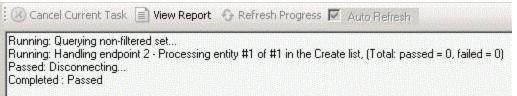
Note:

- 1. Save the configuration (an integrity check is automatically run).
- 2. Click Enable Link.
- 3. Create a defect and set Synchronize with SM Problem to Y.

New Defect	l ×
🗙 Clear 🛛 Attach: 🥒 🥜 📸 🚺 🚔 🔊 🕶 💱 💷 📓	0
* Summary: defect to problem	
Details SM Integration (New)	
Sychornize with SM Problem: Y Problem ID:	
Submit Close	

**Note:** In ALM 11, the **Synchronize with SM Problem** field and the **Problem ID** field reside on the Details tab of the Defect form.

4. Synchronize.



Installation and Administration Guide Chapter 9: QC/ALM Defect -> SM Problem

5. View the Problem in SM.

Problem Cont	rol - Proble	m Identificatio	n and Classificat	ion				9 7 1 4
Record Number:		PM0020		Stat Exp	us: ected Resolutic	n Date:	Open	<b>•</b>
Brief Description:		defect to p	oroblem				L	
Classification	Activities	Attachments	Related Records	🔶 History	🔷 Workflow	🔷 QC Integrat	ion	
Defe	ct ID:		24					
Serve	er/Domain/Proj	ect	local	nost/DEFAUL	ſ/Demo	~		
Crea	ted from:		Crea	ted from Qua	lity Center			

6. View the Defect in QC.

📴 Defect Details	
	0
Defect: 24 defect to problem	
Details SM Integration (Details)	
Operation         Own integration (octains)           Details         Sychornize with SM Problem:   Problem ID: PM0020	
Attachments	
A	
Linked Entities	
History	
Execution Report OK Cancel	

**Note:** In ALM 11, the **Synchronize with SM Problem** field and the **Problem ID** field reside on the Details tab of the Defect form.

# Chapter 10: SM Problem <-> QC/ALM Defect

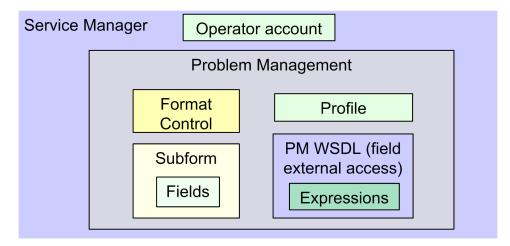
This chapter describes how to synchronize SM Problems with QC/ALM Defects ("Problem <-> Defect").

This chapter includes:

- "Customizing Service Manager/ServiceCenter for Problem Management " below
- "Customizing the QC/ALM Defects Module" on page 177
- "Configuring Links in QC/ALM Synchronizer" on page 186

# Customizing Service Manager/ServiceCenter for Problem Management

The following diagram summarizes the components which require tailoring in Service Manager/ServiceCenter.



To customize Service Manager/ServiceCenter for Problem Management, perform the following tasks:

- 1. "Add Fields" on the next page
- "Specify the External Access Definition on Service Manager" on the next page or "Specify the External Access Definition on ServiceCenter" on page 163
- 3. "Create a Subform" on page 165

- 4. "Add the Subform to a Form" on page 168
- 5. "Add Format Control Calculations/Validations" on page 169

# Add Fields

Add the following required fields to the rootcause table. Do not change them.

	Туре				
Field	Service Manager 7.0x or later	ServiceCenter			
qcintegration.type	Character	Text			
qcintegration.id	Number	Decimal			
qcintegration.project	Character	Text			
qcintegration.created.from	Character	Text			

**Note:** The data type requirements for SM fields are described in "Matching Types" on page 31.

# Specify the External Access Definition on Service Manager

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have PD Content Pack 9.30.2 installed, follow the steps below to specify the External Access Definition on Service Manager:

Note: The following steps also apply to SM 9.4x Classic.

- Create a custom External Access Definition QCIntProblemService by clicking Tailoring > WSDL configuration on Service Manager 7.0x; or clicking Tailoring > Web Services > WSDL configuration on Service Manager 7.1x or later with the following values:
  - Service Name: QCIntProblemService
  - Name: rootcause
  - Object Name: QCIntProblem

- Allowed Actions / Action Names:
  - add / Create
  - save / Update

External Access Definition						
Service Name:	QCIntPro	blemService				
Name: rootcause		)	<ul> <li>Object Name</li> </ul>		QCIntProblem	
Allowed Act	ions 🔶	Expressions	🔶 Fields			
Allowed Actions		Action Names		Action Ty	ре	
add		Create				
save	save					

**Note:** The above values are required (Do NOT change them).

2. Enable the required fields in the web service.

Field	Caption	Туре
id	ProblemID	StringType
sysmodtime	Modified	DateTimeType
qcintegration.id	QCEntityID	IntType
qcintegration.project	QCProject	StringType
qcintegration.type	QCIntegrationType	StringType
qcintegration.created.from	CreatedFrom	StringType
current.phase	CurrentPhase	StringType
category	WorkFlowType	StringType

External Access De	efinition		
Service Name: QCIntF	roblemService	_	
Name: rootcau	Ise	_	
Allowed Actions	Expressions	Fields	
Field	Caption	Туре	
qcintegration.id	QCEntityID	IntType	
id	ProblemID	StringType	
sysmodtime	Modified	DateTimeType	
qcintegration.project	QCProject	StringType	
incident.category	Category	StringType	
subcategory	SubCategory	StringType	
product.type	ProductType	StringType	
problem.type	ProblemType	StringType	
initial.impact	Impact	StringType	
severity	erity Severity		
description	Description	StringType	
assignment	AssignmentGroup	StringType	
ticket.owner	ProblemOwner	StringType	
category	WorkFlowType	StringType	

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on). The above values are required (Do NOT change them).

3. Define expressions for the web service .

#### Note:

- a. Problem Management requires an activity update provided with each save and for better flow, and this activity update will be hard-coded with the following expressions.
- b. Expressions 1 to 4 are for fixing an update issue with the Problem Management web service. For more information, see *SCR 41399*.

No	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)

No	Expression
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save

Service Name: Name:		QCIntProblemService			
				Object Name:	QCIntProble
Allowed Actions	ons 🗇 Fields				
Expressions					
cleanup(\$pm.activity);	:leanup(\$rc.upd	ate);if same(update in \$L.file, update in \$L.file.save)	) then (	(\$L.need.to.update=true)	
\$rc.update=update in	\$L.file;if (denull(	\$rc.update)={}) then (\$rc.update={"QC update ser	nt"})		
	true) then (\$rc.	update={"QC update sent"})			
if (\$L.need.to.update=					
update in \$L.file=upda	te in \$L.file.save	•			
update in \$L.file=upda		stigation and Diagnosis"			

If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, follow the steps below to specify the External Access Definition on Service Manager:

Note: The following steps also apply to SM 9.4x Codeless.

- 1. Refer to step 1.
- 2. Enable the required fields in the web service.

Field	Caption	Туре
id	ProblemID	StringType
sysmodtime	Modified	DateTimeType
qcintegration.id	QCEntityID	IntType

#### **External Access Definition**

Service Nar Name:	me:	* QCIntProblemService		Released:
Object Name	e:	QCIntProblem		
Allowed A	ctions Expressions Fields			
	Field	Caption	Турс	
	id	ProblemID	StringType	
	sysmodtime	Modified	DateTimeType	
	qcintegration.id	QCEntityID	IntType	
	qcintegration.project	QCProject StringType		
	qcintegration.type	QCIntegrationType	StringType	
	qcintegration.created.from	CreatedFrom	StringType	

**Note:** The caption value must be unique and alphanumerics (no spaces) with the first letter capitalized (AValidCaption123, AnotherValidCaption, and so on). The above values are required (Do NOT change them).

3. Define expressions for the web service.

**Note:** Problem Management requires an activity update provided with each save and for better flow, and this activity update will be hard-coded with the following expressions.

No.	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save
5	isKnownError in \$L.file=nullsub(isKnownError in \$L.file,true)

🔝 🍯 OK 💢 Cancel 📣 Previous 🗇	Next 🔂 Add 💾 Save 🐨 Delete 🔍 Find 🗊 F	ill					
External Access Definition record updated.							
External Access Definition							
Service Name:	QCIntProblemService	Released					
Name:	rootcause 🔻	Deprecated					
Object Name:	QCIntProblem						
♦ Allowed Actions ♦ Expressions	♦ Fields						
<pre>\$rc.update=update in \$L.file;i</pre>	o(\$rc.update);if same(update in \$L.file, update in \$L.file. f (denull(\$rc.update)={}) then (\$rc.update={"QC update en (\$rc.update={"QC update sent"}) .file.save						
isKnownError in \$L.file=nulls	b(isKnownError in \$L.file,true)						

# Specify the External Access Definition on ServiceCenter

To specify the External Access Definition on ServiceCenter:

 Click System Definition > Tables > rootcause > Fields and keys definitions for the rootcause table and change the settings of these fields.

No.	Field	Include in API	Field name in API	Field data type in API	Remarks
1	id	Y	ProblemID	StringType	
2	sysmodtime	Y	Modified	DateTimeType	
3	qcintegration.id	Y	QCEntityID	IntType	
4	qcintegration.project	Y	QCProject	StringType	
5	qcintegration.type	Y	QCIntegrationType	StringType	
6	qcintegration.created.from	Y	CreatedFrom	StringType	
7	current.phase	Y	CurrentPhase	StringType	
8	category	Y	WorkflowType	StringType	

Note: The caption value must be unique and alphanumerics (no spaces) with the first letter

capitalized (AValidCaption123, AnotherValidCaption, and so on).

- 2. Create a new Process record named rca.qcupdate.
  - a. Click Utilities > Tools > Document Engine > Process. Type Process Name rca.save and then click Search.
  - b. Change the name to rca.qcupdate and click Add.
  - c. Append the following lines to the Initial Expressions tab, and click **Save**.

No.	Expression
1	cleanup(\$pm.activity);cleanup(\$rc.update);if same(update in \$L.file, update in \$L.file.save) then (\$L.need.to.update=true)
2	<pre>\$rc.update=update in \$L.file;if (denull(\$rc.update)={}) then (\$rc.update={"QC update sent"})</pre>
3	if (\$L.need.to.update=true) then (\$rc.update={"QC update sent"})
4	update in \$L.file=update in \$L.file.save

Process Definition								
Process Name:	rca.qcup	ca.qcupdate						
Save Cursor Positio	)n?			Run Standard Proces	s when complete?			
Run in Window?			Wir	ndow Title:				
♦ Initial Expressions	🔷 Initial Javascript	: 🔷 RAD	Final Expressions	🗇 Final Javascript	Next Process			
if same(nullsub(full.)	name in \$G.rc.enviro	nment, full.	name in \$G.rc.global.er	vironment), true) the	n (\$L.operator=nu	illsub(\$lo.ufname, nullsu		
\$L.stamp=str(tod())	)+" ("+\$L.operator+	•"):"						
if (\$rc.update={} or	* \$rc.update={""}) t	hen (\$rc.up	date=NULL);if (\$kne.up	date={} or \$kne.upd;	ate={""}) then (\$kr	ne.update=NULL);if (\$p		
if (update in \$L.file=	NULL) then (update	in \$L.file={	})					
if (\$G.bg and not nu	(\$G.bg and not null(\$G.bg.activity.type)) then (\$rc.update=nullsub(\$rc.update, \$G.bg.activity.text);\$rc.update=nullsub(\$rc.update, "E							
if (filename(\$L.file)^	="rootcausetask")	otcausetask") then if (status in \$L.file="Past Due" and expected.resolution.time in \$L.file>tod()) then (status in \$L						
\$L.save.status=sta	tuc in ¢l file							
· ·		ti file="Rec	pened" and \$reopen.fla	an=false) then (status	: in ¢L file="Lindate	•d")		
\$reopen.flag=false			penear and preoperium	ig-raise) anen (stata.	nin pennic— opdate	,a		
cleanup(\$pm.activit	y);cleanup(\$rc.upda	te);if same(	update in \$L.file, updat	e in \$L.file.save) ther	(\$L.need.to.upda	te=true)		
\$rc.update=update	in \$L.file;if (denull(\$	irc.update):	={}) then (\$rc.update=	{"QC update sent"})				
if (\$L.need.to.upda	te=true) then (\$rc.u	ipdate={"Q	C update sent"})					
update in \$L.file=up	date in \$L.file.save							

- 3. Update the rca.view state record.
  - a. Click Utilities > Tools > Document Engine > States. Type the state name rca.view and click Search.
  - b. Add the following row to the Non-base methods table, and then click Save.

<b>Display Action</b>	Process Name	Condition	Save First
qcupdate	rca.qcupdate	\$L.mode~="close" and \$L.mode~#"add"	false

- 4. Update the Problem Management integration web service.
  - a. Click Toolkit > WSDL Configuration. In the Name field type rootcause, and then click Search.
  - b. Update the External Access Definition as follows based on table rootcause.

No.	Field	Value
1	Service Name	QCIntProblemService
2	Object Name	QCIntProblem
3	Allowed Actions/ Action Names	add/Create
		qcupdate/Update

#### **External Access Definition**

Name:			ß	Q	Object Name:	QCIntProblem
wed Actions	Expressions	🔷 Data Policy				
Allowed Act	ions			Act	ion Names	
add			Cre	ate		
qcupdate			Up	date		
	Allowed Act	wed Actions Allowed Actions add	Image: Contractions     Image: Contractions       wed Actions     Image: Contractions       Allowed Actions     Image: Contractions       add     Image: Contractions	Interview       Interview	Image: Second Tobiological Actions       Image: Second Tobiological Actions       Image: Second Tobiological Actions       Allowed Actions       Add	Interview     Interview       Interview

Note: Delete all Allowed Actions without an Action Name.

# Create a Subform

To create a subform:

- 1. Create a global list.
  - a. Click Tailoring > Tailoring Tools on Service Manager7.0x or later; or click Utilities > Tools > Global Lists on ServiceCenter to creaa a global list with the following parameters:

No.	Parameter	Value	Remarks
1	List Name	SMQC Integration PM Project List	
2	Regen Every	1 00:00:00	
3	Build List on Startup?	Yes	Check box
4	List Variable	\$G.qcintegration.problem.project	
5	User Defined List?	Yes	Check box
6	Value List	{"server1/domain1/project1", "server2/domain2/project2"}	Change to the values for your system
			<b>Note:</b> Do not include any spaces between slashes.

- b. Save this global list and click **Rebuild Global List** from the Options menu.
- 2. Click Tailoring > Forms Designer on Service Manager 7.0x or later; or click Toolkit > Forms Designeron ServiceCenter to create the pm.qcint.subform subform with the following components:

Note: Click No when the system message "Do you want to use Form Wizard?" appears.

No.	Component	Properties
1	Label	Caption: Synchronize with QC

No.	Component	Properties
2	Combo Box	Input: qcintegration.type
		Value List: 0;1;
		Display List: 0 - Not Synchronize;1 - Synchronize with QC Defect
		Select Only: Yes
		Read-Only Condition: [\$qcint.type.readonly]
3	Label	Caption: Defect ID:
4	Text	Input: qcintegration.id
		Read-Only: Yes
5	Label	Caption: Server/Domain/Project:
6	Combo Box	Input: qcintegration.project
		Value List: \$G.qcintegration.problem.project
		Read-Only Condition: [\$qcint.project.readonly]
		Mandatory Condition: [qcintegration.type]>0
7	Label	Caption: Created from:
8	Text	Input: qcintegration.project
		Read-Only: Yes

÷	
	✓ OK 🗱 Cancel
	🖽 📐 🗂 🗂 🗂 💭 🍋 🍓 🏠 🛄 🚥 🚟 🚥 🖼 🗖
	Synchronize with QC:
	Defect ID:
	perect ID:
	Server/Domain/Project
	Created from:

### Add the Subform to a Form

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have PD Content Pack 9.30.2 installed, follow the steps below to add the subform you created to a form:

Note: The following steps also apply to SM 9.4x Classic.

- From Forms Designer, open the default form of a Problem Management phase (PM.pc.ident.and.class is used as an example in ServiceCenter 6.2/Service Manager 7.0x).
- 2. Add a notebook Tab with the QC Integration caption.
- Add a subform to the new tab with format pm.qcint.subform, which is displayed in the following screenshot:

😽 *Forms Designer: 🗙 🕌 Problem Managemen 🛛 🛱 rootcausephase: 🎽 🗂	The Properties 없	- 8
	Subfo	ormat
	Property	Value
III III 🖪 🚥 🗶 🛲 🧮 🖉 🌾 🔚 🞇 👗 🚺	Display Blank	
	Display Using Table	
Problem Control - Problem Investigation and Diagnosis	Format	pm.qcint.subform
Record Number:	Height	40
	Input	
Expected Resolution E	Name	
Brief Description:	Virtual Join	
	Visible	
♦ SLA ♦ Related Records ♦ Tasks ♦ History ♦ Workflow ♦ QC Integration »s	Visible Condition	
	Width	143
	x	3
	Y	1

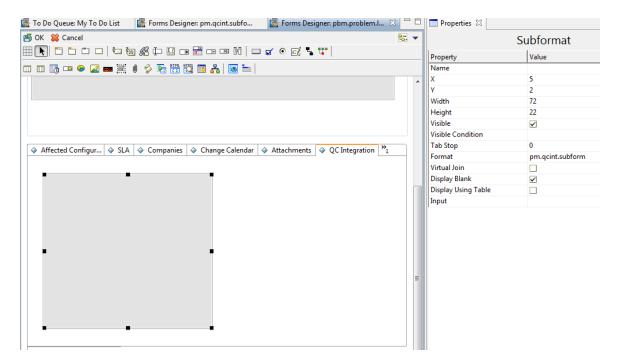
4. Save the changes.

**Note:** If the error message "Format 'pm.qcint.subform' not found (display, show.rio)" appears, restart the SM server to enable the subform.

If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, follow the steps below to add the subform you created to a form:

Note: The following steps also apply to SM 9.4x Codeless.

- From Forms Designer, open the default form of a Problem Management phase (pbm.problem.logging is used as an example on Service Manager 9.3x).
- 2. Add a notebook Tab with the QC Integration caption.
- 3. Add a subform to the new tab with format pbm.problem.logging, which is displayed in the following screenshot:



4. Save the changes.

### Add Format Control Calculations/Validations

**Note:** Steps in this topic are applicable if your Service Manager has no Process Designer (PD) Content Pack installed, or if you are working with SM 9.4x Classic. Otherwise, skip this topic and refer to "Add Rule Set Calculations/Validations" on page 171 for SM 9.3x configurations in the PD environment and SM 9.4x Codeless configurations.

To add format control calculations and validations:

 Open the format control record of the previous Problem Management form PM.pc.ident.and.class. Installation and Administration Guide Chapter 10: SM Problem <-> QC/ALM Defect

#### 2. Click Calculations.

#### 3. Add two rows with the following values:

Row	Parameter	Value
1	display	true
	initial	true
	calculation	<pre>\$qcint.type.readonly=2;if (qcintegration.type in \$file~=0) then (\$qcint.type.readonly=1)</pre>
2	display	true
	initial	true
	calculation	<pre>\$qcint.project.readonly=2;if (qcintegration.type in \$file~=0 and not null (qcintegration.project in \$file)) then (\$qcint.project.readonly=1)</pre>

#### 4. Click Validations.

5. Add one validation with the following values:

No.	Parameter	Value
1	Validation	not null(qcintegration.project in \$file)
2	Message	The Server/Domain/Project is required.
3	Add	qcintegration.type in \$file~=0
4	Update	qcintegration.type in \$file~=0
5	Set Focus to	qcintegration.project

- 6. Save your changes.
- 7. Verify.

Classification	Activities	Attachments	Related Records	🔷 History	🔷 Workflow	🔷 QC Integration	899466
Syr	nchronize with (	ýc:		1 - Synch	ronize with QC I	Defect	-
Defect ID:				7			
Ser	ver/Domain/Pro	oject		localhost/	QADEMO/Demo	)	-
Cre	ated from:			Created f	rom Quality Cer	nter	

# Add Rule Set Calculations/Validations

**Note:** Steps in this topic are applicable for the Service Manager 9.3x with Process Designer (PD) Content Pack 9.30.3 and SM 9.4x Codeless only. Otherwise, refer to "Add Format Control Calculations/Validations" on page 169 for configurations in SM 9.3x non-PD, SM 9.3x with PD Content Pack 9.30.2, and SM 9.4x Classic environment.

To add rule set calculations and validations, follow the steps below:

- 1. "Copy an Existing Workflow" below
- 2. "Associate an Existing Problem Category with the New Workflow" on the next page
- 3. "Create New Rule Set for Initialization and Validation" on page 173
- 4. "Associate the New Workflow with the New Rule Set" on page 175

### Copy an Existing Workflow

You can use copies of the existing workflows in another business process, or make changes to the HP proprietary workflow copies.

To copy an existing workflow, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Copy Existing Workflow**.
- 2. On the Clone a Workflow page, select the workflow you want to copy. For example, Problem.

3. Type SMQCIntPbM in the New workflow name field.

Clone	Clone a Workflow					
		Please specify the new workflow name, as well as the pr	efix for new rule sets if they are to be copied as well.			
		New workflow name:	* SMQCIntPbM			
		Copy rule sets?				
		Rule set prefix:	* SMQCIntPbM			

- 4. Select the Copy rule sets check box if you want to copy rule sets, and then type a rule set prefix.
- 5. Click **OK**.

The newly copied workflow appears in the list on the Clone a Workflow page.

### Associate an Existing Problem Category with the New Workflow

You can update existing problem categories, subcategories and areas and associate the updated categories with the new workflow so that they can be used in another business process. For more information, refer to *HP Service Manager – Process Designer Content Pack Administrator's Guide*.

To associate an existing change category with the new workflow, follow the steps below:

- 1. From the System Navigator, click **Problem Management > Configuration > Problem Categories**.
- 2. Click Search.
- 3. Select the problem category for which you want to add a workflow. For example, problem.
- 4. In the Problem Category page, remove the currently assigned workflow from the Workflow field.
- 5. Type SMQCIntPbM in the workflow field.

6. Click Save to associate the problem category with the workflow.

Problem Category				
Name:	problem	Apply To:	Problem	
Active:				
Description:	incident			
Workflow:	* SMQCintPbM			
Subcategories Workflow				4
	Abandonment	Review	Closure	

### Create New Rule Set for Initialization and Validation

To create a new rule set for initialization, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Rule Sets**.
- 2. Type the values as follow:

Field	Value
ID	pbm.alm.int.init
Available as action	False
Name	Initialize for ALM integration in the Problem Record
Table name	rootcause
HP Proprietary	

- 3. Click New and Save.
- 4. Click Add Rule.
- 5. In the Select Rule Type page, click Run JavaScript.

Installation and Administration Guide Chapter 10: SM Problem <-> QC/ALM Defect

6. In the Run JavaScript page, type the values as follow:

Field	Value
Rule Description	Run Javascript for initializing Integration type and project in the Problem Record
Statement	<pre>vars['\$qcint.type.readonly'] = 2; vars['\$qcint.project.readonly'] = 2; var _null=system.functionsnull; var file = vars.\$L_file; if(file["qcintegration.type"] != 0 &amp;&amp; !_null(file["qcintegration.type"])) { vars['\$qcint.type.readonly'] = 1 } if(file["qcintegration.type"] != 0 &amp;&amp; !_null(file["qcintegration.project"])) { vars['\$qcint.project.readonly'] = 1 }</pre>
	vars['\$qcint.project.readonly'] = 1 }

#### 7. Click **OK**.

8. Click **Save** and **Exit**.

To create a new rule set for validation, follow the steps below:

- 1. From the System Navigator, click **Tailoring > Process Designer > Rule Sets**.
- 2. Type the values as follow:

Field	Value	
ID	pbm.alm.int.validation	
Available as action	False	
Name	Validation for ALM integration in the Problem Record	

Field	Value
Table name	rootcause
HP Proprietary	

- 3. Click New and Save.
- 4. Click Add Rule.
- 5. In the Select Rule Type page, click **Set Mandatory Fields**.
- 6. Refer to step 6 to step 13 as described in "Create New Rule Set for Initialization and Validation" on page 61.

### Associate the New Workflow with the New Rule Set

To associate the new workflow with the new initialization rule set, follow the steps below:

- 1. From the System Navigator, click **Problem Management > Configuration > Workflows**.
- 2. Select Problem in the workflows list.
- 3. Select the first phase in the workflow graph.
- 4. Click **Rule Sets** tab > **Initialization** tab.
- 5. Click **Add** and select the pbm.alm.int.init rule set you just created.

MP Service Manager				User: falcon
	To Do Queue: My To Do List Workflows	Workflow: Problem (8) Rule Sets		
	K 💾 Save   🔍 Zoom in 🔍 Zoom out   🗌 Ad	d phase 💮 Delete 🛛 🚰 Workflow properties		8 🖬 1
Open New Problem Problem Queue	·			
Problem Task Queue				
Search Knowledgebase Search Problems		¢ , ¢	¢ ,	
Search Problem Tasks				
Request Management	Logging Categorizati	on 🕂 🗘 Investigation 🖓 F	lesolution 🗘 Review Closure	
Service Catalog		- $  -$		
Service Desk				
Service Level Management	Abandon	Add Rule Sets - Initialization		
System Administration			Name	
Tailoring		apply.template	Apply Template	
> Audit		create.template	Create Template from Record	
Differential Upgrade		pbm.abandon.wizard	Run Ahandon Wizard	•
Document Engine	Phase - Logging			
Event Services	Details Forms Rule Sets Actions		PBM abandonment	
Knowledge Engineering	Detais Tornis Rule Sets Actoris	pbm.abandonment.field.validation	PBM field validation at abandonment phase	
<ul> <li>Notifications</li> <li>Process Designer</li> </ul>	On enter On exit Initialization	c 🗹 pbm.alm.int.init	Initialize for ALM integration in the Problem Record	
<ul> <li>Process Designer</li> <li>Configuration</li> </ul>	Add 💮 Delete 😔 View 🏠 Up 🚸 Do	pbm.bac.pi.assciate	Associate with BAC PI	
Configuration Copy Existing Workflow		pbm.calculate.rc.calendar	Calculate whether RC calendar needs to be displayed	
Export Workflow	Rule Sets	pbm.categorization.init.status	Initialize pm status at categorization phase	
Rule Sets	pbm.status.list	pbm.categorization.mandatory	PBM mandatory validation at categorization phase	
Workflows	pbm.set.impact.urgency.value			
SQL Utilities	pbm.logging.init.status		OK Cance	4
Tailoring Tools	Dbm.initialize.id	L		
Web Services				
Codes				
Database Dictionary				

- 6. Click **OK**.
- 7. Click Rule Sets tab > On display tab.
- 8. Repeat step 5 and 6.
- 9. Click **Rule Sets** tab > **On enter** tab.
- 10. Click **Add** and select the pbm.alm.int.validation rule set you just created.

MP Service Manage	er				Use
		To Do Queue: My To Do List Workflows	Workflow: Problem 🗵		
12 1 🔂 😂	*	💾 Save   🏵 Zoom in 🔍 Zoom out   🔲 Ad	d phase 💮 Delete   🚰 Workflow properties		E 🗆 I
Worklows Open New Problem Problem Queue Problem Task Queue Search Problem S Search Problems Search Problem Tasks Request Management		Logging D - Calegorizati	on -¢→↓ [restigation - ¢→↓ ]		
Service Catalog Service Desk					
Service Level Management	_	Abandoos			
-		Abandonr	Add Rule Sets - On enter		
ystem Administration			🗖 Id	Name	
alloring			apply.template	Apply Template	*
Audit		•	Create.template	Create Template from Record	
Differential Upgrade			pbm.abandon.wizard	Run Abandon Wizard	
Document Engine Event Services		Phase - Logging	pbm.abandonment	PBM abandonment	
Knowledge Engineering		Details Forms Rule Sets Actions	pbm.abandonment.field.validation	PBM field validation at abandonment phase	
Notifications				Initialize for ALM integration in the Problem Record	
Process Designer	=	On enter On exit Initialization O	n phm alm int validation	Validation for ALM integration in the Problem Record	
Configuration		🖨 Add 🐨 Delete 👶 View 🏫 Up 🚸 Do	pbm.bac.pi.assciate	Associate with BAC PI	-
Copy Existing Workflow		Rule Sets	pbm.calculate.rc.calendar	Calculate whether RC calendar needs to be displayed	
Export Workflow		pbm.logging.mandatory	pbm.categorization.init.status	Initialize pm status at categorization phase	
Rule Sets Workflows		pbm.set.openedby	ponicategorization.inicistatios	Initialize prinstatus at categorization priase	-
SQL Utilities		pbm.set.incidentcategory		OK Can	cel
Tailoring Tools		pbm.set.affected.cis	L		
Web Services		pbm.init.sla			
Codes		E pontant au			
Database Dictionary					
Database Manager					

- 11. Click **OK**.
- 12. Click Save.

# Customizing the QC/ALM Defects Module

The steps for customizing the Defects module are different for different versions of Quality Center/ALM.

# On QC 10 and Earlier

To customize the Defects module on Quality Center 10 or earlier, perform the following tasks:

- 1. "Add Fields" below
- 2. "Add Tabs" on the next page
- 3. "Add Fields to Tabs" on page 180
- 4. "Create a View" on page 182
- 5. "Verify" on page 182

### Add Fields

To add the required fields for Defect module customization:

- 1. Log on to QC as a project administrator.
- 2. Click **Tools / Customize**. The "QC Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (*XX*, *XY* and *XZ* are sequential numbers auto-generated by QC).

Field Name	Field Label	Field Type	Remarks
BG_USER_ <i>XX</i>	Synchronize with SM Problem	Lookup List/YesNo	Select "Verify Value" check box
BG_USER_ <i>XY</i>	Problem ID	String	
BG_USER_ <i>XZ</i>	Created from	String	

Quality Center -	Project Customization			
<u>User Properties</u> Project Users <u>Groups</u> Madula Resease	Project Entities	Field Settings		
Module Access Project Entities Requirement Types Risk-Based Quality Project Lists Automail Alert Rules Workflow	Defect     Defect     System Fields     User Fields     Change id     Forward as problem     REC Problem ID	Field Name: Field Label: Field Type: Field Length:	BG_USER_01 Forward as problem Lookup List 40 Required	·
		Masked	Searchable	

**Note:** The data type requirements for QC fields is described in "Matching Types" on page 31.

### Add Tabs

To add tabs to the Defect form and show fields on these tabs:

- 1. In "QC Project Customization", click **Workflow** > **Script Editor**.
- 2. Select **Defects module script**.

User Properties Project Users Workfl	ow
Module Access         Enables           Project Entities         You can           Requirement Types         Script Ge           Risk-Based Quality Man         Enables	enerator - Add Defect Field Customization you to customize the field's displayed for each user group in the Add Defects dialog bo also specify field order and whether a field is required. Interator - Defect Details Field Customization you to customize the fields displayed for each user group in the Defect Details dialog can also specify field order and whether a field is required.
Alert Rules Script Ed Workflow Enables	litor you to write VBScript code for all Quality Center modules. also use the Script Editor to modify the scripts generated by the above tools.
Script Editor Script Editor Toolbar Button Editor	GetNewBugPageName
Workflow Scripts     Common script     Requirements module script     Test Plan module script     Test Lab module script     Manual Runner script	Bug_New     Bug_MoveTo     Bug_FieldCanChange     Bug_FieldChange     Bug_CanPost     Bug_CanPost     Bug_CanPolete

Add the following code to the GetNewBugPageName event procedure (which is triggered before QC opens the Add Defect dialog box).

```
select case PageNum
case "2"
GetNewBugPageName = "SM Integration (New)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For a new bug, the tab name is SM Integration (New).

4. Add the following code to the **GetDetailsPagename** event procedure (which is triggered before QC displays the Defect Details dialog box).

```
select case PageNum
case "2"
GetDetailsPageName = "SM Integration (Details)"
end select
```

**Note:** The parameter 2 specifies tab 2 (the second tab). For an existing defect, the tab name is SM Integration (Details).

### Add Fields to Tabs

To add fields to tabs:

- 1. In "QC Project Customization", click **Workflow** > **Script Editor**.
- 2. Select **Defects module script**.

User Propertie Project Users	<u>s</u>	Workflo	w
Groups Module Access Project Entities Requirement T Risk-Based Gu Project Lists Automail	VDes	Enables y You can a <u>Script Gen</u> Enables y	nerator - Add Defect Field Customization ou to customize the fields displayed for each user group in the Add Defects dialog bo lso specify field order and whether a field is required. lerator - Defect Details Field Customization ou to customize the fields displayed for each user group in the Defect Details dialog an also specify field order and whether a field is required.
Alert Rules Workflow			lor ou to write VBScript code for all Quality Center modules. Iso use the Script Editor to modify the scripts generated by the above tools.
Script Edito	r		
Script Editor	Toolber Butt	on Editor	- S GetNewBugPageName - S GetDetailsPageName - S Bug New

3. If **WizardFieldCust\_Details** and **WizardFieldCust\_Add** are not found in the list, do the following to generate these two methods.

a. Script Generator - Add Defect Field Customization

#### b. Script Generator - Defect Details Field Customization



4. Add the following code to the **WizardFieldCust\_Details** event procedure.

SetFieldApp "BG\_USER\_XX", True, False, 1, 0 SetFieldApp "BG\_USER\_XY", True, False, 1, 1 SetFieldApp "BG\_USER\_XZ", True, False, 1, 2

The parameters are:

- Field name (BG\_USER\_XX, where XX consists of two digits)
- Visible (True)
- Required (False)
- Page number (start from Ø)
- View order (start from 0)
- 5. Add the following code to the WizardFieldCust\_Add event procedure.

```
SetFieldApp "BG_USER_XX", True, False, 1, 0
SetFieldApp "BG_USER_XY", True, False, 1, 1
SetFieldApp "BG_USER_XZ", True, False, 1, 2
```

6. Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:

```
if (Bug_Fields("BG_USER_XX").Value="Y") then
```

```
Bug_Fields("BG_USER_XX").IsReadOnly=True
```

end if

```
Bug_Fields.Field("BG_USER_XY").IsReadOnly=True
```

The if loop above marks the field "Synchronize with SM Problem" as read-only after selected and saved.

7. Save your changes.

## Create a View

To create a view:

- 1. Log on to ALM with the integration account SMQCIntUser.
- 2. In the Defects module, click **View / Filter/Sort / Set Filters/Sort**. The purpose of this view is to make the QC Synchronizer correctly filter those defects to be synchronized to SM as problems.
- 3. Set Synchronize with SM Problem to Y.
- 4. Add a view to Favorites:
  - Name: SMIntegrationView
  - Location: Private

Add Favoril	te			×
Add Favorit	e			
Name:	SMIntegration∀iew			
Location:	<ul> <li>Private</li> </ul>	0	Public	

In QC Synchronizer this view will be selected as the QC data filter. Without this filter, QC defects cannot be forwarded to SM as Problems.

## Verify

Refer to the following screenshot to verify whether the Defects module on Quality Center 10 or earlier is customized successfully:

Details SM Integration (Details)		
Synchronize with SM Problem: 🝸	Problem ID: PM0017	
Created from: Create	om SM/SC	

# On ALM 11

On ALM 11, you only need to add new fields directly to the Details tab of the Defect form.

To customize the ALM Defects module, perform the following task:

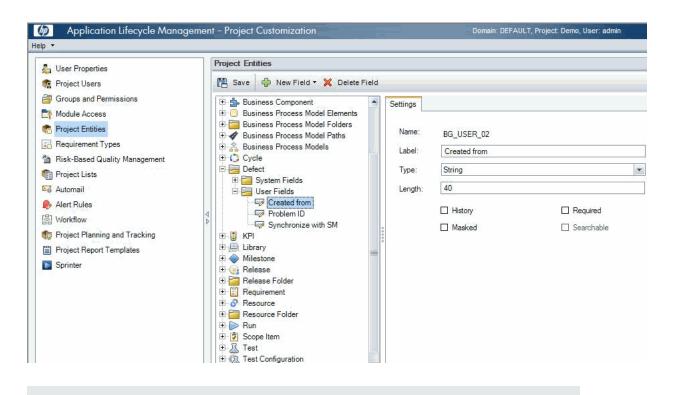
- 1. "Add Fields" below
- 2. "Create a View" on page 185
- 3. "Verify" on page 185

## Add Fields

To add the required fields for Defect customization:

- 1. Log on to ALM as a project administrator.
- Click Tools / Customize. The "Application Lifecycle Management Project Customization" module opens.
- 3. Add the following fields for the defect entity in Project Entities (*XX* and *XY* are sequential numbers auto-generated by ALM).

Field Name	Field Label	Field Type	Remarks
BG_USER_ <i>XX</i>	Synchronize with SM Problem	Lookup List/YesNo	Select the "Verify Value" check box
BG_USER_ <i>XY</i>	Problem ID	String	



**Note:** The data type requirements for QC/ALM fields are described in "Matching Types" on page 31.

- 4. Click Workflow > Script Editor.
- 5. Select Defects module script.
- Set the **Readonly** fields by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines:
  - if (Bug\_Fields("BG\_USER\_XX").Value="Y") then

```
Bug_Fields("BG_USER_XX").IsReadOnly=True
```

end if

```
Bug_Fields.Field("BG_USER_XY").IsReadOnly=True
```

The if loop above marks the field "Synchronize with SM Problem" as read-only after selected and saved.

7. Save your changes.

## Create a View

To create a view:

- 1. Log on to ALM with the integration account SMQCIntUser.
- 2. In the Defects module, click **View / Filter/Sort / Set Filters/Sort**. The purpose of this view is to make the ALM Synchronizer correctly filter those defects to be synchronized to SM as problems.
- 3. Set Synchronize with SM Problem to Y.
- 4. Add a view to Favorites:
  - Name: SMIntegrationView
  - Location: Private

Add Favoril	te	×
Add Favorit	te	
Name:	SMIntegration∀iew	
Location:	Private     O     Public	:

In ALM Synchronizer this view will be selected as the ALM data filter. Without this filter, ALM defects cannot be forwarded to SM as Problems.

## Verify

Open a new defect in ALM, select **Y** in the **Synchronize with SM** field, and click **Save**. If the Defects module on ALM 11 is customized successfully, the Defect form is displayed as follow:

Details					
Actual Fix Time:			Synchronize with SM:	Y	
Closing Date:		•	Closed in Version:		•
* Detected By:	SMQCIntUser	₩ .	Detected in Cycle:		•
Detected in Release:		•	* Detected on Date:	2010-11-12	•
Detected in Version:		•	Estimated Fix Time:		
Planned Closing Ver		•	Priority:		•
Project:		•	Reproducible:	Y	•
Assigned To:		•	* Severity:	2-Medium	•
Status:	New	•	Subject:		•
Target Cycle:		•	Target Release:		•
Change ID:			Created from:		
Problem ID:			Modified:		

# Configuring Links in QC/ALM Synchronizer

To configure and test a link in the QC/ALM synchronizer, perform the following tasks:

- "Specify Endpoints / Type of Link" below
- "Define Filters" on the next page
- "Define Field Mappings" on the next page
- "Define Events" on page 190
- "Test the Link" on page 191

# Specify Endpoints / Type of Link

Specify the connection properties as described in "Create a Link" with the following settings specific for this type of link:

- 1. Step 1: "Endpoint 2 type" = SM ProblemManagement.
- 2. Step 3: "Service URL" =

http://<*service\_manager\_host*>:<*port*>/sc62server/PWS/QCIntProblemService.wsdl

3. Step 4: "Select entity types" = Problem by Defect (this is the only available selection)

# **Define Filters**

On the Filters tab, select filter **SMIntegrationView** for the QC endpoint. If the filter is not available, see "Create a View" on page 143.

_ QC	SM ProblemManagement
C No Filter	No Filter
<ul> <li>Use filter (for creation events):</li> </ul>	C Use filter (for creation events):
Private: SMIntegrationView	

# Define Field Mappings

If the Service Manager Process Designer (PD) Content Pack is not installed, or you have installed PD Content Pack 9.30.2, see the following summary for basic field mappings:

Note: The following mappings also apply to SM 9.4x Classic.

QC	Directions	SM	Constant Value	Remarks
Problem ID	<-	ProblemID		Synchronize back on create: Yes
Defect ID	->	QCEntityID		Synchronize back on create: Yes
Synchronize with SM Problem			Y	
		QCIntegrationType	1	
Created from			Created from SM/SC	
		CreatedFrom	Created from QC	

QC	Directions	SM	Constant Value	Remarks
	->	CurrentPhase	XXX	Replace <i>XXX</i> with a valid phase name, such as "Problem Investigation and Diagnosis".
				This field mapping is optional for demo data of Service Manager 7.10.
	->	QCProject	(your setup)	This value should be same with the "QC Project" parameter in the Connectivity tab.
	->	WorkFlowType	YYY	Replace YYY with a valid category name, such as <i>ITIL</i> for demo data of SM 7.0x/SC 6.2; <i>BPPM</i> is for demo data of Service Manager 7.10. This field mapping is optional for demo
				data of Service Manager 7.10.

Sample field mappings are shown in the following screenshot:

Марре	Mapped Fields					
Туре	QC Field	Direction	SM ProblemManagement Field			
<u> </u>	Summary	<>	Description			
G	Defect ID	>	QCEntityID			
G	Severity	<>	Severity			
G	Problem ID	<	ProblemID			
<u>(</u> )	Sychornize with SM Problem	<	Value: Y			
8	Value: Created from Quality C	>	CreatedFrom			
~	Value: 1	>	QCIntegrationType			
$\otimes$	Value: AUTO	>	AssignmentGroup			
~	Value: BOB.HELPDESK	>	ProblemOwner			
$\otimes$	Value: client system	>	Category			
<₿	Value: software	>	SubCategory			
8	Value: email client	>	ProductType			
<u>(</u>	Value: outlook	>	ProblemType			
8	Value: 4 - User	>	Impact			
~	Value: Problem Identification	>	CurrentPhase			
< <>	Value: localhost/DEFAULT/	>	QCProject			
()	Value: ITIL	>	WorkFlowType			
8	Created from	<	Value: Created from SM/SC			

If the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, see the above table plus the following row for basic field mappings:

QC	Directions	SM	Constant Value	Remarks
	->	lsKnownError	true	This field mapping is added in the PD Content Pack 9.30.3. This field mapping marks the current record as KnownError.

Sample field mappings are shown in the following screenshot:

Mapped Fields					
Туре	QC Field	Direction	SM ProblemManagement Field		
Ē	Summary	<>	Description		
<u>D</u>	Defect ID	>	QCEntityID		
<b>P</b>	Severity	<>	Severity		
Ð	Problem ID	<	ProblemID		
<i>\</i>	Sychomize with SM Problem	<	Value: Y		
$\diamond$	Value: Created from Quality C	>	CreatedFrom		
$\langle \rangle$	Valua: 1	>	QCIntegrationType		
8	Value: AUTO	>	AssignmentGroup		
$\diamond$	Value: BOB.HELPDESK	>	ProblemOwner		
$\diamond$	Value: client system	>	Category		
$\diamond$	Value: toltware	>	SubCategory		
8	Value: email client	>	ProductType		
<i>\</i>	Value: outlook	>	ProblemType		
$\langle \rangle$	Value: 4 - User	>	Impact		
<i>\</i>	Value: Problem (dentification	>	CurrentPhase		
Ø	Value: localhosi/DEFAULT/	>	QCProject		
<i>\</i>	Value: ITIL	>	WorkFlowType		
8	Created from	<	Value: Created from SM/SC		
$\langle \rangle$	Value: true	>	IsKnownError		

# **Define Events**

The following table lists the event settings for the two endpoints.

Events Tab Settings	QC Action (Event)	SM Action (Event)
Creation	Create a corresponding record in the other endpoint.	Create a corresponding record in the other endpoint.
Update	Update its corresponding record in the other endpoint.	Update its corresponding record in the other endpoint.
Deletion	Do nothing.	Do nothing.

The following screenshot displays the settings:

## Installation and Administration Guide Chapter 10: SM Problem <-> QC/ALM Defect

General Connectivity Scheduling Filters Events Field Mapping Advan	ced
	SM ProblemManagement
Creation	Creation
When a record is created in this endpoint	When a record is created in this endpoint
Create a corresponding record in the other endpoint	C Create a corresponding record in the other endpoint
Do nothing	C Do nothing
Update	Update
When a record is updated in this endpoint	When a record is updated in this endpoint
Update its corresponding record in the other endpoint	Update its corresponding record in the other endpoint
Do nothing	Do nothing
Deletion (Full Synchronization Only)	Deletion (Full Synchronization Only)
When a record is deleted from this endpoint	When a record is deleted from this endpoint
Do nothing	Do nothing
Delete its corresponding record in the other endpoint	Delete its corresponding record in the other endpoint
Recreate based on its corresponding record in the other endpoint	Recreate based on its corresponding record in the other endpoint

# Test the Link

To test the link:

### Note:

- 1. A mandatory field (in either SM and QC) does not accept a null value. Synchronization may fail if a mandatory field is mapped to a field that can be null.
- 2. The following sample steps are for your reference only. The exact steps required on your system may differ significantly. The phase in which the QC Integration tab appears may be different on your system.
- 1. Save the configuration (an integrity check is automatically run).
- 2. Click Enable Link.
- 3. Create a problem in SM and select "1-Synchronize with QC Defect".

							$\sim$	$\phi$
Problem Cont	rol - Probler	n Identificatio	on and Classificat	ion				
Record Number:		PM0021		Sta		l	Updated	•
Brief Description:		defect <-:	> problem new problen		ected Resolutio	n Date:		▼
Classification	Activities	Attachments	Related Records	🔷 History	🔷 Workflow	🔷 QC Integratio	n	
Sync	hronize with QC	:	1 - 5	iynchronize wi	ith QC Defect	~		
Defe	ct ID:							
Serve	er/Domain/Proje	ct	local	host/DEFAUL	T/Demo	~		
Crea	ted from:							

4. Create a defect in QC and set Synchronize with SM Problem to **Y**.

New Defect	_ 🗆 X
🗙 Clear   Attach: 🥒 🥜 📸 🛐 🚔 🕵 🕶 💱 💷 👹	0
* Summary: defect <-> problem new defect	
Details SM Integration (New)	
Sychornize with SM Problem:	
Created from:	
Submit Close	

#### 5. Synchronize.

Cancel Current Task in View Report Refresh Progress Auto Refresh

Running: Task execution started.

Running: Connecting to endpoint 1...

Running: Querying non-filtered set...

Running: Handling endpoint 2 - Processing entity #1 of #1 in the Create list, (Total: passed = 0, failed = 0)

Running: Handling endpoint 1 - Processing entity #1 of #1 in the Create list, (Total: passed = 2, failed = 0)

Passed: Disconnecting...

Completed : Passed

#### 6. View the problem in SM.

roblem Control - Prob	olem Identification an	d Classification			
Record Number:	PM0021	Status:		Updated	<b>~</b>
		Expected Re	esolution Date:		-
rief Description:	defect <-> probl	em new problem			
Classification	es 🔷 Attachments 🗇 Ri	elated Records 🛛 🗇 History 🔷 Wor	'kflow 🛛 🧇 QC Inted	ration	
			v qe intog		
Synchronize with	OC:	1 - Synchronize with QC De	efect	~	
·				×	
Defect ID:		26			
				_	
Server/Domain/Pr	roject	localhost/DEFAULT/Demo		<b>V</b>	
				-	
Created from:					
Created from:					
Created from:					
Problem Control - Prol	blem Identification a	nd Classification			
	blem Identification an PM0022	Status:		Open	· ·
roblem Control - Prol Record Number:	<u>ک</u> M0022	Status: Expected R	esolution Date:	Open	▼ ▼
Problem Control - Prol Record Number:		Status: Expected R	esolution Date:	Open	
Problem Control - Prol Record Number: Brief Description:	אני אונט איז אונט איז אאסטער איז	Status: Expected R	esolution Date:		
Problem Control – Prol Record Number: Brief Description:	אני אונט איז אונט איז אאסטער איז	Status: Expected R			
Problem Control - Prol Record Number: Brief Description:	אני אונט איז אונט איז אאסטער איז	Status: Expected R			
Problem Control - Prol Record Number: Brief Description:	[>M0022 defect <-> prot	Status: Expected R	orkflow 🛛 🔷 QC Inte		
Problem Control - Prol Record Number: Brief Description: Classification	[>M0022 defect <-> prot	Status: Expected R blem new defect telated Records 🗇 History 🗇 Wo	orkflow 🛛 🔷 QC Inte	gration	
Problem Control - Prol Record Number: Brief Description: Classification 🛛 🕹 Activiti	[>M0022 defect <-> prot	Status: Expected R olem new defect Related Records 🔷 History 🔷 Wo	orkflow 🛛 🔷 QC Inte	gration	
Problem Control - Prol Record Number: Brief Description: Classification	>M0022         defect <-> prot         ies       ♦ Attachments       ♦ F         h QC:	Status: Expected R blem new defect telated Records 🗇 History 🗇 Wo	orkflow 🛛 🔷 QC Inte	gration	
Problem Control - Pro Record Number: Brief Description: Classification 🗇 Activiti Synchronize with Defect ID:	>M0022         defect <-> prot         ies       ♦ Attachments       ♦ F         h QC:	Status: Expected R olem new defect Related Records I History I Wo 1 - Synchronize with QC D 25	orkflow 🛛 🔷 QC Inte	gration	

7. View the defect in QC.

📑 Defect Det	ails	_ 🗆 🗙
	▶  🞗 🏲 ! 🗉 • 🗊	0
Defect:	25 defect <-> problem new defect	
2	Details     SM Integration (Details)       Sychornize with SM Problem:     Problem ID:   PM0022	
Details	Sychornize with SM Problem: Problem ID: PM0022	-
°Cij	Created from:	
🗒 Defect Deta	ails	
		0
Defect:	26 defect <-> problem new problem	
	Details SM Integration (Details)	
Details	Sychornize with SM Problem: Y Problem ID: PM0021	
17	Created from: Created from SM/SC	
Attachments		
â		
Linked Entities		
<b>\$</b>		
History		

**Note:** In ALM 11, the **Synchronize with SM Problem** field and the **Problem ID** field reside on the Details tab of the Defect form.

# Chapter 11: Upgrading the Integration

This chapter describes how to upgrade the integration. If the integration components such as Service Manager/ServiceCenter, QC/ALM, and QC/ALM Synchronizer need to be upgraded, refer to the Upgrade Guides of these products.

This chapter includes:

- "Upgrading to the Latest Release" below
- "Post-Upgrade Tasks" on page 205

# Upgrading to the Latest Release

To upgrade the integration to the latest release, perform the following tasks:

- 1. "Back up Jar Files and Links" below
- 2. "Deploy the Latest Adapters" below
- 3. "Upgrade for Various Synchronization Scenarios" on the next page

## Back up Jar Files and Links

To back up jar files and links:

- 1. Back up all sm-\*.jar files (including stub jar) in the <QCS\_Install\_Dir>\adapters\lib directory.
- Back up the configurations of all links in the QCS client by clicking Link > Export > Link Configuration Into XML File....
- 3. Back up the data of all links in the QCS client by clicking Link > Export > Link Data Into Backup File....

## Deploy the Latest Adapters

To deploy the latest adapters:

- 1. Stop the HP Synchronizer server by clicking **All Programs > HP Quality Center Synchronizer > Stop Synchronizer**.
- 2. Remove all sm-\*.jar files from the <*QCS\_Install\_Dir*>\adapters\lib directory.
- Install the SMQC Patch 2 package.
   Download the installer for Patch 2 from the HP Quality Center Add-ins website (http://updates.merc-int.com/qualitycenter/qc90/sync/sm/index.html) and run it.
- Deploy all jars in the <release-package>\adapter folder and the stub jar to the <QCS\_Install\_ Dir>\adapters\lib directory. See "Deploying the Adapters" on page 16 for more details.
- 5. Start the HP Synchronizer server by clicking All Programs > HP Quality Center Synchronizer > Start Synchronizer.

## Upgrade for Various Synchronization Scenarios

This section describes how to upgrade various synchronization scenarios to the latest release.

This section includes:

- "Upgrade in Non-PD Environment" below
- "Upgrade from Non-PD Environment to PD Content Pack 9.30.2 (for SM 9.3x only)" on page 202
- "Upgrade from Non-PD Environment to PD Content Pack 9.30.3/SM 9.4x Codeless " on page 202
- "Upgrade from 9.3x with PD Content Pack 9.30.2 to 9.30.3/SM 9.4x Codeless" on page 204

## Upgrade in Non-PD Environment

This section describes how to upgrade various synchronization scenarios to the latest release in SM non-PD environment.

This section includes:

- "SM Change -> QC/ALM Defect" on the next page
- "SM Change -> QC/ALM Requirement" on page 198
- "SM Problem -> QC/ALM Defect" on page 199

Installation and Administration Guide Chapter 11: Upgrading the Integration

- "QC/ALM Defect -> SM Problem" on page 199
- "SM Problem <-> QC/ALM Defect" on page 202

## SM Change -> QC/ALM Defect

To upgrade the synchronization scenario of "SM Changes to QC/ALM Defects" ("Change -> Defect"), perform the following tasks:

### 1. Upgrade SM/SC Customization

No upgrade required.

## 2. Upgrade QCALM Customization

Perform the following steps to add a new field "Created from" in the defect entity:

- a. Log on to QC as a project administrator.
- b. Click **Tools / Customize**. The "QC Project Customization" module opens.
- c. Add the following field for the defect entity in Project Entities (*XY* is a sequential number autogenerated by QC).

Field Name	Field Label	Field Type	Remarks
BG_USER_XY	Created from	String	

- d. In the "QC Project Customization" module, click Workflow.
  - i. Add the following code to the **WizardFieldCust\_Details** and **WizardFieldCust\_Add** event procedures

SetFieldApp "BG\_USER\_XY", True, False, 1, 1

ii. Set the field to Read-only by adding the following lines to the **Bug\_New** and **Bug\_Moveto** subroutines.

Bug\_Fields.Field("BG\_USER\_XY").IsReadOnly=True

e. Save your changes and log out.

### 3. Upgrade QC/ALM Synchronizer Customization

#### Perform the following steps:

- a. In the Quality Center Synchronizer client, edit the link and refresh Schemas.
- b. Add the following constant mapping to the link for this synchronization scenario.

QC	Direction	SM	Constant Value
Created from	<-		Created from SM/SC

c. Save the link.

## SM Change -> QC/ALM Requirement

To upgrade the synchronization scenario of "SM Changes to QC/ALM Requirements" ("Change -> Requirement"), perform the following tasks:

## 1. Upgrade SM/SC Customization

No upgrade required.

### 2. Upgrade QC/ALM Customization

Perform the following steps to add a new field "Created from" in the requirement entity:

- a. Log on to QC as a project administrator.
- b. Click **Tools / Customize**. The "QC Project Customization" module opens.
- c. Add the following fields for the requirement entity in Project Entities (*XY* are sequential numbers auto-generated by QC).

Field Name	Field Label	Field Type
RQ_USER_ <i>XY</i>	Created from	String

- In Requirement Types add the "Created from" field to the Business type requirement.
   The Business type is the default requirement type for incoming requirements (other types can be used).
- e. In the "QC Project Customization" module, click **Workflow**.

In the Script Editor for the Requirements module script, add the following code to **Req\_New** and **Req\_Moveto** to set fields as read-only and place the fields on the tabs (**Req\_New** is called

when a new Requirement is created; **Req\_Moveto** is called when an existing Requirement is opened):

Req\_Fields.Field("RQ\_USER\_XY").IsReadOnly=True

SetReqField "RQ\_USER\_XY", True, False, 1, 1

f. Save your changes and log out.

#### 3. Upgrade QC/ALM Synchronizer Customization

Perform the following steps:

- a. In the Quality Center Synchronizer client, edit the link and refresh Schemas.
- b. Add the following constant mapping to the link for this synchronization scenario.

QC	Direction	SM	Constant Value
Created from	<-		Created from SM/SC

c. Save the link.

## SM Problem -> QC/ALM Defect

No upgrade required.

## QC/ALM Defect -> SM Problem

To upgrade the synchronization scenario of "QC/ALM Defects to SM Problems" ("Defect -> Problem"), perform the following tasks:

### 1. Upgrade SM/SC Customization

Perform the following steps:

- a. Log on to SM/SC as a system administrator.
- b. Click System Definition > Tables > rootcause > Fields.
- c. Add two new fields as follow:

	Туре	
Field	Service Manager 7.0x or later	ServiceCenter 6.2
qcintegration.type	Character	Text
qcintegration.created.from	Character	Text

### d. Customize forms.

Add a field "Created From" to subform "pm.qcint.subform". See "Add Fields" on page 127.

e. Configure WSDL.

Open the "QCIntProblemService" service in WSDL Configuration, and enable the following two fields:

Field	Caption	Туре
qcintegration.type	QCIntegrationType	StringType
qcintegration.created.from	CreatedFrom	StringType

See "Specify the External Access Definition on Service Manager" on page 127 and "Specify the External Access Definition on ServiceCenter" on page 131 for more information.

f. Restart the server if you are using ServiceCenter 6.2.

### 2. Upgrade QC/ALM Customization

Perform the following steps to add a new field "Created from" in the defect entity:

- a. Log on to QC as a project administrator.
- b. Click **Tools / Customize**. The Module "QC Project Customization" module opens.
- c. Rename "Forward as Problem" to "Synchronize with SM Problem"

Project Entities	- Field Settings	
⊕∽∰ Cycle ⊟∽∰ Defect	Field Name:	BG_USER_01
⊞ <u>e-</u> System Fields ⊟ <u>e-</u> User Fields	Field Label:	Synchronize with SM Problem
·····nec Change ID ·····nec Created from	Field Type:	Lookup List 💌
Froblem ID     Synchronize with SM Problem	Field Length:	40
∃∰ Release ∃∰ Release Folder		
E	History	🔲 Required
E- III Run	Masked	🗌 Searchable
⊒∰ Test ⊒∰ Test Instance ⊒∰ Test Set	Lookup List	
	YesNo	👻 New List Goto List

- d. Log off and log on again with the integration account.
- e. Update filter "SMIntegrationView" to "Synchronize with SM Problem = Y".



f. Log off.

#### 3. Upgrade QC/ALM Synchronizer Customization

Perform the following steps:

- a. Deploy the stub jar again. See "Generating/Deploying the Stub" on page 17.
- b. In the QC/ALM Synchronizer client, edit the link and refresh Schemas.
- c. Add the following two constant mappings to the link for this synchronization scenario.

QC	Direction	SM	Constant Value
	-> QCIntegrationType		1
	->	CreatedFrom	Created from Quality Center

d. Save the link.

## SM Problem <-> QC/ALM Defect

No upgrade required.

# Upgrade from Non-PD Environment to PD Content Pack 9.30.2 (for SM 9.3x only)

This section describes how to upgrade various synchronization scenarios from SM 9.3x non-PD environment to SM 9.3x with PD Content Pack 9.30.2 environment.

This section includes:

• SM Change -> QC/ALM Defect

To upgrade the synchronization scenario of "SM Changes to QC/ALM Defects" ("Change -> Defect"), refer to "Add the Subform to a Form" and "Add Rule Set Calculations/Validations" in the SM Change -> QC/ALM Defect chapter.

• SM Change -> QC/ALM Requirement

To upgrade the synchronization scenario of "SM Changes to QC/ALM Requirements" ("Change -> Requirement"), refer to Add the Subform to a Form and Add Rule Set Calculations/Validation in the SM Change -> QC/ALM Requirement chapter.

# Upgrade from Non-PD Environment to PD Content Pack 9.30.3/SM 9.4x Codeless

This section describes how to upgrade various synchronization scenarios from SM 9.3x non-PD environment to SM 9.3x with PD Content Pack 9.30.3. The following steps are also applicable when upgrading synchronization scenarios from SM 9.3x non-PD environment to SM 9.4x Codeless.

This section includes:

• SM Change -> QC/ALM Defect

To upgrade the synchronization scenario of "SM Changes to QC/ALM Defects" ("Change -> Defect"), refer to "Add the Subform to a Form" and "Add Rule Set Calculations/Validations" in the SM Change -> QC/ALM Defect chapter. • SM Change -> QC/ALM Requirement

To upgrade the synchronization scenario of "SM Changes to QC/ALM Requirements" ("Change -> Requirement"), refer to Add the Subform to a Form and Add Rule Set Calculations/Validation in the SM Change -> QC/ALM Requirement chapter.

• SM Problem -> QC/ALM Defect

To upgrade the synchronization scenario of "SM Problems to QC/ALM Defects" ("Problem -> Defect"), perform the following tasks:

a. Upgrade SM Customization

Refer to "Specify the External Access Definition on Service Manager", "Add the Subform to a Form" and "Add Rule Set Calculations/Validations" in the SM Problem -> QC/ALM Defect chapter.

b. Upgrade QC/ALM Synchronizer Customization

Refer to "Define Field Mappings" in the SM Problem -> QC/ALM Defect chapter.

• QC/ALM Defect -> SM Problem

To upgrade the synchronization scenario of "QC/ALM Defects to SM Problems" ("Defect -> Problem"), perform the following tasks:

a. Upgrade SM Customization

Refer to "Add the Subform to a Form" in the QC/ALM Defect -> SM Problem chapter.

b. Upgrade QC/ALM Synchronizer Customization

Refer to "Define Field Mappings" in the QC/ALM Defect -> SM Problem chapter.

• SM Problem <-> QC/ALM Defect

To upgrade the synchronization scenario between QC/ALM Defects and SM Problems ("Defect <-> Problem"), perform the following tasks:

a. Upgrade SM Customization

Refer to "Specify the External Access Definition on Service Manager", "Add the Subform to a Form" and "Add Rule Set Calculations/Validations" in the SM Problem -> QC/ALM Defect chapter. b. Upgrade QC/ALM Synchronizer Customization

Refer to "Define Field Mappings" in the SM Problem -> QC/ALM Defect chapter.

# Upgrade from 9.3x with PD Content Pack 9.30.2 to 9.30.3/SM 9.4x Codeless

This section describes how to upgrade various synchronization scenarios from SM 9.3x with PD Content Pack 9.30.2 to SM 9.3x with PD Content Pack 9.30.3. The following steps are also applicable when upgrading synchronization scenarios from SM 9.3x with PD Content Pack 9.30.2 to SM 9.4x Codeless.

This section includes:

• SM Problem -> QC/ALM Defect

To upgrade the synchronization scenario of "SM Problems to QC/ALM Defects" ("Problem -> Defect"), perform the following tasks:

a. Upgrade SM Customization

Refer to "Specify the External Access Definition on Service Manager", "Add the Subform to a Form" and "Add Rule Set Calculations/Validations" in the SM Problem -> QC/ALM Defect chapter.

b. Upgrade QC/ALM Synchronizer Customization

Refer to "Define Field Mappings" in the SM Problem -> QC/ALM Defect chapter.

• QC/ALM Defect -> SM Problem

To upgrade the synchronization scenario of "QC/ALM Defects to SM Problems" ("Defect -> Problem"), perform the following tasks:

a. Upgrade SM Customization

Refer to "Add the Subform to a Form" in the QC/ALM Defect -> SM Problem chapter.

b. Upgrade QC/ALM Synchronizer Customization

Refer to "Define Field Mappings" in the QC/ALM Defect -> SM Problem chapter.

• SM Problem <-> QC/ALM Defect

To upgrade the synchronization scenario between QC/ALM Defects and SM Problems ("Defect <-> Problem"), perform the following tasks:

a. Upgrade SM Customization

Refer to "Specify the External Access Definition on Service Manager", "Add the Subform to a Form" and "Add Rule Set Calculations/Validations" in the SM Problem -> QC/ALM Defect chapter.

b. Upgrade QC/ALM Synchronizer Customization

Refer to "Define Field Mappings" in the SM Problem -> QC/ALM Defect chapter.

# Post-Upgrade Tasks

## If you are using the integration solution release 1.00:

A new parameter **Socket timeout (Minutes)** has been introduced since Patch 1. To leverage this parameter, you need to export/import links to make this new parameter available on the Quality Center Synchronizer client.

Perform the following steps:

- Export all links as XML files in the Quality Center Synchronizer client by clicking Link > Export > Link Configuration into XML File....
- 2. Delete each link in the HP Quality Center Synchronizer client by clicking **Link** > **Delete**.
- 3. Restore all links by importing the XML files into the HP Quality Center Synchronizer client by clicking Link > Create From > Link configuration XML File....

# Appendix A: Error Messages

This appendix describes the following categories of error messages:

- "Required Fields" on the next page
- "Installation" on page 208
- "Configuration" on page 208
- "Runtime" on page 211
- "XML Validation" on page 215

# **Required Fields**

The following field names are hard-coded.

Field Name	Module	Action	Error Message/Symptom
qcintegration.type	SM Database	Synchronization	No errors in the log and the records failed to be synchronized.
qcintegration.project	SM Database	Synchronization	No errors in the log and the records failed to be synchronized.
QCIntChangeService	SM WSDL Configuration	Generate/Deploy Stub	The stub for Service Manager Change generation failed.
QCIntChange	SM WSDL Configuration	Create a link	System.Web.Services.Protocols.SoapException:java.lang.reflect. InvocationTargetException
ChangeNumber	SM WSDL Configuration	Synchronization	Query: SM ChangeManagement: Can not getRecordIDs. Synchronize: Processing synchronization events failed. Error : java.lang.NoSuchMethodError
Modified	SM WSDL Configuration	Synchronization	Query: SM ChangeManagement: Can not getRecordIDs. Synchronize: Processing synchronization events failed. Error : java.lang.NoSuchMethodError / Query: SM ProblemManagement: Can not getRecordIDs. Synchronize: Processing synchronization events failed. Error : java.lang.NoSuchMethodError
QCIntProblemService	SM WSDL Configuration	Generate/Deploy Stub	Stub for Service Manager Problem generation failed.

Field Name	Module	Action	Error Message/Symptom
QCIntProblem	SM WSDL Configuration	Create a link	System.Web.Services.Protocols.SoapException:java.lang.reflect. InvocationTargetException.
ProblemID	SM WSDL Configuration	Synchronization	Query: SM ProblemManagement: Can not getRecordIDs. Synchronize: Processing synchronization events failed. Error: java.lang.NoSuchMethodError.

# Installation

MSG_ ID	Message	Cause	Solution
INS_1	The stub for Service Manager Change generation failed.	The Change WSDL URL is not valid.	<pre>Provide the valid Change WSDL URL. (For example, http://localhost:13080/sc62server/PWS/QCIntChangeService.wsdl).</pre>
INS_2	The stub for Service Manager Problem generated failed.	The Problem WSDL URL is not valid.	<pre>Provide the valid Problem WSDL URL. (For example, http://localhost:13080/sc62server/PWS/QCIntChangeService.wsdl).</pre>

# Configuration

MSG_ ID	Message	Cause	Solution
CFG_ 1	Can not select "Change Management" and "Problem Management" from the endpoint 2 type.	SM adapter cannot be loaded successfully.	Ensure the stub, adapter and dependency jars are in < <i>QCS_Install_Dir</i> >\adapters\lib.

MSG_ ID	Message	Cause	Solution
CFG_ 2	No create/delete event on change entity is allowed.	For "Change->Defect" and "Change->Requirement" in the Events tab, 'Create a / Delete its corresponding record in the other endpoint' for QC Endpoint Events is selected.	Select <b>Do nothing</b> in the radio-box.
CFG_ 3	No delete event on the problem entity.	In the Events tab, 'Delete its corresponding record in the other endpoint' is selected.	Select <b>Do nothing</b> for all Deletion (Full Synchronization Only) in the Events tab.
CFG_ 4	Missing connection parameter: UserName.	'User name' is empty.	Enter a user name in the Connectivity tab.
CFG_ 5	Missing connection parameter: Service URL.	'Service URL' is empty.	Enter a Service URL in the Connectivity tab.
CFG_ 6	Missing connection parameter: QC Project.	'QC Project' is empty.	Enter a QC Project name in the Connectivity tab.
CFG_ 7	Missing connection parameter: Service URL.	'Service URL' is empty.	Enter a Service URL in the Connectivity tab.
CFG_ 8	Connection parameter: Configuration File Path is not valid.	File path is invalid and the specified file does not exist.	Enter a valid configuration file path name in the Connectivity tab (or leave it empty).

MSG_ ID	Message	Cause	Solution
CFG_ 9	To connect to endpoint of type SM ChangeManagement. Error: com.hp.qc.synchronizer.adapters.exceptions. AdapterException: Fail to connect to SM: Connection refused: connect. ERROR #2- Fail to connect to SM: Connection refused: connect.	SM Server is shutdown or not available.	Start the SM Server or make it available.
CFG_ 10	ERROR #1- adapter.CONNECTION_FAILURE : Failed to connect to endpoint of type SM ChangeManagement. Error: com.hp.qc.synchronizer.adapters.exceptions. AdapterException: Fail to connect to SM: The web service of SM is not reachable! ERROR #2- Fail to connect to SM: The web service of SM is not reachable!	Web service is not available (for example, is not configured).	Make the Web service available.
CFG_ 11	ERROR #1- adapter.CONNECTION_FAILURE : Failed to connect to endpoint of type SM ChangeManagement. Error: com.hp.qc.synchronizer.adapters.exceptions. AdapterException: The URL of SM web service is not valid! ERROR #2- The URL of SM web service is not valid!	URL format is wrong.	<pre>Correct the URL. The format is: http://<sm server="">:<port>/sc62server/PWS/ [QCIntChangeService QCIntProblemService].wsdl</port></sm></pre>

MSG_ ID	Message	Cause	Solution
CFG_ 12	Retry times must be an integer between 0 and 3. (0 means disabled).	The value for the parameter <b>Retries on Locked Record</b> in the Advanced tab is out of range (0~3).	Enter an integer (0 ~ 3) for <b>Retries on Locked Record</b> in the Advanced tab.
CFG_ 13	Retry interval must be an integer between 1 and 10.	The value for parameter <b>Retry</b> Interval(Seconds) in the Advanced tab is out of range (1 to 10).	Enter an integer (1 to 10) for <b>Retry Interval(Seconds)</b> in the Advanced tab.
CFG_ 14	Socket timeout must be an integer between 0 and 120. (0 means default timeout).	The value for parameter: <b>Socket Timeout (Minutes)</b> in the Advanced tab is out of range (0 to 120).	Enter an integer (0 to 120) for <b>Socket Timeout</b> <b>(Minutes)</b> in the Advanced tab.

# Runtime

MSG_ ID	Message	Cause	Solution
RUN_ 1	Required field <field Name&gt; can not be empty or SPACE filled.</field 	Synchronized null /space value to a required field from sponsor to receiver.	Ensure that required field values are not null or filled with spaces.
RUN_ 2	error when reading web service response from SM: Resource Unavailable	Synchronize updates from QC side to locked change/problem records.	Close the locked records in SM side.

MSG_ ID	Message	Cause	Solution
RUN_ 2	Error when reading web service response from SM: Not authorized	<ul> <li>There are two possible causes:</li> <li>Insufficient rights for the SM Integration user when creating/updating defect/requirement in QC or running synchronization to create/update the corresponding change/problem.</li> <li>Maximum active logins for integration account is exceeded.</li> </ul>	<ol> <li>Check the rights of integration account.</li> <li>Check and make sure that <b>Unlimited</b> session in the Security tab is selected.</li> </ol>
RUN_ 3	Error 23scxmlapi(23) - XML DOM exception caught - code 5 msg An invalid or illegal XML character is specified	Synchronization is performed with an illegal WSDL caption.	Correct the WSDL configuration.
RUN_ 4	Update failed 1/2/3, retry in 10 seconds, error message=	Record is locked in SM.	Close the locked record.
RUN_ 5	Update failed for 3 times, skip, error message=	Record is locked in SM.	Synchronize the record manually or run a full synchronization to run all missing updates.
RUN_ 6	Cannot get field for <field Name&gt;.</field 	SM adapter cannot get a field.	Ensure that the stub jar has been generated correctly.

MSG_ ID	Message	Cause	Solution
RUN_ 7	Value cannot be reached for <field name="">.</field>	SM adapter cannot find this field from the stub class.	Ensure that the stub jar has been generated correctly.
RUN_ 8	Mapping error, no such property <property name=""> defined in type <type name&gt;.</type </property>	SM adapter cannot find this property.	Ensure that the stub jar has been generated correctly.
RUN_ 9	Error during setting value for key <key name=""> with value <value>.</value></key>	Dynamic model cannot find this key.	Ensure that the stub jar has been generated correctly.
RUN_ 10	Mapping file <file path=""> cannot be found!</file>	The configuration file isn't found.	Please check the configuration file.
RUN_ 10	<module builder="" class<br="">name&gt; cannot be created because of <message>.</message></module>	SM adapter cannot load a specified class.	Ensure that the stub jar has been generated correctly.
RUN_ 11	Can not convert to <target class name&gt; from value <value>.</value></target 	Value cannot be converted to target type.	Ensure that the WSDL does not expose non- supported data types.
RUN_ 12	Exception when getting SM response, return code: <return code="">.</return>	Problem with SM communication.	Refer to references for error messages.

MSG_ ID	Message	Cause	Solution
RUN_ 13	Error when reading web service response from SM, record is locked [changeID= <recordid>], message=<message>.</message></recordid>	Record is locked.	Close the locked record.
RUN_ 14	The data in the ' <field name&gt;' field of record <record id=""> - of file <file name&gt; contains data that does not conform to the SOA data type in datadict.</file </record></field 	The SOAP field data type in the WSDL is not correct. If the field is of the Number type, the value in the database is out of the range of the specified SOAP type. For example, when choosing IntType (data range: (-2,147,483,648 to 2,147,483,647) for a Number field, if this field has a value of 2,147,483,648 (2^31), it will cause this error when reading the record through the web service interface.	If this field has a Number type, choose <b>DecimalType</b> in the WSDL. Otherwise select the correct SOAP type.
RUN_ 15	Unable to create envelope from given source:	The name of a structure field in Service Manager may have non-English characters.	Use English characters in the name of a structure field.
RUN_ 16	QC: findRequirementByld: Failed getting requirement with id: <id> Failed to update, record was not found or deleted on target null</id>	Deleting or removing a record may result in this problem, because incremental synchronization will fail to find the record.	Restore this record or just run a "Full Synchronization" to remove this mapping relationship established for the record before.
RUN_ 17	Invalid byte 2 of 3-byte UTF-8 sequence.	There are some special non-English characters in the values of fields.	Update the WSDL definition for this field in Service Manager by leaving the "Type" field bank instead of specifying "StringType" for this field.

# XML Validation

MSG_ ID	Error Message	Cause	Solution
XML_ 1	Failed to validate the configuration file: cvc-elt.1: Cannot find the declaration of element 'test'.	Root element is not <b>mapping</b> .	Add a root element <b>mapping</b> .
XML_ 2	Failed to validate the configuration file: cvc-complex-type.2.4.b: The content of element 'mapping' is not complete. One of '{module}' is expected. cvc-complex-type.2.4.b: The content of element 'mapping' is not complete. One of '{module}' is expected.	No <b>module</b> element in the <b>mapping</b> element.	Add a <b>module</b> element in the <b>mapping</b> root element.
XML_ 3	Failed to validate the configuration file: cvc-complex-type.2.4.d: Invalid content was found starting with element '{module}'. No child element is expected at this point.	More than two <b>module</b> elements in the mapping file.	Make sure the <b>mapping</b> element has only one or two <b>module</b> elements.
XML_ 4	Failed to validate the configuration file: cvc-enumeration-valid: Value 'others' is not facet-valid with respect to enumeration '[change, problem]'. It must be a value from the enumeration. cvc-attribute.3: The value 'others' of attribute 'name' on element 'module' is not valid with respect to its type, 'ModuleName'.	Name of a module is not <b>problem</b> or <b>change</b> .	The <b>name</b> attribute of a <b>module</b> element should be <b>change</b> or <b>problem</b> .

MSG_ ID	Error Message	Cause	Solution
XML_ 5	Failed to validate the configuration file: cvc-complex-type.2.4.b: The content of element 'module' is not complete. One of '{field}' is expected.	No <b>field</b> element in the <b>module</b> element.	Define <b>field</b> elements in each <b>module</b> element.
XML_ 6	Failed to validate the configuration file: cvc-enumeration-valid: Value 'Unknown' is not facet-valid with respect to enumeration '[String, Number, Date, Attachment, Single_ Value_List, Multi_Value_List]'. It must be a value from the enumeration. cvc-attribute.3: The value 'Unknown' of attribute 'type' on element 'field' is not valid with respect to its type, 'FieldType'.	The <b>field</b> element has a wrong type attribute.	The <b>type</b> attribute of a <b>field</b> element must be enumeration '[String, Number, Date, Attachment, Single_Value_List, Multi_Value_List]'
XML_ 7	Failed to validate the configuration file: cvc-datatype-valid.1.2.1: 'wrong' is not a valid value for 'boolean'. cvc-attribute.3: The value 'wrong' of attribute 'readonly' on element 'field' is not valid with respect to its type, 'boolean'.	The <b>field</b> element has a wrong <b>readonly</b> attribute.	The <b>readonly</b> attribute of a <b>field</b> element should be <b>true</b> or <b>false</b> .
XML_ 8	Failed to validate the configuration file: cvc-enumeration-valid: Value 'wrong' is not facet-valid with respect to enumeration '[mandatory, optional, recommended]'. It must be a value from the enumeration. cvc-attribute.3: The value 'wrong' of attribute 'required' on element 'field' is not valid with respect to its type, 'FieldRequired'.	The <b>field</b> element has a wrong <b>required</b> attribute.	The <b>required</b> attribute of a <b>field</b> element should be <b>mandatory</b> , <b>optional</b> or <b>recommended</b> .
XML_ 9	Failed to validate the configuration file: cvc-complex-type.4: Attribute 'type' must appear on element 'field'.	The <b>field</b> element has no <b>type</b> attribute.	The <b>type</b> attribute must be defined in each <b>field</b> element.

MSG_ ID	Error Message	Cause	Solution
XML_ 10	Failed to validate the configuration file: cvc-complex-type.4: Attribute 'name' must appear on element 'field'.	The <b>field</b> element has no <b>name</b> attribute.	Define a <b>name</b> attribute in the <b>field</b> element.
XML_ 11	Failed to validate the configuration file: cvc-complex-type.2.4.d: Invalid content was found starting with element 'items'. No child element is expected at this point.	The <b>field</b> element has more than one child element <b>items</b> .	Define only one <b>items</b> element in each <b>field</b> element.
XML_ 12	Failed to validate the configuration file: cvc-complex-type.2.4.b: The content of element 'items' is not complete. One of '{item}' is expected.	The <b>items</b> element has no child element <b>item</b> .	Add <b>item</b> elements in each <b>items</b> element.
XML_ 13	Failed to validate the configuration file: cvc-complex-type.4: Attribute 'value' must appear on element 'item'.	The <b>item</b> element has no <b>value</b> attribute.	Define a <b>value</b> attribute for each <b>item</b> element.
XML_ 14	Failed to validate the configuration file: cvc-minLength-valid: Value '' with length = '0' is not facet-valid with respect to minLength '1' for type 'Item'. cvc-complex-type.2.2: Element 'item' must have no element [children], and the value must be valid.	The <b>item</b> element has no <b>text</b> value.	Define a <b>text</b> value for each <b>item</b> element.

MSG_ ID	Error Message	Cause	Solution
XML_ 15	cvc-minLength-valid: Value '' with length = '0' is not facet-valid with respect to minLength '1' for type 'NonEmptyString'. cvc-attribute.3: The value '' of attribute 'value' on element 'item' is not valid with respect to its type, 'NonEmptyString'.	The <b>value</b> attribute has an empty value.	Define a value for the <b>value</b> attribute in each <b>item</b> element.
XML_ 16	cvc-minLength-valid: Value '' with length = '0' is not facet-valid with respect to minLength '1' for type 'NonEmptyString'. cvc-attribute.3: The value '' of attribute 'name' on element 'field' is not valid with respect to its type, 'NonEmptyString'.	The <b>name</b> attribute has an empty value.	Define a value for the <b>name</b> attribute of each <b>item</b> element.
XML_ 17	cvc-enumeration-valid: Value 'Attachment' is not facet-valid with respect to enumeration '[String, Number, Date, Single_Value_List, Multi_Value_List]'. It must be a value from the enumeration. cvc-attribute.3: The value 'Attachment' of attribute 'type' on element 'field' is not valid with respect to its type, 'FieldType'.	The <b>field</b> element has a type of <b>Attachment</b> .	Remove the <b>Attachment</b> type element.
XML_ 18	Fail to validate the configuration file: cvc-datatype-valid.1.2.1: 'xxx' is not a valid value for 'integer'. cvc-attribute.3: The value 'xxx' of attribute 'length' on element 'field' is not valid with respect to its type, 'positiveInteger'.	The <b>field</b> element has an incorrect length attribute value.	Correct the value of the <b>field</b> element.

# 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 Installation and Administration Guide (Defects and Requirements Exchange with HP Service Manager and HP Application Lifecycle Management 1.03 for Process Designer (PD) Content Pack 9.30.2 and 9.30.3)

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!



